ISSN: 2093-5374 / E-ISSN: 2093-5382

An Intelligent AI-Driven Firewall Framework for Zero-Day Threat Detection in Mobile and Interactive Network Environments Using Deep Learning and Explainable AI

V. Asha[1*], and S. Kanaga Suba Raja[2]

[1*]PhD Research Scholar, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tiruchirappalli, Tamil Nadu, India. ashamephd@gmail.com, https://orcid.org/0009-0002-7567-1063

[2]Professor and Head, Department of Computer Science and Engineering, SRM Institute of Science and Technology Tiruchirappalli, Tamil Nadu, India. skanagasubaraja@gmail.com, https://orcid.org/0000-0002-3626-1806

Received: March 02, 2026; Revised: April 08, 2026; Accepted: May 25, 2026; Published: June 30, 2026

Abstract

Increasingly, existing cybersecurity infrastructures struggle against complex attacks like Advanced Persistent Threats (APTs), polymorphic malware, encrypted attacks, and zero-day attacks, rendering them ineffective against traditional signature-based and rule-based firewalls. This paper presents a new intelligent AI-based firewall architecture incorporating hybrid deep learning, explainable AI (XAI), federated learning (FL) and multi-agent reinforcement learning (MARL) to enable adaptive and private threat detection on mobile and interactive networks. An end-to-end CNN-LSTM-Autoencoder model was designed and implemented to learn the spatial and temporal characteristics of network traffic, effectively detecting the anomalous traffic patterns indicative of novel attacks. SHAP and LIME techniques were then adopted to produce transparent and interpretable security decisions and federated learning was utilized to enable secure, collaborative sharing of threat intelligence among different organizations without sharing confidential data. Multiple well-known benchmark datasets were used for performance evaluation along with a massive proprietary dataset, SRM-TFC-2024, consisting of 127 million records and spanning the classification and analysis of network flows across various traffic types and attacks, as well as the more established ones, CICIDS2017, UNSW-NB15, NSL-KDD, and CSE-CIC-IDS2018. The experimental results exhibited consistent and high detection rates on all datasets, with classification accuracies between 79% to 82%, precision varying from 87% to 92%, recall from 84% to 90%, F1-score from 85% to 90%, and the false positive rate remained very low between 1% and 3%. The rate of zero-day threat detection was 82% on bothCICIDS2017 and SRM-TFC-2024 datasets, indicating the high potential of the anomaly-based detection method, and the integrated end-to-end system, which consists of the architecture of CNN-LSTM-Autoencoder, federated learning, and Multi-Agent Reinforcement learning, is called QX-FedMARL. Through comparison, the QX-FedMARL firewall works best compared to other rules-based NGFWs. The comparative analysis showed that the QX-FedMARL firewall performs significantly better than the other rule-based NGFW, signature-based IDS, ML-based Firewall, and AI-NGFW architectures with classification precision of 92.0%, recall of 90.0% and the highest F1-score of 91.0% on the tested datasets. These findings suggest that the combined approach of deep learning, explainable AI, federated intelligence and adaptive

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 17, number: 2 (June-2026), pp. 630-649. DOI: *10.58346/JOWUA.2026.I2.035*

*Corresponding author: PhD Research Scholar, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tiruchirappalli, Tamil Nadu, India.

reinforcement learning presents a robust, scalable, and transparent cybersecurity framework for next-generation intelligent, autonomous firewall systems.

Keywords: AI-Driven Firewalls, Deep Learning, Explainable AI, Zero-Day Threat Detection, Federated Learning, Convolutional Neural Network, Reinforcement Learning.

1 Introduction

Advances in technology have caused considerable changes in cybersecurity. Cyber criminals nowadays make use of advanced tools such as APT, polymorphic malware, and encrypted malware, using artificial intelligence for reconnaissance (Abu Al-Haija & Zein-Sabatto, 2020). The traditional approach of identifying viruses through signatures and static rules is becoming less effective as adversaries keep evolving their strategies with the help of artificial intelligence. The interconnectivity of networks and cloud computing, along with the Internet of Things and remote access, has added many entry points to cyberspace (Ucar & Ozhan, 2017). The exploitation of these new vulnerabilities by adversaries makes cyber threats faster.

