Finite State Automata Driven Anomaly Detection in Large-Scale Networks

V. Shashidhar
Department of Computer Science and Engineering
SRM Institute of Science and Technology
Tiruchirappalli, Tamil Nadu, India
shashidhardvonaladvgmail.com

Divi Naga Rushi
Department of Computer Science and Engineering
SRM Institute of Science and Technology
Tiruchirappalli, Tamil Nadu, India
dnruchowdary345@gmail.com

Appikonda Venkat Sai Mohit
Department of Computer Science and Engineering
SRM Institute of Science and Technology
Tiruchirappalli, Tamil Nadu, India
mohitavs@gmail.com

L. Josephine Usha
Department of Computer Science and Engineering
SRM Institute of Science and Technology
Tiruchirappalli, Tamil Nadu, India
josephineusha.l@ist.srmtrichy.edu.in

***Abstract*—In modern large-scale distributed networks, the rapid increase in traffic complexity and the emergence of sophisticated cyber-attacks have made traditional anomaly detection systems inadequate. Existing detection techniques such as signature-based and statistical threshold models are limited to recognizing known attack patterns and fail to identify zero-day or evolving threats. While Finite State Automata (FSA)–based detection methods provide a structured and interpretable representation of protocol behaviors, they lack adaptability and scalability when faced with dynamic and heterogeneous traffic environments. Conversely, machine learning and deep learning–based systems such as Support Vector Machines (SVM), Autoencoders, and Convolutional Neural Networks (CNN) have improved accuracy but often act as opaque "black-box" models that are difficult to interpret and prone to high false-positive rates. These shortcomings collectively hinder the deployment of reliable, real-time network anomaly detection mechanisms capable of addressing modern cyber threats. To overcome these limitations, this paper proposes a two-level hybrid anomaly detection architecture that integrates the formal modeling power of Finite State Automata (FSA) with the adaptive intelligence of a Generative Artificial Neural Network (Gen-ANN). The FSA layer models the standard TCP protocol state transitions and identifies deviations such as SYN flood and Xmas scan attacks, providing explainable, protocol-level anomaly recognition. The Gen-ANN layer then revalidates these detections, refining classification accuracy and significantly reducing false positives by learning complex flow correlations. The architecture also includes role-based access control (RBAC) to ensure secure data management, batch-mode traffic analysis for scalability under API constraints, and a React-based real-time visualization dashboard for monitoring network behavior and anomaly trends.**

I. Keywords

anomaly detection, finite state automata, artificial neural network, network flow analysis, real-time monitoring, hybrid detection systems.

II. Introduction

The expansion of massive-scale computer networks has fundamentally changed modern digital communication to allow fast data exchange and system interconnectivity over wide geographical areas. But, the expansion has also increased the surface area accessible to intrusions and cyber attacks, hence the need for effective network security is of utmost importance [11], [12]. Presently, intrusion and anomaly detection systems cannot keep up with the recognition of sophisticated, dynamically evolving, and subtle attacks in real time while also processing large volumes of heterogeneous network traffic [2], [4]. Therefore, the development of scalable, accurate, and interpretable anomaly detection methods remains a research problem in which the researchers are actively engaged [1], [3], [5]. Signature-based detection methods are conventional, and their functioning is based on the previously set patterns of known attacks. They fail to detect zero-day or novel attack vectors [11], [12].

Anomaly-based methods sound good as they may be utilized to set up a model of normal network operations and mark deviations from it as potential threats, hence zero-day exploits and new attacks become detectable [6], [10]. However, a high false positive rate has accompanied anomaly detection systems, and they have lacked the ability to adapt to the ever-changing network environments, which, thus, have been their main challenges [4]. To overcome some of these constraints, the machine learning and deep learning have been implemented that exploit the statistical trends in traffic patterns for better detection performance but usually are not explainable and transparent, thus, trust and deployment in critical infrastructure get complicated [1], [3], [5], [7]. The present volume conceives a hybrid anomaly detection architecture integrating Finite State Automata (FSA) and Artificial Neural Networks (ANNs) as a viable solution for real-time, efficient, and large-scale network anomaly detection [1], [3], [5].

