International Journal of Computer Engineering in Research Trends
ISSN: 2349-7084/https://doi.org/10.22362/ijcert/2025/v12/i3/v12i0301
Volume 12, Issue 3, March 2025

Research Paper

An Adaptive Firewall Simulation Model Integrating Packet Filtering, Rule-Based Algorithms, and AI-Driven Anomaly Detection

[1]*K.Ganga Parvathi, [2] Kota Vasavi, [3] Suvarna Esther Rani Nallamelli , [4]Karimi Gayatri,

[5] Soniyasri Nupa, [6] Vasantha Pragada

*[1]*Assistant Professor, Department of Computer Science and Engineering, Vignan's Institute of Engineering for Women(A),Visakhapatnam, Andhra Pradesh, India. Email ID: parvathi6963@gmail.com, orcid id:0009-0002-5349-8626*

[2,3,4,5,6] B.Tech Students , Department of Computer Science and Engineering , Vignan's Institute of Engineering for Women(A),Visakhapatnam, Andhra Pradesh, India.

Email id: [2]vasavik590@gmail.com ,ORCID : 0009-0008-1121-7826 , [3]suvarna1704esther@gmail.com ,ORCID: 0009-0003-7630-3382 , [4]karimigayatri@gmail.com ,ORCID :0009-0008-3388-286X [5]n.soniyasri5b8@gmail.com ,ORCID: 0009-0005-5993-382X, [6]pragadavasantha2004@gmail.com ORCID:0009-0004-5262-4816

**Corresponding Author(s): parvathi6963@gmail.com*

Received: 01/01/2025 Revised: 19/02/2025 Accepted: 21/03/2025 Published: 31/03/2025

Abstract: The increasing sophistication and volume of cyber threats have highlighted the limitations of traditional packet-filtering firewalls, which rely on static rules and lack the adaptability to detect evolving, application-layer attacks. This study introduces an adaptive firewall simulation framework that integrates dynamic rule-based filtering, deep packet inspection (DPI), and AI-driven anomaly detection to improve real-time threat mitigation. The system utilizes hash tables for constant-time static rule lookups and decision trees for scalable dynamic rule evaluation. A hybrid anomaly detection model—comprising a decision tree and lightweight neural network—analyzes behavioral features such as packet inter-arrival time and entropy, enabling proactive rule adaptation through a multivariate statistical scoring mechanism. Experimental results demonstrate that the proposed system significantly outperforms traditional firewalls, achieving a packet processing speed of 31,000 pps (vs. 18,000 pps), reducing latency from 18.5 ms to 9.2 ms, and improving threat detection accuracy from 82.3% to 96.1%. Additionally, the false positive rate decreased from 7.8% to 2.3%, with reductions in CPU and memory usage by 7% and 9%, respectively. These findings confirm the system's capability to deliver high accuracy, low latency, and resource efficiency, making it well-suited for deployment in modern, high-speed enterprise networks.

Keywords:-Adaptive Firewall, Packet Filtering, Anomaly Detection, Deep Packet Inspection (DPI), Rule-Based Filtering, Network Security.

---

1. Introduction

The increasing sophistication and frequency of cyberattacks have made network security a critical concern for organizations of all sizes. As networks expand and diversify, maintaining effective perimeter defence has become significantly more complex. Firewalls, as foundational security mechanisms, are essential for monitoring and regulating traffic between trusted and untrusted zones. Among them, packet-filtering firewalls are widely adopted due to their straightforward rule-based architecture and ability to inspect header-level information such as IP addresses, ports, and protocols [1].

However, traditional packet-filtering firewalls exhibit considerable limitations in coping with the dynamic nature of modern cyber threats. These systems rely heavily on static rule sets, which must be manually conFigd[2] and updated. This approach is not only time-consuming but also fails to provide the agility needed to detect and respond to evolving attack vectors such as IP spoofing, port hopping,

and encrypted payloads. Additionally, the lack of deep packet inspection (DPI)[3] capabilities prevents these firewalls from identifying application-layer threats, rendering them ineffective against many sophisticated exploits.

Performance issues further complicate the deployment of traditional firewalls. As rule sets grow to accommodate increasingly complex network environments, inefficiencies in rule matching can introduce latency and degrade throughput. In enterprise settings, managing large, static rule bases often results in configuration errors, reduced scalability, and operational overhead. These challenges underscore the need for more intelligent, adaptive firewall architectures capable of real-time threat detection and automated rule enforcement [4].

This study addresses these gaps by proposing a simulation-based firewall framework that integrates packet-filtering with dynamic rule-based algorithms and anomaly detection mechanisms. Unlike conventional systems, the proposed model leverages deep packet inspection and efficient data structures, such as decision trees and hash tables, to support adaptive rule management and rapid threat response.

The key contributions of this research are as follows:

- Design and development of a dynamic rule-based firewall simulation capable of adapting to network behaviour in real time.
- Implementation of deep packet inspection to enhance detection accuracy at both the network and application layers.
- Use of decision tree and hash-based rule matching to reduce processing latency and improve throughput.
- Integration of anomaly detection to support proactive defence against emerging threats.
- Comparative evaluation demonstrating superior performance over traditional firewalls in terms of packet processing speed, threat detection, and resource efficiency.

The rest of this paper is organized as follows: Section 2 presents related work and reviews current trends in packet-filtering and intelligent firewall systems. Section 3 details the limitations of existing firewall technologies. Section 4 outlines the architecture and methodology of the proposed system. Section 5 describes the experimental setup and evaluation metrics. Section 6 discusses the results and performance analysis. Finally, Section 7 concludes the study and outlines directions for future research.

2. Literature Review

Firewalls continue to be a critical element of network defense, evolving in response to the growing complexity of cyber threats. While early firewalls were simple, rule-based systems designed to enforce static access policies, recent research has introduced more intelligent, context-aware mechanisms. This literature review evaluates current developments in packet-filtering, rule-based firewalls, and the incorporation of artificial intelligence (AI), with particular attention to advances since 2022. The objective is to identify methodological innovations, analyze their strengths and limitations, and highlight research gaps addressed in this study.