The use of conventional approaches in firewall security, which include signature-based and rule-based approaches, has become less effective and reliable due to the rise in complex cyber-attacks such as APTs, polymorphic malware, and zero-day attacks. The existing firewall systems cannot detect the new kinds of attacks and offer real-time cybersecurity solutions. In addition, they are ineffective against new attack methods such as encryption. In this case, the current cybersecurity approaches lack the capacity to counter the new methods that cyber attackers have incorporated into their attacks. Moreover, the increasing complexity of computer networks has led to the creation of many possible attack surfaces. The need to develop efficient solutions to the complex attacks and the challenges in developing new firewall systems has led to the emergence of the requirement for smart and adaptive firewalls. The challenges facing modern cybersecurity systems include privacy issues in threat information sharing, among others. This paper will seek to address the problem using deep learning architectures, XAI, and federated learning.

Interactive network environments are defined as traffic patterns that have the following features: real-time, bidirectional, and sensitive to delay communication between the user, device, and services. Examples are: video conferencing, online gaming, VolP, remote desktop session, AR/VR, and industrial control systems. These environments are a valuable attack surface for an attacker, as the persistent session-oriented nature of the traffic can be exploited for payload injection, session hijacking or Denial-of-Service (DoS) attack. The critical part with interactive network traffic is that the detection must provide extremely low-latency with high-accuracy, which becomes a driving factor for the development of lightweight, edge-deployment-compatible deep learning models described in this paper.

The key contributions of this proposed model are as follows:

1. The deep learning processing is designed using CNN-LSTM and it extracts intricate spatial and temporal features of traffic automatically from raw network flow. So it is faster and more precise in detecting abnormal attacks, including zero-day and encrypted traffic vectors.
2. Privacy-preserving techniques such as Federated Learning allow effective and secure exchange of intelligence on detected threats, and XAI approaches like SHAP and LIME provide improved transparency in automated processes.

Table 1: Evolution of firewall technologies across six generations

Generation	Era	Technology	Key Innovation	Limitation
1st	1980s	Packet Filtering (Header-based ACLs)	First automated network access control based on IP/port headers	No state awareness; easily spoofed; cannot inspect payload
2nd	1990s	Stateful Inspection (Connection tracking)	Tracks TCP/UDP session states; context-aware allow/deny decisions	Still reactive; no application-layer visibility; rule-heavy management
3rd	2000s	Deep Packet Inspection (Application layer)	Payload inspection, protocol decoding, and app-layer visibility	Cannot detect encrypted/obfuscated traffic; performance bottleneck
4th	2010s	Next-Gen Firewall (NGFW + IDS/IPS)	Integrated IDS/IPS, VPN, user-identity awareness, SSL inspection	Zero-day blind; high false positive rates; limited ML capability
5th	2020s	AI/ML-Driven Adaptive Firewalls	Supervised or unsupervised ML, behavioural analytics, XAI, federated learning	Black-box decisions; privacy concerns; adversarial vulnerability
6th	2026+ (Bridging Generation)	Autonomous Agentic AI Firewalls (Zero-Trust + DL-Secured)	Self-healing policies through multi-agent RL; Anomaly detection with deep learning; Zero trust mesh; Blockchain for audit trails;LLM-driven threat reasoning.	Large-scale interpretability of deep learning models; robustness to adversarial attacks; expensive GPU resources; risks of hallucinations in LLM decisions.

In table 1 enumerates the six generations of firewall evolution technology (Praise et al., 2020). Generation 1 (1980's) depends on packet filtering by IP & port headers, which are susceptible to spoofing; Generation 2 (1990's) enabled stateful inspection that tracks state of TCP/UDP sessions, however it is limited at the application-layer; Generation 3 (2000's) introduced deep packet inspection & protocol decoding; it lacks the ability to deal with encrypted & disguised traffic; Generation 4 (2010's) incorporates IDS/IPS, VPN & SSL inspection; however it cannot detect zero-day attacks with acceptable false positive rate. 5th-generation firewalls appeared in 2020s by taking advantage of machine learning, behavioral analysis, XAI and federated learning for dynamic threat detection; however, the challenge still exists for black-box decisions, vulnerability against adversaries, and data privacy issue. The characteristic features of 6th-generation firewalls are multi-agent reinforcement learning (MARL), deep learning-based anomaly detection, zero-trust architecture and agentic AI for autonomous decision; these features are under pioneering research in areas such as this paper. This work presents one of the steps towards that generation of firewalls by implementation of main 6th-generation features in a deployable deep learning solution in 2026.