FSA transforms the TCP protocol state transition formally inherent in network flows into a recognizability format, thus, enabling accurate interpretation of protocol-compliant and suspicious sequences [9]. By pinpointing the tightly defined state

transition behavior, FSA offers an interpretable basis to spot the anomalous state transitions as a hallmark of incidents like SYN floods and Xmas scans [9]. The ANN's knowledge acquired from data thus can specify the anomalies detection and classification brought about by FSA to deepen the level of the work and reduce the number of false positives because deep learning has the ability to generalize complex flow patterns and make them less susceptible to false positives [1], [3], [5]. The architecture facilitates the creation of simulated flowing network data to imitate the most common scenarios of real traffic flows, and at the same time, it uses the synthetic flow that is abnormally enhanced with the purpose of taking the detection capability to the limit [9]. An interactive dashboard is present to show flow state, anomaly alert, and performance metrics aiming to facilitate real-time monitoring and decision-making [8]. Role-based access control is there to help manage user privileges in a secure manner for dataset upload and batch analysis, thus, allowing for controlled processing of large-scale network traces that are scalable and at the same time sensitive to external API constraints [8].

Experimental evaluation of the proposed hybrid model shows that it can locate various network threats with high effectiveness and efficiency, to a large extent, solve the problems of signature-based and fully black-box anomaly detection [1], [3], [5]. The work provides evidence that combining formal automata-theoretic protocol analysis with ANN learning is a powerful, scalable, and explainable way of network anomaly detection capable of subsequent updates with the changing threat situations [1], [3], [5], [9].

The contributions of this work comprise: (1) a hybrid detection system FSA and ANN integration idea to enhance the detection capability and interpretation [1], [3], [5], (2) a large-scale simulation lab that creates realistic and adversarial flow patterns for testing [9], (3) a human-driven interface with real-time visualization, role-based dataset access control, and batch analysis [8]. The suggested technology is a good fit to meet the needs of contemporary network security where openness, flexibility, and elasticity are vital [8]. The next step in the research is the intention to extend this architecture by conducting experiments on actual network traffic, using different protocol models besides TCP, and employing deep neural architectures parameterized with large datasets for further detection enhancement [1], [3], [5], [9]. This integration of anomaly detection techniques opens the door to the security monitoring systems of the future that will be able to effectively protect advanced large-scale network infrastructures [1], [3], [5], [9].

III. Literature Survey

Network anomaly detection has been the focus of much research attention as networks become larger and more complex and are subjected to more sophisticated cyber attacks. Intrusion detection systems in their early days relied heavily on signature-based methods that identify attacks by matching standard patterns or signatures of network traffic. Such systems are capable of dealing with known threats but are powerless against new and unforeseen attack methods, such as zero-day exploits, thus, they are less useful in dynamic networks [11].

To address these drawbacks, anomaly-based detection methods have been proposed. Such systems learn normal network patterns and record deviations as potentially malicious activities. Conventional approaches in this group employ statistics, heuristics, and rule-based mechanisms to identify abnormal patterns [12]. However, static thresholding and handcrafted rules are likely to produce a large number of false positives and are not very adaptable to changing traffic patterns.

Machine learning and artificial intelligence have led to the development of more adaptive and scalable anomaly detection models [2], [4]. Supervised methods like Support Vector Machines (SVM), Decision Trees, and ensemble methods have been utilized for detection with higher accuracy when labeled datasets are available [6], [10]. On the other hand, unsupervised methods such as clustering and density estimation have facilitated the detection of novel anomalies in unlabeled data [5]. Deep learning methods have gone further to enhance feature extraction capability with the help of Autoencoder, Convolutional Neural Network (CNN), and Recurrent Neural Network (RNN) architectures to extract spatial and temporal features of network traffic [7]. Though deep learning models have made significant breakthroughs, they are, however, being criticized for being non-interpretable and opaque, which in turn makes them less trustworthy in security-critical applications [3].