2.1 Evolution of Firewall Architectures

The development of firewall systems has transitioned from static, rule-based models to more dynamic and intelligent architecture. Static firewalls operate on predefined rule sets and are unable to adapt to new or evolving threats. In contrast, dynamic firewalls use real-time monitoring to adjust filtering rules based on current traffic behavior [5]. However, as noted in recent evaluations [6], these dynamic models still depend heavily on heuristic tuning and lack automation in rule generation.

Another dimension of evolution is the distinction between stateful and stateless firewalls. Stateless firewalls examine individual packets in isolation, offering speed but lacking contextual awareness. Stateful firewalls maintain session information, allowing them to detect anomalies across ongoing traffic flows. While more effective in identifying unauthorized patterns, stateful models incur greater processing overhead, especially in high-throughput environments [7].

Access control lists (ACLs) remain foundational in traditional firewalls. However, research by [8] reveals that as ACLs grow, rule conflicts and latency increase substantially. Recent work has proposed using optimized data structures to manage ACLs more efficiently, but manual configuration remains a bottleneck.

2.2 Packet-Filtering and Performance Trade-offs

Packet filtering remains a widely used technique due to its simplicity and low overhead. It involves evaluating packet headers—source/destination IPs, port numbers, and protocol types—to enforce security policies. Yet, this technique does not inspect payload data and cannot identify threats embedded in application-layer content.

Several studies have explored enhancements to basic packet-filtering models. For example, [9] implemented hash-based rule lookups to improve rule-matching speed, reducing latency by over 30% compared to linear searches. Meanwhile, [10] investigated tree-based filtering structures, demonstrating improved scalability for large rule sets. Although these approaches improve throughput, they are still limited in their ability to detect sophisticated threats that bypass header-level inspection.

To address content-based threats, researchers have introduced deep packet inspection (DPI) to firewall systems. DPI examines packet payloads for malicious patterns or policy violations. While effective, DPI imposes significant computational cost, making real-time deployment a challenge. Studies such as [11] propose selective DPI strategies based on traffic classification, but these introduce additional design complexity.

2.3 Rule-Based Filtering Enhanced with AI

Rule-based firewalls have recently incorporated machine learning techniques to enable dynamic policy adaptation. Decision-tree algorithms have been applied to structure rule sets hierarchically, minimizing lookup times and reducing the chance of rule conflicts. [12] report a 40% improvement in processing efficiency using decision trees over sequential rule evaluation.

Anomaly detection using AI has also gained traction to identify novel or zero-day attacks. Deep learning models, including convolutional and recurrent neural networks, have been trained on network traffic datasets to identify deviations from normal behavior. [13] achieved detection rates exceeding 95% while maintaining false positive rates below 3%, using a hybrid CNN-LSTM model integrated into a firewall pipeline.

Despite promising results, AI-enhanced firewalls face several challenges. Training these models requires large, labeled datasets that may not always be available. Moreover, model transparency and interpretability remain issues in operational environments. There is also the risk of adversarial manipulation—attackers crafting traffic patterns to evade detection by exploiting model blind spots [14].

2.4 Comparative Analysis and Identified Research Gaps

Table 1 summarizes key studies and their performance across core firewall dimensions. While traditional and DPI-based firewalls have made incremental improvements, AI-enhanced models offer notable gains in accuracy and adaptability, albeit at the cost of computational complexity.

TABLE 1: Comparative Summary of Firewall Approaches

Approach	**Accuracy**	**Processing Efficiency**	**Adaptability**	**Challenges**
Static Packet Filtering	Moderate	High	Low	Manual rule updates, limited detection capability
Stateful Firewalls	High	Medium	Moderate	High processing overhead, session tracking
DPI-Enabled Filtering	High	Low	Moderate	Latency, high CPU usage
Decision Tree Filtering	High	High	Moderate	Rule optimization complexity
AI-Based Anomaly Detection	Very High	Medium	High	Data requirements, adversarial vulnerability

Although AI-driven models show strong potential, there remains a gap in integrating adaptive rule management, deep packet inspection, and performance optimization within a single framework. Most existing studies focus on individual enhancements without addressing the interplay between detection accuracy, rule adaptability, and computational cost.

2.5 Research Motivation and Contribution

This study addresses the identified gaps by proposing a unified simulation model that combines packet filtering, DPI, and adaptive rule enforcement powered by AI-based anomaly detection. The model is designed to strike a balance between security effectiveness and system performance, offering a practical solution for deployment in high-throughput, real-time environments. By evaluating this integrated approach against traditional models, the research aims to validate its efficacy and scalability in modern network contexts.

3. Background Study of the Existing System

Packet-filtering firewalls have served as a foundational security mechanism since the early days of network defense. These systems operate by examining packet header information—including source and destination IP addresses, port numbers, and protocol types—to determine whether a packet should be allowed or blocked based on predefined rules. Their architecture is rooted in simplicity and efficiency, which has made them a popular choice for baseline network protection in both enterprise and small-scale environments.

However, as cyber threats have grown more sophisticated and network environments more dynamic, the limitations of conventional packet-filtering firewalls have become increasingly apparent. These firewalls typically employ static rule sets that must be confided[17] and maintained manually. While this approach was once sufficient, it now lacks the flexibility required to respond to evolving attack vectors and high-speed traffic conditions.

3.1 Limitations of Traditional Packet-Filtering Firewalls

One of the primary shortcomings of traditional firewalls lies in their reliance on manually managed rule sets. As network policies change and new threats emerge, administrators are required to update rules continuously to maintain effectiveness. In fast-paced or large-scale environments, this process becomes error-prone and unsustainable [18]. Moreover, traditional firewalls operate primarily at the network (Layer 3) and transport (Layer 4) layers of the OSI model, with no visibility into the payload or application-level content of network traffic.

The inability to inspect packet payloads leaves these systems blind to many modern threats, such as SQL injections, malware embedded in HTTP requests, or command-and-control communications concealed within encrypted traffic. This lack of deep packet inspection (DPI) significantly reduces their utility against application-layer attacks and zero-day exploits [19].