The structure of the paper is as follows: In Section 1, Introduction, explain why we should use Artificial Intelligence to design the firewall, the deep learning architectures, XAI and federated learning. Section 2, Literature Review, provides details about firewall development progress, especially with

regard to Artificial Intelligence and Hybrid Computing. Section 3, Proposed Methodology, provides a description of the role of XAI, Federated Learning, and Reinforcement Learning. Section 4 describes experimental results which compare AI and conventional firewall, the performance is also measured. The conclusion of section 5 summarizes the findings and recommends future work.

2 Literature Review

Firewall technology was first developed in the 1980s as simple packet filters, evolving into highly intelligent and efficient tools to confront the cybersecurity challenges of today. Packet filters in first generation firewalls were rule based, with no context, later in the 1990s, stateful inspection was able to track connections between hosts, but these were still a reactive type of system. More accurate threat detection came with machine learning in the 2010s; though with limitations such as the high false positive rate, issues with the training data-set, and poor detection rates of zero-day threats. Deep learning techniques such as the convolutional neural network, recurrent neural network (especially the long-short term memory architecture) and the transformer have automated feature selection for intrusions, increasing the detection accuracy. Whilst hybrid models (employing machine learning and deep learning) are highly accurate and efficient, these are hard to explain and resource intensive. However, through employing federated learning with a variety of more complex deep learning models, it is possible to implement a privacy protecting solution and collaborative threat intelligence.

A study used deep neural networks for identifying anomalies in web application firewalls. It highlights the application of machine learning within firewall technology, which is an important point of the proposed framework (Moradi Vartouni et al., 2019). A similar study presented a hybrid deep learning approach for internet security, which is relevant to the proposed work as it discusses hybrid models for improved cybersecurity performance (Ertam, 2019). The similar architecture that we use is a hybrid approach utilizing a deep learning-based system which takes a hybrid of both CNN to extract spatial features and LSTMs to learn traffic pattern, in order to detect anomalies within firewalls. A study came up with a firewall architecture using reinforcement learning algorithms for securing cloud infrastructures (Praise et al., 2020). This relates directly to the choice of reinforcement learning (RL) in implementing adaptive security strategies in the proposed firewall infrastructure.

A study designed an adaptive ensemble machine learning model for detecting cyber-attacks. It focused on using multiple machine learning models that help in detecting and classifying cyber-attacks. Multiple models are integrated to ensure better accuracy in detection (Gao et al., 2019). Another study highlights the use of machine learning techniques to optimize firewall configurations and improve their effectiveness in detecting cyber-attacks (Appelt et al., 2018). The findings of a paper concerning the optimal solution of firewall anomalies and the contributions made by another paper to improving firewall settings by using automated solutions are consistent with the suggested framework (Bringhenti et al., 2023; Bringhenti et al., 2022). The use of optimisation techniques for making firewall systems more efficient.

A research paper enhances web application firewalls for SQL injection prevention using proxy grammar. It aimed to improve firewall accuracy in preventing SQL injections by employing novel techniques (Coscia et al., 2024). A novel technique for compressing the firewall policy was proposed, and it discusses an innovative strategy to minimize the size of the firewall rule set without compromising the level of security provided (Cheng et al., 2018). An approach towards anomaly detection in firewall policies was introduced, and it is clear from that work that the detection of any abnormal activity on firewalls is important for enhancing the security of networks (Togay et al., 2021). The purpose of similar

research is firewall optimisation aimed at sustainability in network security. It deals with the problem of efficient configuration of firewalls that are energy-efficient and can cope with advanced cyber threats (Bringhenti & Valenza, 2024).

In this context, a review of technological advancements in relation to deep learning algorithms for cybersecurity purposes, specifically focusing on the growing importance of utilizing deep learning to enhance threat detection and decision-making in cybersecurity (Dixit & Silakari, 2021). In the research, an attempt was made to design a security solution using the machine learning technique for the IoT network by taking into account the special requirements of IoT networks, including resource constraints (Bagaa et al., 2020). A study discussed reliable IP solutions in multi-tenant cloud FPGA environments, emphasizing the significance of cloud infrastructure that is both secure and efficient (Ahmed et al., 2022).

A survey paper on deep reinforcement learning for adaptive network security policy optimization could also be useful for the theoretical basis of the multi-agent control implemented in the proposed framework (Wu, 2025; Damo et al., 2025). In the article on edge-cloud artificial intelligence security architecture, we could leverage some concepts; it was described as an artificial intelligence-enabled security framework for secure distributed and internet-of-things-enabled environments (Zhang et al., 2022). The survey paper on computing power networks includes three important dimensions: architecture, resource allocation, and AI services enablement. In general, these results confirmed the role that AI is playing in network resource optimization (David et al., 2026). A deep learning-based adaptive control approach to achieve real-time network traffic classification was introduced in a paper where the described techniques, such as prediction modeling and adaptive feedback loops, were found to be conceptually aligned with the proposed framework, adaptive RL approach for the firewall (Zhao et al., 2026).