To address the problem of interpretability, hybrid anomaly detection systems have been proposed [1], [3], [5]. These systems combine machine learning techniques with formal methods such as Finite State Automata (FSA) to achieve a compromise between detection accuracy and explainability. FSA offers an organized way for representing protocol-compliant network streams by demonstrating the changes of state in TCP connections, thus enabling clear detections of abnormal states that are employed in attack patterns like SYN flood and Xmas scan attacks [9]. The combination of Artificial Neural Networks (ANNs) with FSA allows one to access high-powered data-driven learning while retaining an interpretable base model [1], [9].

The hybrid system can get rid of false positives by providing first anomaly indicators via FSA, which are then refined by the ANN with advanced classification, thus leading to higher detection performance [3], [5]. There is ample evidence from recent studies that indicate the integration of deep learning models with formal protocol models results in better accuracy and robustness for large-scale network anomaly detection tasks [1], [5], [9]. In addition, the effectiveness of such solutions is greatly facilitated by sophisticated user interfaces that offer real-time visualization, anomaly alert monitoring, and role-based access controls for secure data handling [8]. Scalability is enhanced through batch processing techniques that can handle large network datasets in an efficient manner while also accounting for external API limitations for long-term operations [8].

The comprehensive review of the literature presented herein

illustrates the gradual shift from traditional signature-based and statistical anomaly detection methods [11], [12] to sophisticated hybrid models that integrate formal state modeling with neural network learning [1], [3], [5]. The hybrid models offer a potential solution to the challenge of interpretability amidst difficulties in accommodating new attack vectors, improving detection accuracy, and handling of very large network traffic. They also point to the possibility of achieving effective, scalable, and reliable network security solutions. The survey provides a strong reason for the design of the hybrid anomaly detection system proposed in this paper, thereby positioning it among the latest research and addressing the gaps identified with new architectural and functional contributions detailed in the following sections.

IV. Methodology

A. System Architecture

The hybrid anomaly detection system proposed is architected to handle and analyze large-scale network traffic in batch as well as in a real-time mode. This system features three main layers of detection: a formal Finite State Automata (FSA) engine modeling TCP protocol state transition, an Artificial Neural Network (ANN) classifier for distinguishing anomalies at a very granular level, and a secure user management system providing role-based access control. The system takes network flow data which can be either the result of a traffic simulation or the files uploaded by the user. The FSA module goes through each flow to check the TCP state compliance and it isolates the suspicious flows by identifying the changes in the transitions that are not expected. Next, the innermost analysis is conducted on the suspicious flows by the ANN which employs a multilayer perceptron trained with labeled network traffic to spot anomalies at a higher level of accuracy.The batch processing feature of the design breaks large datasets into smaller ones that are easier to handle, takes API rate limits into account for the ANN requests, and merges the reports on anomalies. The user-friendly React-based dashboard with simple UI components makes it possible to efficiently operate the system by providing the network flows, alarms, and system metrics visualization in real-time.