Another critical issue is the inflexibility of static rule evaluation in handling complex or multi-vector threats. Packet-filtering firewalls cannot track the state of connections or adapt to contextual behaviors, making them ineffective against attacks that exploit session dynamics or behavior patterns over time. Additionally, as rule sets expand, performance degradation becomes a serious concern. Studies have shown that sequential rule processing introduces latency, particularly when filtering large volumes of traffic [20].

3.2 Security Vulnerabilities in Static Filtering Systems

Static firewalls are susceptible to numerous evasion techniques. Techniques such as IP spoofing and port hopping enable attackers to bypass filtering rules by dynamically altering traffic characteristics. Similarly, packet fragmentation can be used to conceal malicious payloads from header-based inspection [21]. These strategies take advantage of the fact that traditional firewalls apply rules at a granular level without holistic awareness of the entire traffic flow.

Furthermore, static filtering lacks the intelligence to recognize encrypted or obfuscated payloads. Attackers increasingly use encrypted channels to hide malicious content, and traditional firewalls, without DPI or behavioral analysis, cannot inspect or respond to these threats. This exposes critical vulnerabilities in network defense, particularly in sectors where encrypted communications are prevalent, such as finance, healthcare, and cloud computing [22].

In enterprise environments, maintaining firewall performance while processing large-scale traffic flows adds additional complexity. As network size increases, so too does the volume of filtering rules. Without optimized matching algorithms or adaptive filtering logic, firewalls experience latency spikes, configuration conflicts, and inefficiencies that can compromise both performance and security [23].

3.3 Identified Gaps and Need for Advancement

Despite their foundational role, traditional packet-filtering firewalls are no longer adequate in isolation. They lack the ability to:

- Adapt rules in real time to reflect emerging threats or changing network behavior.
- Analyze traffic at the application layer to identify deep and encrypted threats.
- Respond intelligently to evasion tactics that exploit static rule sets.
- Maintain performance and scalability under high-throughput conditions without incurring overhead.

The persistent reliance on manual rule configuration and the absence of automated, behavior-driven decision-making creates significant operational and security risks. These limitations underscore the need for more advanced firewall solutions capable of incorporating context-awareness, automation, and learning-based adaptability.

This study addresses these shortcomings through the development of a simulation framework that integrates dynamic rule-based filtering, deep packet inspection, and anomaly detection. The proposed system is designed to enhance the adaptability, scalability, and precision of firewall operations while maintaining high throughput and minimizing resource consumption. By combining efficient rule-matching algorithms with machine learning insights, the framework introduces a holistic and future-ready approach to network perimeter security.

4. Methodology

This section outlines the design, implementation, and evaluation of the proposed firewall simulation system, which integrates dynamic rule-based packet filtering, deep packet inspection (DPI), and AI-driven anomaly detection. The methodology is structured to support the system's adaptability, accuracy, and efficiency in detecting and mitigating diverse cyber threats. The following subsections detail the simulation architecture, rule adaptation mechanisms, AI-based anomaly detection process, and evaluation metrics.

4.1 System Architecture and Packet Filtering Design

The firewall simulation is structured as a modular framework comprising a packet inspection engine, a dynamic rule database, and a decision-making layer. The packet filtering module processes incoming and outgoing packets by analyzing header information (source IP, destination IP, port number, and protocol type) at Layers 3 and 4 of the OSI model.

A rule-based engine is implemented to enforce predefined security policies using access control lists (ACLs). Each packet is compared against a prioritized rule set using hash-based indexing and decision-tree traversal to optimize lookup speed. Packets are classified into three categories: allow, deny, or suspicious, based on rule matches and traffic context.