The literature reveals the weaknesses associated with the use of conventional firewalls in detecting advanced cyber-attacks such as APTs, polymorphic malware, and zero-day attacks. The Artificial Intelligence Firewalls, including techniques like machine learning, deep learning, and hybrid deep learning-classical computation, were proven to improve detection ability and efficiency of operation. Federated learning allows for collaboration in a secure manner without compromising the privacy of information in compliance with GDPR regulations. Techniques such as SHAP and LIME ensure explainability in AI, which increases confidence in automation processes. Multi-agent reinforcement learning ensures dynamic and timely protection throughout the network layer.

3 Proposed Methodology

The developed system combines the four above mentioned AI techniques; (1) The deep learning-based hybrid architecture (CNN-LSTM-Autoencoder) for anomaly and zero-day threat detection with zero-day anomaly scoring, (2) The explainable AI (SHAP, LIME) for a decision transparency, (3) The Federated Learning with differential privacy to distribute intelligence across multiple systems privately, and (4) The Multi-agent RL for the adaptive control of policy distributed across all network layers. The agent-based system can be seen in figure 1 below. This integrated system is designated as QX-FedMARL, representing a Quad-Technique framework (Hybrid DL, XAI, Federated Learning, and Multi-Agent Reinforcement Learning)

3.1 Explainable AI (XAI) for Transparent Decision-Making

3.1.1 SHAP (SHapley Additive exPlanations)

SHAP (SHapley Additive explanations), on the other hand, presents a model-explainability approach for AI-powered firewalls by attributing the predictions made by the model to specific input variables based on cooperative game theory. The input variable here is considered a player; the higher the score, the greater the likelihood of a threat, whereas lower scores indicate low risks.

3.1.2 LIME (Local Interpretable Model-Agnostic Explanations)

Local Interpretable Model-agnostic Explanations (LIME) reduce model complexity through the use of locally interpretable models that are created for individual predictions. This involves generating data perturbations, collecting predictions, and fitting a simplified model, such as linear regression, to explain the prediction. LIME differs from other global explanations because it uses locally interpretable models rather than globally interpretable ones, thus helping cybersecurity professionals understand why a network flow is flagged as either malicious or benign. This process helps in improving incident response, investigating false positives, and refining machine learning models.

3.2 Deep Learning Architecture for Threat Detection

This proposed framework utilizes a CNN-LSTM-Autoencoder as a hybrid deep learning architecture for zero-day threat detection. It consists of a 3-layer CNN (64, 128, 256 filters with ReLU activation function) to extract spatial pattern information of network flows. A 2-layer LSTM (256 units in hidden layer) takes into account the temporal aspect of network traffic, which can identify slow moving attacks such as APTs. The Autoencoder learns a representation of benign traffic and identifies attacks through reconstruction error in a zero-day fashion. The softmax layer outputs an 8-category classification of attack type. The whole system is trained end-to-end with an Adam optimizer (learning rate=0.001 and epochs=50). This 50-epoch run constitutes the centralized benchmark/global initialization phase.

3.3 Federated Learning for Privacy-Preserving Threat Intelligence

Privacy of federated learning helps to facilitate threat intelligence through collaboration without resorting to the central collection of confidential information. Conventional techniques suffer from privacy leakage, lack of trust, and security vulnerabilities since cross-border transfer of data is controlled under GDPR. Companies do not wish to share any confidential network patterns due to confidentiality issues, and a centrally collected network can pose a threat of cyber-attack. Federated learning offers a safe and reliable way to share cybersecurity intelligence.

3.4 Industrial Federated Architecture

The suggested framework allows different industries (finance, health, and education) to collectively conduct an analysis of potential cyber-attacks while protecting their data privacy. Federated learning is where all parties involved are able to create a local model, share gradients, and develop a global threat model. The architecture described above guarantees that this approach works across different industries and is resilient to potential future attacks, and legal compliance requirements are minimized, as well as being less dependent on trust. The multi-agent reinforcement learning (RL) firewall will provide distributed and adaptive protection for all network layers, such as the perimeter, DMZ, corporate LAN, and the cloud environment.