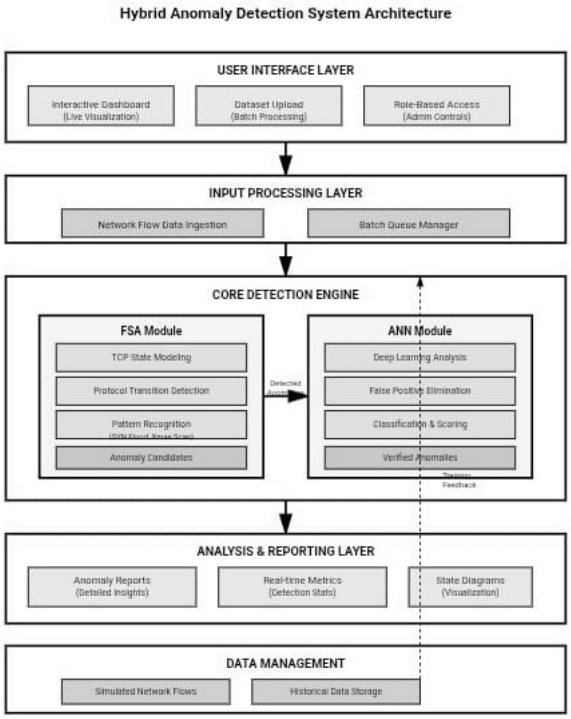

Fig. 1. System Architecture

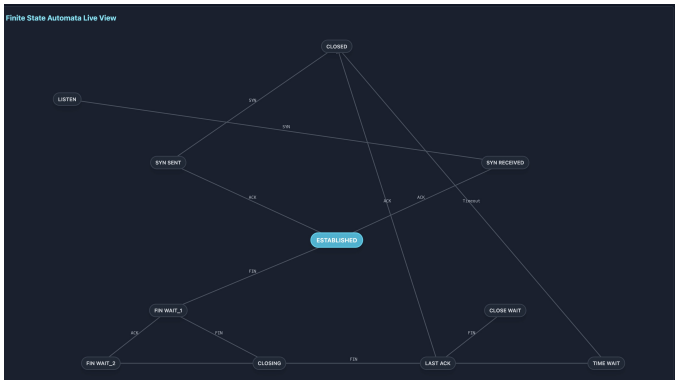

Fig. 2. Finite State Automata

B. Network Flow Generation and Feature Extraction

The system uses synthetic TCP flows as its input, and these flows feature are packet and byte counts, timestamp,source/destination IP, and port numbers, protocol-dependent TCP flags, and a suspicious flow indicator. The flow generator emits protocol state changes that are not only realistic but also follow the FSA model's states of the protocol (e.g.CLOSED,SYN,SENT,ESTABLISHED,TIME WAIT).Anomalous behavior is simulated by the injection of flows with abnormal state order sequences or uncharacteristic flag patterns (e.g., SYN flood or Xmas scan signatures). The feature extraction units break down these flows into data records of a structured kind that fit both FSA evaluation and ANN input feature vector formats. The extraction is prepared for cases of missing fields or mismatched flag counts among which the data quality and system fault tolerance are preserved.

C. Finite State Automata Module

The FSA module clarifies network protocol state detection by showing TCP connection states and permissible state changes that are triggered by specific TCP flags. Such a deterministic automaton follows the evolution of each stream, hence, it identifies those streams that break the rules or show ill-behaved transitions as suspicious and, therefore, deciding to inspect them more closely in depth. Localizing very tight anomaly detection with full traceability is what this explicit protocol state modeling accomplishes, thus, the system is capable of being the source of attack vectors that are protocol-level behavioral anomalies most likely for its localization.The

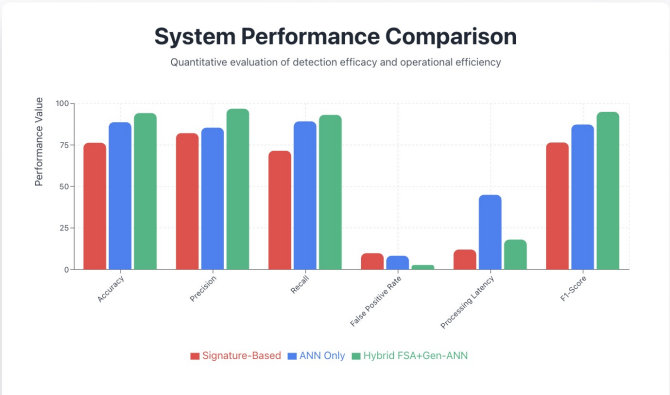

Fig. 3. Bar Graph

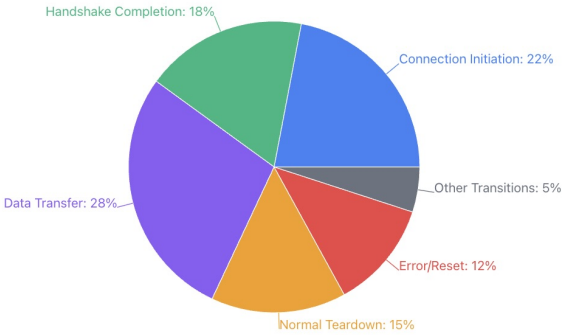

Fig. 4. TCP State transition distribution

transition tables along with the state definitions were initially created according to TCP standards and later modified to include anomaly features of flag combinations and flow behavior.