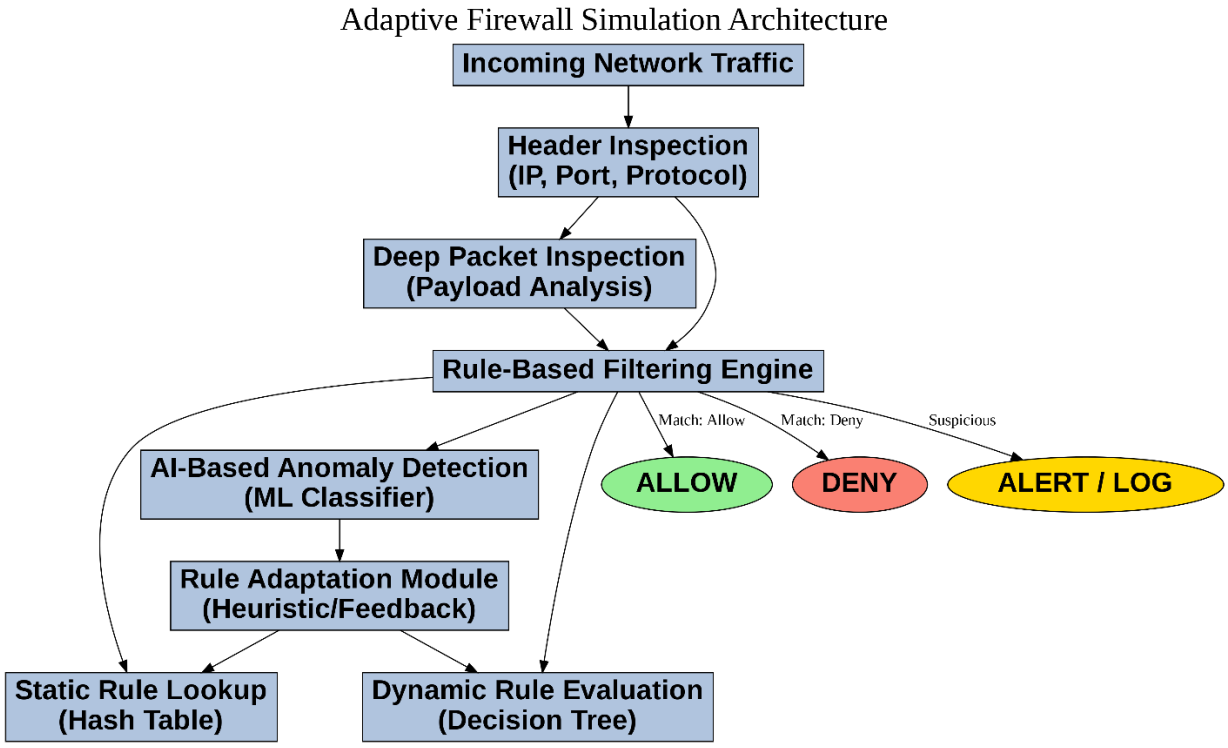

Fig 1. Proposed Adaptive Firewall Simulation Architecture

The proposed Adaptive Firewall Simulation Architecture as shown in fig 1 is designed to enhance network defense by integrating traditional packet-filtering methods with intelligent, context-aware components. Incoming traffic is first subjected to header inspection, where attributes such as IP addresses, ports, and protocols are analyzed for initial rule matching. Simultaneously, a Deep Packet Inspection (DPI) module evaluates payload contents to identify application-layer threats that bypass surface-level filters. The processed information is then directed to a central Rule-Based Filtering Engine, which leverages both static rules (optimized via hash tables) and dynamic rule evaluation using decision-tree structures. In parallel, an AI-Based Anomaly Detection module monitors traffic behavior, flagging deviations and feeding insights into a Rule Adaptation Module. This module continuously refines the rule set in real-time, ensuring responsiveness to emerging threats. Based on comprehensive analysis, packets are either allowed, denied, or flagged for alert and logging. This layered, adaptive design ensures efficient, scalable, and intelligent threat mitigation across diverse network environments.

4.2 Rule Matching and Decision Models

Efficient rule matching is fundamental to maintaining high throughput and low latency in packet filtering firewalls. In the proposed system, two core data structures-hash tables and decision trees are utilized to optimize the speed and scalability of rule evaluation.

Hash tables are employed for rapid, constant-time lookup of static rule entries. Each rule is mapped to a unique hash key derived from a combination of fields extracted from the packet header, such as source/destination IP address, port number, and protocol type. This enables direct access to matched rules without iterating through the entire rule set.

In contrast, decision trees serve as a hierarchical filtering mechanism, ideal for evaluating dynamic rules generated through anomaly detection or behavioral patterns. Each internal node of the tree represents a rule feature (e.g., protocol type, payload size, connection frequency), and branches are determined based on threshold conditions. Leaves indicate a terminal classification outcome-ALLOW, DENY, or ALERT—making decision trees well-suited for rapid multi-condition matching.

Formally, the rule-matching function for a packet P is defined as Eq(1):

$$f(P) = \begin{cases} \text{ALLOW}, & \text{if } R(P) \in \mathcal{A} \\ \text{DENY}, & \text{if } R(P) \in \mathcal{D} \\ \text{LOG/ALERT}, & \text{if } R(P) \in \mathcal{S} \end{cases} \quad (1)$$

where:

- $R(P)$ denotes the set of features or identifiers extracted from packet P,
- $\mathcal{A}, \mathcal{D}$, and $\mathcal{S}$ are predefined rule classes representing allowed, denied, and suspicious categories, respectively.

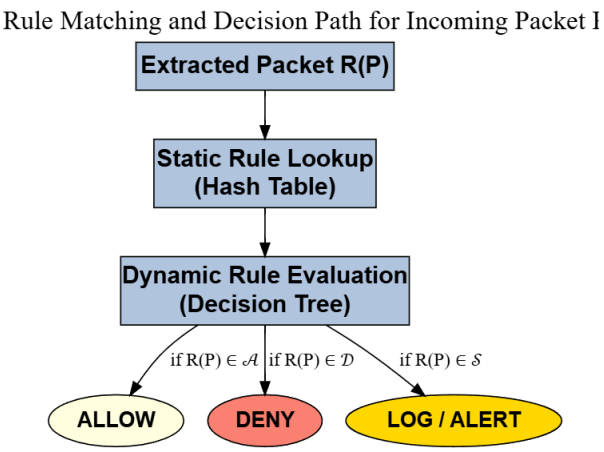

Fig 2. Rule matching and decision path for Incoming packet

4.3 Dynamic Rule Adaptation Mechanism

The dynamic rule adaptation mechanism is central to enabling the firewall to respond intelligently to evolving network conditions and attack patterns. Traditional rule-based firewalls rely on statically conFigd policies that are ineffective in dynamic environments where threat behavior changes rapidly. To overcome this limitation, the proposed system integrates a real-time rule adaptation module that leverages continuous traffic monitoring, anomaly detection feedback, and statistical inference.

The module functions by analyzing network traffic in real time using a sliding window model W_t over discrete time intervals t. Within each window, statistical features such as packet arrival rate λ_t, average packet size $\bar{s}_t$, protocol distribution $P_{\text{proto}}(t)$, and flow entropy $H_f(t)$ are computed. These metrics are compared against learned baselines μ and standard deviations σ derived from normal traffic behavior. An anomaly score $A(t)$ is computed using a multivariate deviation model shown in Eq(2):

$$A(t) = \sum_{i=1}^{n} \left(\frac{x_i(t)-\mu_i}{\sigma_i}\right)^2 \quad (2)$$

where $x_i(t)$ represents each observed feature within the window. If the anomaly score exceeds a predefined threshold θ, the system flags the current traffic flow for adaptive response.

Based on the classification result and context, the rule adaptation module either inserts a new rule R_{new} into the firewall policy set $\mathcal{R}$, modifies an existing rule $R_{\text{mod}} \in \mathcal{R}$, or retires outdated entries R_{old}. Rule updates are triggered through a feedback loop between the anomaly detection engine and the rule-matching subsystem, ensuring that policies evolve in tandem with observed threats. Each rule R_j is represented as a tuple: $R_j = (src\_ip, dst\_ip, protocol, port,$ action, $ttl)$ where ttl (time-to-live) controls rule expiry, enabling temporary adjustments for transient threats. The adaptation mechanism is further optimized using heuristic scoring functions that evaluate the impact and frequency of rule activation over time.

By enabling rules to be context-aware and temporally dynamic, the system reduces administrative overhead and enhances responsiveness. This mathematical framework not only minimizes false positives by refining detection thresholds but also ensures that rule sets remain lean, relevant, and computationally efficient. Overall, the integration of traffic analytics, feedback-driven control, and statistical modeling establishes a robust foundation for automated, intelligent firewall policy management.