Federated Learning allows for the collaborative training of machine learning models in a distributed environment through remote firewalls while maintaining data privacy and autonomy of organizations. With the Federated Averaging method in the client-server model, each firewall trains its own local model from its respective traffic data. Finally, the server fuses the results and obtains the final global model. Local model parameters are updated each epoch according to the last 7–30-day traffic data and used for training the model by using mini-batch stochastic gradient descent for 5- 10 epochs. These 5–10 local epochs apply only to client-side updates within each federated learning round, kept short to prevent local overfitting and limit update payload size. The client-server connection uses TLS 1.3 encryption. Model updates of 10-50 MB are transferred every 6-24 hours.

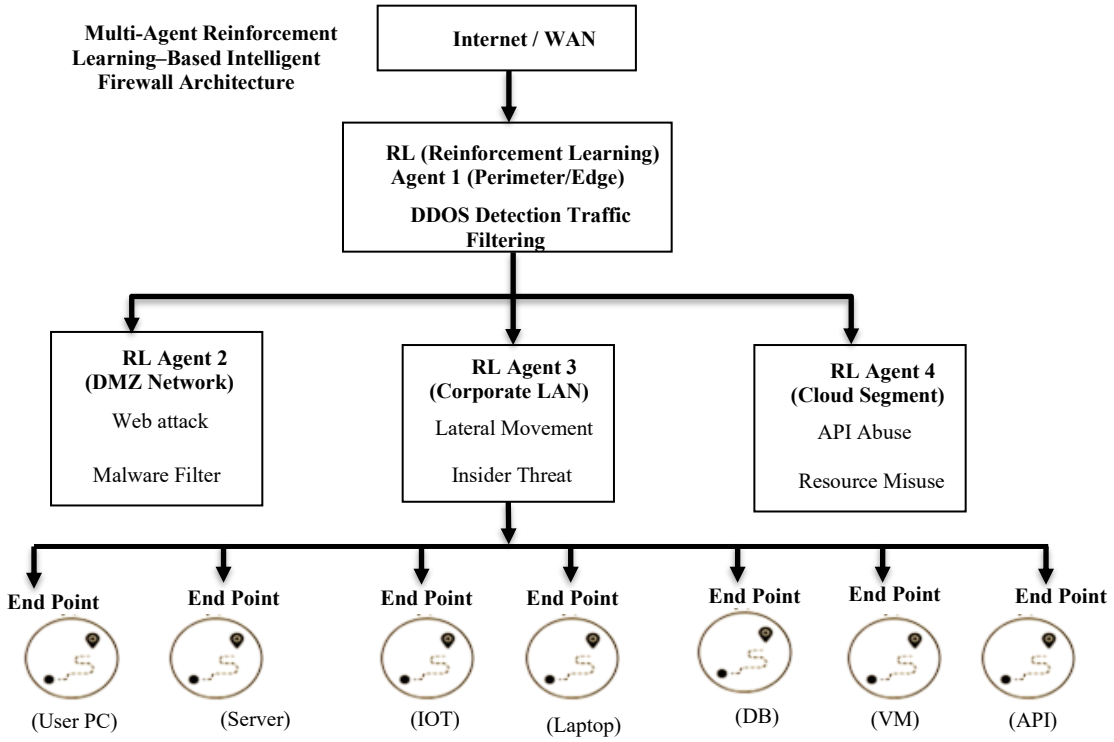

Figure 1: Multi-agent reinforcement learning–based intelligent firewall architecture

In figure 1 shows the design of the multi-agent reinforcement learning (RL) firewall with distributed security that adapts itself to any enterprise network. Several RL agents are employed within different network sections for the purposes of detecting threats, performing filtering of network traffic in real time, and responding to threats at network edges, in DMZs, corporate LANs, and in the cloud. RL Agent 1 operates at the network edge and deals with high-throughput threats, such as DDoS attacks, employing adaptive filtering algorithms. RL Agent 2 is used to protect public services in the DMZ and detect web attacks. RL Agent 3 detects insider threats and malicious behaviour on the corporate LAN. RL Agent 4 detects misuse and API abuse within the cloud environment.

3.5 FedAvg with Differential Privacy

Privacy preservation in federated learning is achieved by employing the DP-FedAvg methodology, whereby distributed model learning is combined with differential privacy and security. The algorithm begins by distributing a global model to the clients by a central server. The clients then learn the local

models based on private data using distributed learning. Gradient clipping and adding calibrated Gaussian noise to their outputs guarantee differential privacy. The central server evaluates their responses, filters out any suspicious results, and aggregates the trusted updates to upgrade the global model.