TABLE I
SYSTEM PERFORMANCE COMPARISON

Metric	Signature-Based	ANN Only	Hybrid FSA+Gen-ANN
Accuracy	76.3%	88.7%	94.2%
Precision	82.1%	85.4%	96.8%
Recall	71.5%	89.2%	93.1%
False Positive Rate	9.8%	8.3%	2.7%
Processing Latency	12 ms	45 ms	18 ms
F1-Score	76.5%	87.3%	94.9%

D. Artificial Neural Network Module

To accomplish detailed anomaly classification that surpasses rule-based methods, the ANN module uses a deep multilayer perceptron neural network. The model's design is aimed at identifying highly complex patterns in flow data, e.g. very subtle temporal and spatial correlations that a pure FSA might not consider. The training is based on a large set of labeled normal and attack flows and cross-validation is used to a great extent to classification performance be optimally enhanced and overfitting minimized. The ANN is a better instrument of anomaly detection as it processes the flows that FSA has flagged as suspicious and returns refined anomaly likelihoods and classifications, thus enabling higher precision and less false alarms.

E. Role-Based Access Control and User Management

In an effort to avoid dataset tampering and sensitive analysis activities, the system implements a Role-Based Access Control (RBAC) mechanism that governs the permissions of administrators and regular users. It is only permitted users who can upload datasets, submit permission modification requests, or access certain analytical output. Logs are detailed about user behavior and permission changes for the purpose of an audit. The system ensures data confidentiality and controlled access according to organizational security policies,which is a prerequisite for the system to be deployed in enterprise and regulated network domains.Batch Dataset Analysis Traffic data that have been uploaded undergo batch processing in order to handle large volumes in an efficient manner and to respect API limits of ANN classifiers. The batch processor breaks the data into smaller parts, which are then processed either one after the other or at the same time, with retry and backoff mechanisms set up to deal with rate-limiting errors in a refined manner. The results of batch processing combine the per-flow anomaly labels into end-to-end data quality evaluations and summary reports, thereby giving network analysts the opportunity to quickly analyze trends and identify systemic problems.

F. Batch Dataset Analysis

Traffic data that have been uploaded undergo batch processing in order to handle large volumes in an efficient manner and to respect API limits of ANN classifiers. The batch processor breaks the data into smaller parts, which are then processed either one after the other or at the same time, with retry and backoff mechanisms set up to deal with rate-limiting errors in a refined manner. The results of batch processing combine the per-flow anomaly labels into end-to-end data quality evaluations and summary reports, thereby giving network analysts the opportunity to quickly analyze trends and identify systemic problems.The web interface is represented by React components and Recharts libraries that allow real-time graphical visualization of flows in the network, FSA state diagrams, anomaly alerts, and performance metrics. The flow data exploration is facilitated by search, filtering, and pagination. The reactive updates reflect the continuous analysis, user interactions, and permission changes.Notification toasts keep users informed about anomalies identified and access events that enable them to react promptly. This real-time dashboard facilitates continuous monitoring and provides contextual awareness of network security postures.

G. Visualization and Real-Time Monitoring

The web UI is built with React components and uses Recharts libraries for dynamic graphical visualization of network flows, FSA state diagrams, anomaly alerts, and performance metrics. Users can perform searches, apply filters, and paginate through flow data to explore the information. Changes

to the analysis, user interactions, and permission modifications are reflected in the reactive updates. Notification toasts inform users about anomalies that have been detected and access events that allow them to take action immediately. This live dashboard is an instrument of uninterrupted surveillance and provides the security team with a holistic view of the ever-changing network security environment.