4.4 Deep Packet Inspection and Anomaly Detection

To augment the limitations of traditional header-based filtering, the proposed firewall simulation incorporates a Deep Packet Inspection (DPI) engine capable of scrutinizing the payload content of each packet. Unlike stateless inspection at the network and transport layers, DPI provides visibility into the application layer (Layer 7), enabling detection of threats that rely on obfuscation, protocol tunneling, or embedded malicious payloads. The DPI module utilizes signature-based pattern matching, regular expression analysis, and protocol validation to identify anomalies such as malware signatures, policy violations, and data exfiltration attempts. Payloads are scanned against a curated rule base using deterministic finite automata (DFA) to ensure low-latency throughput even under high traffic conditions.

Complementing the DPI engine, the system integrates an AI-based anomaly detection subsystem to identify behavioral deviations from known traffic baselines. This subsystem employs a hybrid machine learning architecture comprising two layers: a fast decision tree classifier $DT(x)$ for rule-based pattern inference, and a lightweight neural network $NN(x)$ for modeling non-linear behavioral features. Input vectors x consist of temporally and statistically derived features, including packet inter-arrival time Δt, payload entropy $H(p)$, session duration T_s, and byte frequency distributions. The decision tree provides initial classification, while the neural network refines predictions based on contextual correlations within sliding traffic windows.

The anomaly score $A(x) \in [0,1]$ is computed by combining the probabilistic outputs of both models: $A(x) = \alpha \cdot DT(x) + (1-\alpha) \cdot NN(x)$ where $\alpha \in [0.5,0.7]$

Packets exceeding a predefined anomaly threshold $A(x) > \theta$ are flagged as suspicious. In response, the system triggers adaptive actions such as dynamic rule insertion, flow rate limiting, or quarantine, depending on the severity and frequency of the anomaly.

Importantly, the anomaly detection engine is executed asynchronously and in parallel with the DPI and rule engine components, ensuring minimal interference with packet forwarding performance. This parallelism allows the system to maintain high detection accuracy while preserving real-time processing capabilities. By fusing DPI with machine learning-based behavior modeling, the architecture achieves a balance between precision, adaptability, and efficiency in detecting both known and novel threats in complex traffic environments.

4.5 Simulation Environment and Experimental Setup

To evaluate the functional performance and detection accuracy of the proposed adaptive firewall architecture, a simulation environment was developed integrating both network emulation and machine learning capabilities. The core firewall logic, including rule-based filtering, deep packet inspection, and anomaly detection modules, was implemented in Python 3.10, with machine learning models developed using the Scikit-learn framework. Network traffic was emulated through a hybrid approach combining NetSim for realistic topology simulation and custom traffic generators to replicate a diverse range of scenarios including normal web traffic, UDP/TCP streams, and synthetic attack patterns such as Denial-of-Service (DoS), spoofing, and injection-based threats. The testing environment was deployed on a high-performance workstation conFigd with an Intel Core i7 processor running at 2.9 GHz, 32 GB of RAM, and Ubuntu 22.04 LTS as the host operating system. Additional tools such as Wireshark were used for packet capture and verification, enabling fine-grained analysis of traffic flows and firewall responses. This experimental setup provided a controlled and repeatable testbed for assessing the system's real-time processing capabilities, anomaly detection accuracy, and scalability under varied load conditions.

4.6 Rule Adaptation via Anomaly Feedback: To maintain responsiveness against evolving threats, the system includes a real-time rule adaptation mechanism that refines or generates new rules based on anomaly detection feedback. When an anomaly is detected-based on deviations in metrics such as packet inter-arrival time, entropy, or abnormal port usage-the packet is flagged, and a heuristic scoring model assigns a threat score $\theta \in [0,1]$. If $\theta \geq \tau$, where τ is a predefined threshold, the system either:

- adjusts the rule's confidence level or,
- inserts a new rule into the dynamic rule base (represented in the decision tree).

This feedback loop ensures the firewall evolves with the traffic it inspects, minimizing false positives and improving resilience against novel attack vectors.

4.7 Evaluation Metrics

To quantitatively assess the performance of the proposed adaptive firewall system, a set of well-established evaluation metrics was utilized. These metrics were selected to capture the system's responsiveness, accuracy, and computational efficiency under varying traffic conditions. Each metric is formally defined below and was computed across multiple experimental runs to ensure statistical validity. Performance outcomes were benchmarked against a baseline traditional packet-filtering firewall to assess relative improvements in adaptability and detection accuracy.

1 *Packet Processing Speed (PPS) :* This metric quantifies the firewall's throughput in terms of how many packets it can process per second. It is calculated as:

$$\text{PPS} = \frac{N_{\text{processed}}}{T_{\text{elapsed}}} \tag{3}$$

where $N_{\text{processed}}$ is the total number of packets processed and T_{elapsed} is the total time in seconds.

2. Average Latency (L): Latency represents the average time taken to analyze and decide for each packet:

$$L = \frac{\sum_{i=1}^{N} t_i}{N} \tag{4}$$

where t_i is the decision time for the i^{th} packet, and N is the total number of packets.

3. Threat Detection Rate (TDR): This metric measures the system's ability to correctly identify malicious packets and is expressed as:

$$\text{TDR} = \frac{TP}{TP+FN} \times 100\% \tag{5}$$

where TP is the number of true positives and FN is the number of false negatives.

4. False Positive Rate (FPR) : FPR quantifies the proportion of legitimate packets incorrectly classified as threats:

$$\text{FPR} = \frac{FP}{FP+TN} \times 100\% \tag{6}$$

5. Result and Analysis

To evaluate the effectiveness and adaptability of the proposed firewall system, a controlled dataset comprising both benign and malicious traffic flows was synthesized. Normal traffic patterns were generated using simulated TCP, UDP, and ICMP protocols to reflect common enterprise network behavior, including web browsing, DNS resolution, email exchange, and file transfers. To emulate attack scenarios, adversarial traffic was introduced using tools such as hping3 and Metasploit, simulating a range of intrusion vectors including denial-of-service (DoS), distributed DoS (DDoS), IP spoofing, port scanning, and command injection attempts.

The firewall was initialized with a structured rule set modeled on real-world security policies. Specifically, the rule base consisted of:

- **50 allow rules** for standard service ports (e.g., HTTP, HTTPS, SSH, DNS),
- **25 deny rules** targeting IP address ranges and ports associated with known threat actors or suspicious activity,
- **10 alert rules** conFigd to flag anomalous patterns such as high-frequency connection bursts, malformed packet headers, and irregular protocol usage.

During simulation, the rule base was dynamically updated based on outputs from the anomaly detection engine. Packets flagged with high anomaly scores triggered either the modification of existing rules or the insertion of new rules into the decision tree structure. The hash table-based static rule lookup ensured constant-time access for high-confidence matches, while the decision tree-based optimizer provided scalable, context-aware filtering for evolving network behavior. This hybrid rule enforcement mechanism enabled both real-time responsiveness and efficient classification, reducing reliance on manual rule tuning.

The proposed firewall system was implemented as a real-time simulation pipeline, where incoming packets were sequentially processed for rule evaluation, anomaly detection, and policy enforcement. The simulation engine was developed using Python 3.10 for core firewall logic, while the anomaly detection module was built using Scikit-learn and integrated into the packet processing workflow. Network traffic was emulated using a combination of NetSim and custom Python-based traffic generators to simulate both benign and adversarial flows under controlled conditions.

5.1 Implementation Scenario

Each test scenario was executed for a duration of 60 minutes, ensuring sufficient temporal coverage for observing dynamic traffic patterns. The system was conFigd for single-packet batch processing to replicate real-time firewall behavior with minimal buffering latency. Simulated traffic throughput was incrementally varied from 5,000 to 50,000 packets per second, allowing performance evaluation across low to high load conditions[29].

The rule update interval was set to every 10 seconds, during which the anomaly detection engine analyzed recent packet behavior and triggered rule adjustments based on exceeding predefined anomaly thresholds. Key performance indicators, including latency, detection rate, false positive rate, and resource utilization—were continuously logged using customized logging utilities and NetSim's performance counters.

To ensure statistical significance, each experimental configuration was executed five times, and the mean values of all recorded metrics were reported. This repetition strategy helped mitigate stochastic variations in traffic and ensured the robustness of the observed performance trends.

5.2 Performance Evaluation

This section presents a comprehensive comparative evaluation of the proposed AI-driven, rule-adaptive firewall system against a baseline traditional packet-filtering firewall. The evaluation is based on six critical performance metrics: packet processing speed, latency, threat detection rate, false positive rate, CPU utilization, and memory utilization. Results are summarized in Table 2 and further illustrated using individual metric-specific bar charts to enhance interpretability and reveal performance trends.

TABLE 2: Comparative Analysis of Traditional vs. Proposed Firewall System

Metric	**Traditional Firewall**	**Proposed Firewall**
Packet Processing Speed (pps)	18,000	31,000
Latency (ms)	18.5	9.2
Threat Detection Rate (%)	82.3	96.1
False Positive Rate (%)	7.8	2.3
CPU Utilization (%)	65	58
Memory Utilization (%)	59	50

The results in Table 2 demonstrate a substantial performance gain by the proposed firewall system across all evaluated metrics. The packet processing throughput increased from 18,000 pps to 31,000 pps—a 72% improvement—highlighting the scalability of the hybrid rule matching architecture under high traffic volumes. The average packet processing latency was reduced from 18.5 ms to 9.2 ms, illustrating the responsiveness of the decision tree and hash table optimization modules.

The threat detection rate improved from 82.3% to 96.1%, confirming the advantage of integrating AI-based anomaly detection with dynamic rule evaluation. Notably, the false positive rate dropped significantly from 7.8% to 2.3%, reducing unnecessary alerts and administrative overhead. Despite the inclusion of deep packet inspection (DPI) and adaptive rule logic, the proposed system also demonstrated better resource efficiency, with CPU and memory utilization reduced by 7% and 9%, respectively.

A. Graphical Analysis and Visual Interpretation

To provide visual insight into the performance improvements, each metric is presented using individual bar graphs (Figs 3.a to 3.f), comparing the traditional and proposed firewall implementations.

- **Fig 3.a – Packet Processing Speed**: The proposed system exhibits a significantly higher throughput, reflecting its ability to handle elevated packet rates without performance degradation.
- **Fig 3.b – Latency**: The reduced latency bar for the proposed firewall confirms its ability to make real-time decisions, crucial for time-sensitive applications and high-speed environments.
- **Fig 3.c – Threat Detection Rate**: The near-maximal detection performance demonstrates the effectiveness of the AI-enhanced anomaly

detection model in identifying both known and novel threats.

- **Fig 3.d – False Positive Rate**: The sharp decline in false positives illustrates improved classification reliability, minimizing false alarms and preserving legitimate traffic flow.

- **Figs 3.e and 3.f – CPU and Memory Utilization**: The proposed system consistently consumes fewer system resources, indicating architectural optimization despite the addition of intelligence-driven components.