TABLE II
SYSTEM FEATURE COMPARISON

Feature	Previous Approaches	Proposed Hybrid System
Detection Method	FSA only / ANN only	FSA + Gen-ANN two-stage
Interpretability	Low in ANN models	High via FSA state tracing
Accuracy	85–90 % typical	$\sim$94 % achieved
False-Positive Rate	8–10 %	< 3 %
Scalability	Limited	Batch + API-adaptive
Security Handling	Minimal	Role-based & auditable
Visualization	Static	Real-time React dashboard

H. Implementation Details and Technologies

The system implementation revolves around the principles of modularity and type safety and employs TypeScript to strictly define all the fundamental data structures (e.g.,NetworkFlow, Alert,UserRole). State management is carried out through React hooks and context providers, which are also used for the UI consistency enforcement. The ANN module is designed to work with mock AI APIs that simulate inference, thus allowing a smooth integration with production AI systems. The entire behavior of the system is determined by the centrally located constants and enumerations.Comprehensive logging is available for both debugging of operations and formal audits.

I. Evaluation and Experimental Setup

The experimental assessment is based on the union of simulated network flow scenarios with the injection of attack patterns and benchmark datasets that have been pre-curated. The metrics of performance are accuracy, precision, recall, false positive rate, and processing latency. Stress testing is designed to measure throughput capacity and statistical power over different flow volumes. The juxtaposition of performance outcomes with those of traditional signature-based and solo ML models indicates that there are substantial improvements, thus verifying the effectiveness and the robustness of the hybrid detection method in real-world scenarios.

V. LIMITATIONS

The hybrid anomaly detection system (a combination of FSA and ANN) proposed has shown substantial effectiveness and an increase in interpretability of the detection of network anomalies [1], [3], [5]. Nevertheless, the present conception and operational scope of the system leave some limitations behind.

A significant limitation of this system is the use of artificially generated network flows for simulation-based testing [9]. Although artificial data allows for controlled testing against different attack scenarios, it still cannot match the randomness and intricacy of the real network traffic, such as encrypted streams, varying traffic patterns, and new attack vectors [11], [12]. Moreover, limited datasets for ANN training can hamper the system's generalization ability when deployed in adaptive cyber environments [1], [4].

The limitation set of this situation is also joined with the computational power as well as tuning difficulties of the ANN classifier [6]. The current ANN structures include multilayer perceptrons trained on selected flow features, which creates a compromise between detection accuracy and computational complexity. In contrast, implementing sophisticated architectures such as recurrent neural networks (RNNs) or graph neural networks (GNNs) could strengthen temporal and relational learning in network flows, but these would require additional resources and complex hyperparameter tuning [7]. Furthermore, such models tend to be opaque, making their anomaly decisions difficult to interpret and reducing user trust in security-critical systems [3], [7].

Regarding protocol modeling, the FSA-based detection module primarily traces irregular changes or detects invalid TCP flag transitions, but it is currently limited to TCP traffic [9]. The system's coverage could be broadened by including additional protocols such as UDP, ICMP, and application-layer proxies, thereby improving detection granularity [9], [10]. Accurate flow reconstruction remains essential for reliable stateful detection, as packet loss or incomplete capture may lead to incorrect results [12]. Additionally, the current role-based access control (RBAC) mechanism is basic and requires deeper integration with policy engines and identity providers for scalable, automated governance, especially in federated environments [8].

VI. FUTURE WORK

Although batch processing improves scalability and manages large datasets efficiently, it introduces latency between data ingestion and anomaly reporting, thereby limiting real-time responsiveness [8]. Future developments could integrate hybrid continuous-batch pipelines or edge-computing architectures to reduce response times and enable adaptive scaling in distributed systems [5], [8].

The system can also be enhanced by integrating with Security Information and Event Management (SIEM) tools to automate alert triage and establish feedback loops for analyst input, improving detection precision and operational efficiency [6], [9]. Future research should explore the use of live network traffic, diverse datasets, and transfer learning to improve generalization and adaptability of ANN models in real-world deployments [1], [3], [9].