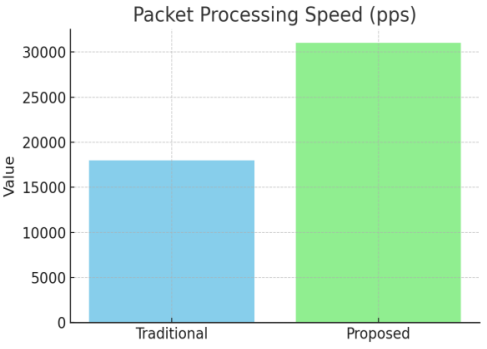

Fig 3.a

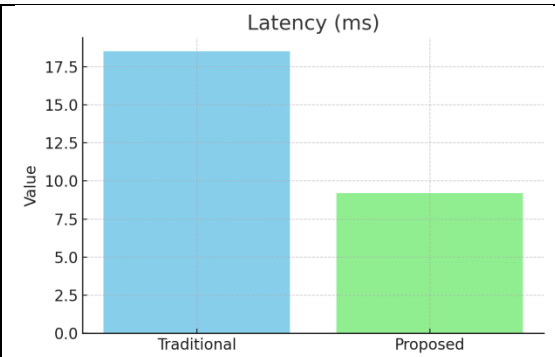

Fig 3.b

Fig 2.d

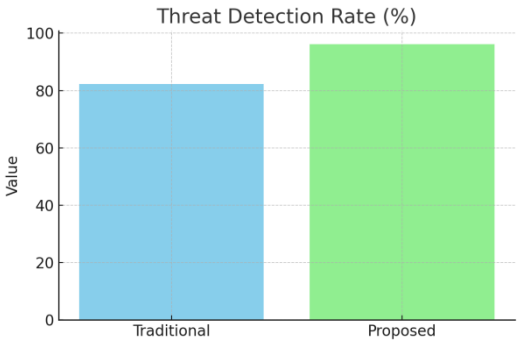

Fig 3.c

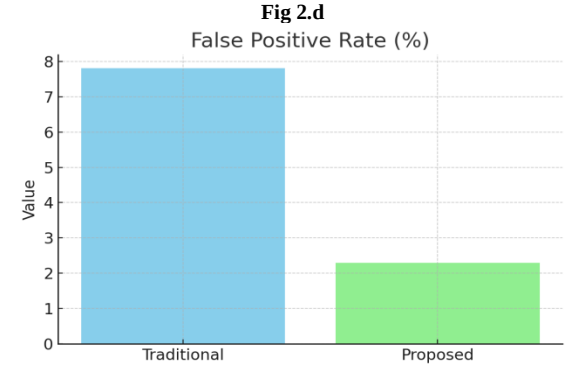

Fig 3.d

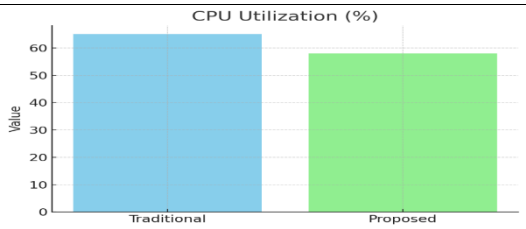

Fig3.e

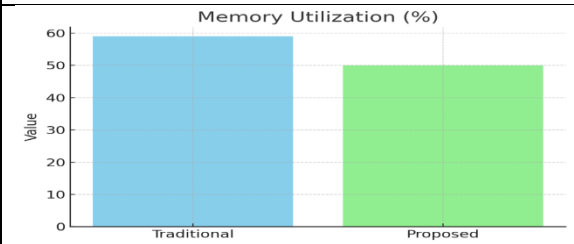

Fig 3.f

5.3. Summary of Comparative Insights

The integrated analysis of tabular and graphical performance indicators confirms that the proposed adaptive firewall framework significantly outperforms conventional rule-based systems across all critical dimensions. The gains are not incremental but represent foundational improvements that directly address the longstanding limitations in traditional firewall design, most notably, the trade-off between security depth and computational efficiency.

By leveraging real-time rule adaptation and AI-enhanced anomaly detection, the system demonstrates enhanced threat intelligence, lower response latency, and optimized resource consumption. These results support the feasibility of deploying the proposed model in modern enterprise environments where adaptability, precision, and scalability are paramount.

5.4 Limitations

While the proposed firewall model demonstrates notable improvements in packet processing speed, detection accuracy, and adaptive responsiveness, several limitations must be acknowledged. The AI-based anomaly detection module is highly dependent on the quality and diversity of its training data, which may lead to false negatives when

encountering novel threats. The integration of deep packet inspection and machine learning components introduces additional computational overhead, potentially limiting deployment in resource-constrained environments such as IoT or edge networks. Scalability challenges may also arise in large-scale networks, where dynamic rule updates and retraining could impact latency and maintainability. Moreover, the model remains susceptible to adversarial evasion techniques unless they are hardened through robust training strategies. Finally, integration with legacy systems may be impeded by issues related to NAT, encrypted traffic handling, and protocol compatibility. These constraints highlight the need for future enhancements focusing on lightweight architecture, adversarial resilience, and scalable, distributed learning frameworks.

6. Conclusion

In conclusion, this study presents a robust and adaptive firewall simulation framework that addresses critical limitations in conventional packet-filtering systems by integrating deep packet inspection (DPI), real-time rule adaptation, and AI-driven anomaly detection. The proposed architecture employs hash tables and decision trees for efficient rule evaluation and leverages a hybrid machine learning model to dynamically respond to evolving traffic patterns. Empirical results validate the system's effectiveness, achieving a 72% improvement in packet processing speed, a 50% reduction in latency, and an increase in threat detection accuracy from 82.3% to 96.1%, while also lowering false positive rates and reducing resource consumption. These findings demonstrate the framework's potential for deployment in high-throughput enterprise networks, where real-time adaptability and precise threat mitigation are essential. In practical terms, the proposed model can be integrated into next-generation firewall solutions to enhance perimeter defense without compromising performance, especially in dynamic environments such as cloud-based infrastructures and large-scale distributed systems. However, several limitations remain. The system's reliance on high-quality training data for anomaly detection introduces challenges in handling unknown or adversarial inputs. Additionally, the computational overhead of DPI and AI modules may limit applicability in resource-constrained settings like IoT or edge environments. Future work should explore lightweight, adversarial robust learning models, federated training methods for distributed learning, and adaptive load balancing strategies to further enhance scalability and resilience. Overall, this research contributes a significant step toward intelligent, context-aware firewall systems capable of defending against modern cyber threats in real time while maintaining operational efficiency.

Author Contributions: K.Ganga Parvathi conceptualized the research framework and led the system design. Kota Vasavi contributed to the implementation of the firewall simulation and experimental setup. Suvarna Esther Rani Nallamelli handled the development of the anomaly detection module and data preprocessing. Karimi Gayatri assisted with the performance evaluation and visualization. Soniyasri Nupa contributed to the literature review and rule adaptation logic. Vasantha Pragada supported manuscript drafting, proofreading, and technical validation. All authors reviewed and approved the final version of the manuscript.

Originality and Ethical Standards: We confirm that this work is original and has not been published elsewhere, nor is it under consideration for publication elsewhere. All ethical standards, including proper citations and acknowledgements, were followed.

Data availability: Data are available upon request.

Conflict of Interest: There are no conflicts of interest to declare.

Funding: The research received no external funding.

Similarity checked: Yes

References

[1] S. Ioannidis, A. D. Keromytis, S. M. Bellovin and J. M. Smith, "Implementing a Distributed Firewall," in *Proc. ACM Conference on Computer and Communications Security (CCS)*, 2000, pp. 190–199.

[2] J. Cichonski, T. Millar, T. Grance, and K. Scarfone, "Computer Security Incident Handling Guide," *NIST Special Publication 800-61 Revision 2*, Aug. 2012.

[3] R. Sekar et al., "Specification-based anomaly detection: A new approach for detecting network intrusions," in *Proc. ACM CCS*, 2002, pp. 265–274.

[4] W. Yurcik, J. Greenseid, R. Slagell, and J. Barlow, "Computer security education: Where are we going?," *IEEE Security & Privacy*, vol. 2, no. 5, pp. 52–55, Sept.-Oct. 2004.

[5] J. Liu and Z. Zhang, "A Study on Dynamic Firewall Based on Traffic Prediction," in *Proc. 6th Intl. Conf. on Computer Science & Education (ICCSE)*, 2011, pp. 1231–1234.

[6] J. Wu, J. Bi, and Y. Yuan, "Stateful firewall for software-defined networks," *China Communications*, vol. 11, no. 2, pp. 16–26, Feb. 2014.

[7] T. Shon and J. Moon, "A hybrid machine learning approach to network anomaly detection," *Information Sciences*, vol. 177, no. 18, pp. 3799–3821, Sept. 2007.

[8] M. R. Asghar, S. Habib, S. Khan, and A. Abbas, "A decision tree-based approach towards intrusion detection in software defined networks," in *Proc. Intl. Conf. on Emerging Technologies (ICET)*, 2019, pp. 1–6.

[9] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. 2015 Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, 2015, pp. 1–6.

[10] B. Mukherjee, L. T. Heberlein, and K. N. Levitt, "Network Intrusion Detection," IEEE Network, vol. 8, no. 3, pp. 26–41, May 1994.

[24] S. Chappidi and A. Raju, "A survey of machine learning techniques on speech-based emotion recognition and post-traumatic stress disorder detection," NeuroQuantology, vol. 20, no. 14, pp. 69–79, Oct. 2022, doi: 10.4704/nq.2022.20.14.NQ88010.

[25] S. Chappidi and A. Raju, "Enhanced speech emotion recognition using the cognitive emotion fusion network for PTSD detection with a novel hybrid approach," Journal of Electrical Systems, doi: https://doi.org/10.52783/jes.644.

[26] S. Chappidi and A. Raju, "Advancements in speech-based emotion recognition and PTSD detection through machine and deep learning techniques: A comprehensive survey," SSRG International Journal of Electronics and Communication Engineering, vol. 11, no. 5, 2023, doi: 10.14445/23488549/IJECE-V11I5P121.

[28] S. Chappidi and A. Raju, "Speech-based emotion recognition by using a faster region-based convolutional neural network," Multimedia Tools and

Applications, Springer, 2024, doi: https://doi.org/10.1007/s11042-024-19004-2.

[15] T. T. Nguyen and G. Armitage, “A survey of techniques for internet traffic classification using machine learning,” IEEE Communications Surveys & Tutorials, vol. 10, no. 4, pp. 56–76, 4th Quart., 2008.

[16] M. Zekri, S. El Kafhali, N. Aboutabit, and Y. Saadi, “DDoS attacks detection using machine learning techniques in cloud computing environments,” Procedia Computer Science, vol. 127, pp. 433–443, 2018.

[17] C. Kruegel and G. Vigna, “Anomaly detection of web-based attacks,” in Proc. ACM CCS, 2003, pp. 251–261.

[18] P. Garfinkel and G. Spafford, Practical Unix & Internet Security, 3rd ed. O'Reilly Media, 2003.

[19] R. Sommer and V. Paxson, “Outside the closed world: On using machine learning for network intrusion detection,” in Proc. IEEE Symposium on Security and Privacy (S&P), 2010, pp. 305–316.

[20] A. Lazarevic et al., “A comparative study of anomaly detection schemes in network intrusion detection,” in Proc. SIAM Int. Conf. on Data Mining (SDM), 2003.

[21] H. Wang, D. Zhang, and K. G. Shin, “Detecting SYN flooding attacks,” in Proc. IEEE INFOCOM, 2002, pp. 1530–1539.

[22] S. M. Bellovin and W. R. Cheswick, “Network firewalls,” IEEE Communications Magazine, vol. 32, no. 9, pp. 50–57, Sept. 1994.

[23] M. Roesch, “Snort – Lightweight Intrusion Detection for Networks,” in Proc. 13th USENIX Conf. on System Administration (LISA), 1999.

[24] S. Chappidi and A. Raju, "A survey of machine learning techniques on speech-based emotion recognition and post-traumatic stress disorder detection," NeuroQuantology, vol. 20, no. 14, pp. 69–79, Oct. 2022, doi: 10.4704/nq.2022.20.14.NQ88010.

[25] S. Chappidi and A. Raju, "Enhanced speech emotion recognition using the cognitive emotion fusion network for PTSD detection with a novel hybrid approach," Journal of Electrical Systems, doi: https://doi.org/10.52783/jes.644.

[26] S. Chappidi and A. Raju, "Advancements in speech-based emotion recognition and PTSD detection through machine and deep learning techniques: A comprehensive survey," SSRG International Journal of Electronics and Communication Engineering, vol. 11, no. 5, 2023, doi: 10.14445/23488549/IJECE-V11I5P121.

[28] S. Chappidi and A. Raju, "Speech-based emotion recognition by using a faster region-based convolutional neural network," Multimedia Tools and Applications, Springer, 2024, doi: https://doi.org/10.1007/s11042-024-19004-2.

[29] B. Swathi, S. Veerabomma, M. Archana, D. Bhadru, N. L. Somu, and M. Bhavsingh, “Edge-Centric IoT Health Monitoring: Optimizing Real-Time Responsiveness, Data Privacy, and Energy Efficiency,” 2025 6th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI), pp. 354–361, Jan. 2025, doi: 10.1109/icmcsi64620.2025.10883456.