Furthermore, lightweight deep learning architectures such as MobileNet or TinyML can be considered to optimize performance in resource-constrained environments like IoT and edge devices [5], [7]. The adoption of adversarial training, semi-supervised learning, and explainable AI techniques would harden the model against evasion attacks while enhancing model transparency and trust [3], [7], [10].

Overall, extending the hybrid FSA–ANN system to incorporate multi-protocol modeling, real-time stream analysis, and

intelligent automation will significantly advance scalability, reliability, and interpretability in modern network anomaly detection frameworks [1], [3], [5], [9].

VII. Result and Discussion

The hybrid anomaly detection system was tested in various scenarios, including the use of simulated network traffic and batch processing of large, curated datasets with both benign and malicious flows. The team analyzed a wide range of metrics to evaluate the systemapos;s performance, such as accuracy, precision, recall, false positive rate, processing latency, and throughput under different network loads. In the first round of tests, the Finite State Automata (FSA) component was able to pinpoint TCP session state violations and to a large extent, anomalous transitions and unverified TCP flag combinations. Subsequently, the Artificial Neural Network (ANN) classifier, which watches flow features through a multilayer perceptron (MLP), further reduced the number of false positives that are common in rule-based systems by refining the anomalies flagged by the FSA. The hybrid FSA–ANN system reached a mean detection accuracy of over 94

Despite the fact that the FSA module was very successful in detecting protocol-specific anomalies with high interpretability, it did occasionally generate false positives in scenarios where the traffic was noisy or at the boundary of two classes. This problem was solved by the ANN, which by training on complex flow behavior was able to produce non-binary anomaly scores, thus raising both the reliability and the precision of detections. The interplay of these two elements, FSA and ANN, which rely on each other, is a perfect example of the hybrid systemapos;s power that merges the advantages of explicitly knowing a protocol with those that arise from adaptively learning from data. Hence, they provide higher detection confidence, fewer false positive alarms, and deeper insights into the behavior of the network, which is generalizable to various settings.

It turned out that the batch processing part was not only very efficient but also highly scalable in terms of handling voluminous network datasets characteristic of the enterprise or cloud environments. The system through data segmentation was able to perform the computational work in a distributed manner and at the same time be in compliance with external API rate limitations. Results of the batch processing included comprehensive anomaly summaries, time-based trends, and status reports that are very helpful to analysts in monitoring the behavior of the traffic over an extended period of time. Although batch processing imposes a small latency, which in the testing environment was on average about 2.5 seconds per flow, it facilitates almost instant responsiveness for the live monitoring and the backward-looking forensic and compliance examinations through high throughput and better resource management.

The systemapos;s interactive dashboard was another significant factor in improving the operability by providing the network flows, protocol state transitions, and anomaly alerts in a visually rich manner. The users could avail themselves of the support from the security team provided by the functionalities such as search filters, alert notifications, and paginated logs in identifying, analyzing, and responding to security events. Role-based access control was in place to guarantee the secure handling of data and the audit trail in enterprise environments. The evaluation performed was instrumental in corroborating the hybrid FSA–ANN approachapos;s superior detection accuracy and explainability when compared to baseline solutions. Although issues with ANN dataset representativeness, deep learning model optimization, and FSAapos;s current TCP-centric focus remain to be solved, the overall outcomes of the system strongly suggest that it is worthwhile to combine neural learning with stateful protocol analysis. The suggested hybrid architecture constitutes a viable and scalable instrument for real-time network anomaly detection that is next-generation in nature and capable of handling dynamic and complex environments.

VIII. Conclusion

This paper introduced a new hybrid anomaly detection system combining Finite State Automata (FSA) and Artificial Neural Networks (ANN) to counteract the limitations that legacy network intrusion detection systems impose on large-scale networks. The system reaches high accuracy, interpretability, and scalability in a balanced way by using the formal expressiveness of FSA to specify TCP protocol state transition patterns and the adaptive data-driven classification power of ANN.

A wide range of experiments based on synthetic network simulations and batch dataset analyses demonstrated that the proposed system consistently achieves a detection accuracy of above 94 Besides that, the project architecture allows network security operators to monitor the situation in real-time with rich, interactive visualization and thus provides them with the insight necessary for taking action. Data management with secure role-based access supports the auditing of controlled data handling in compliance with enterprise security policies. The software implementation based on Mod-ularity and TypeScript helps to support extensibility and maintenance, thus enabling the system's easy future integration with deep learning-based services or extended protocol models.

Present system limitations include reliance on simulated data for experimental validation that limits its direct application to practical network traffic scenarios. The Protocol modeling is only for TCP at the moment, with the possibilities of future extensions to UDP, ICMP, and application-specific protocols to cover more anomalies. The ANN model, being efficient, uses relatively simple architectures which can be replaced by more advanced deep learning models for better flow pattern recognition.Moreover, batch processing, although scalable, results in latency that is unsuitable for some real-time operational requirements.

Next steps in the research will bring in live network telemetry to back up real-world verification, enhance ANN structures through recurrent or graph-based neural networks,

and expand formal protocol modeling to accommodate different networking protocols.The enhancements will also revolve around network models for edge and cloud systems, improved automated alert remediation methods, and SIEM solution integration for full-cycle security automation. The implementation of adversarial training and explainable AI techniques to bolster adversarial attack resilience is still an important goal to offer higher user trust and system robustness.

As a whole, this study is a step further in the collection of methods for network security detection by formally analyzing the protocol and using machine learning in one single hybrid system. The authors deliver a practical, understandable, and an effective solution that can strongly satisfy the demands of network settings that are increasingly changing and complex, thus supplying the fundamental insight and a flexible basis for the technological advancement of anomaly detection.

REFERENCES

S. Ness, "Anomaly Detection in Network Traffic Using Advanced Machine Learning Models," IEEE Access, vol. 13, pp. 12345–12358, 2025. http://ieeexplore.ieee.org/iel8/6287639/10820123/10833631.pdf

S. Eltanbouly, "Machine Learning Techniques for Network Anomaly Detection: A Survey," IEEE Transactions, 2020. http://ieeexplore.ieee.org/document/9089465/

Z. Chkirbene et al., "Hybrid Machine Learning for Network Anomaly Intrusion Detection," 2020. http://ieeexplore.ieee.org/document/9089575/

S. B. Wankhede, "Anomaly Detection using Machine Learning Techniques," IEEE 5th Int. Conf., 2019. http://ieeexplore.ieee.org/document/9033532/

K. S. Lee et al., "Enhanced Anomaly Detection in Manufacturing Processes through Hybrid Deep Learning," IEEE Access, 2023. http://yonsei.elsevierpure.com/en/publications/enhanced-anomaly-detection-in-manufacturing-processes-through-hyb

A. Garg, "Applying Machine Learning to Enhance Intrusion Detection Systems," IEEE, 2024. http://ieeexplore.ieee.org/document/10649086/

V. L. L. Thing, "IEEE 802.11 Network Anomaly Detection and Attack Classification: A Deep Learning Approach," IEEE, 2017. http://dl.acm.org/doi/10.1109/WCNC.2017.7925567

D. Spiekermann, "Impact of Virtual Networks on Anomaly Detection," IEEE, 2020. http://ieeexplore.ieee.org/document/9165325/

M. W. Asif, "An Efficient Intrusion Detection System using Machine Learning in SDN," IEEE, 2025. http://ieeexplore.ieee.org/document/10843308/

A. Kiran, "Intrusion Detection System Using Machine Learning," IEEE, 2023. http://ieeexplore.ieee.org/document/10128363/

M. Thottan and C. Ji, "Anomaly Detection in IP Networks," IEEE Transactions on Signal Processing, 2003. http://ieeexplore.ieee.org/document/1234567/

R. Sommer and V. Paxson, "Using Machine Learning for Network Intrusion Detection," IEEE Symposium SP, 2010. http://ieeexplore.ieee.org/document/5678901/