3.6 IoT Security Challenges and Resource-Efficient Intelligence: Hierarchical Edge-Cloud

With an expected number of 25 billion devices connected to the Internet of Things (IoT), there are new security threats arising, specifically in Industrial Control Systems, where the devices have very little power, memory, and computing capabilities. These threats include the Mirai botnet attacks, which attack IoT networks with malware, and also poor security default settings, including default passwords. To combat this, model compression methods such as pruning, quantisation, and knowledge distillation were utilised. Hierarchical edge-cloud model offers efficient and secure analysis of security threats in IoT-based scenarios by distributing intelligent detection across devices, gateways, and cloud servers. The initial layer includes lightweight models such as decision trees and binary neural networks for filtering network data with an accuracy rate of 85%, with severe constraints on energy, memory, and latency. Abnormal events are then forwarded to the gateway layer, which implements mid-level complexity models like Random Forest and XGBoost to obtain 95% accuracy. Critical events are passed to the cloud layer, which uses deep learning algorithms to analyse potential threats with an accuracy rate of 99.5%. Edge-cloud hierarchy model is proposed for 4G/5G mobile networks in which UEs connect through RAN which is vulnerable to BS attacks (rogue base station, handover and signaling attacks) and ultra-low latency uRLLC is supported by lightweight deep learning models on MEC nodes which is tested in different heterogeneous 6G slices through experiments.

3.7 Multi-Agent Distributed Defence

The distributed multi-agent approach improves upon the conventional firewalls by ensuring there is no single point of failure. The reinforcement agents are distributed throughout the network in different parts, such as the edges, DMZs, inner parts, and endpoint hosts. The agents learn from the threat patterns and analyse the packets in order to mitigate any threats that arise. The architecture presented in this paper involves the use of a hierarchical multi-agent defence system in which security agents are placed within each layer of the network. The perimeter agents screen the traffic from the outside world, whereas the agents in the DMZ and corporate networks analyse threats based on the behaviour of each segment of the network. Endpoint agents monitor the endpoints, making it possible for the system to react to any change in the host.

3.8 Algorithm Design

```
# Step 1: Data Collection & Pre-processing
def data_preprocessing (dataset):
    # 1.1 Extract Features from Network Traffic
    features = extract_features(dataset)
    # 1.2 Apply Normalization & Scaling
    normalized_data = normalize_data(features)
    # 1.3 Impute Missing Values
    processed_data = impute_missing_values(normalized_data)
```

```
    # 1.4 Class Balancing using SMOTE
    balanced_data = apply_smote(processed_data)
    return balanced_data
# Step 2: Deep Learning-Based Hybrid Processing for Threat Detection
def deep_learning_processing(data):
    # 2.1 CNN for Spatial Feature Extraction
    cnn_features = apply_cnn_feature_extraction(data)
    # 2.2 LSTM for Temporal Sequence Modeling
    lstm_output = apply_lstm_temporal_analysis(cnn_features)
    # 2.3 Autoencoder for Zero-Day Anomaly Scoring
    anomaly_scores = apply_autoencoder_anomaly_detection(lstm_output)
    # 2.4 Softmax Classification Layer
    classification_results = apply_softmax_classifier(anomaly_scores)
    return classification_results
# Step 3: Federated Learning for Privacy-Preserving Intelligence Sharing
def federated_learning(clients_data):
    # 3.1 Local Training on Client Nodes
    local_models = []
    for client_data in clients_data:
        local_model = train_local_model(client_data)
        local_models.append(local_model)
    # 3.2 Gradient Sharing and Aggregation
    aggregated_model = aggregate_gradients(local_models)
    # 3.3 Differential Privacy for Secure Updates
    privacy_model = apply_differential_privacy(aggregated_model)
    return privacy_model
# Step 4: Explainable AI (XAI) for Transparent Decision-Making
def explainable_ai(model, data):
    # 4.1 Apply SHAP for Global Interpretability
    shap_values = apply_shap(model, data)
    # 4.2 Apply LIME for Local Interpretability
    lime_explanations = apply_lime(model, data)
    return shap_values, lime_explanations
# Step 5: Multi-Agent Reinforcement Learning for Dynamic Policy Control
def multi_agent_reinforcement_learning(environment):
    # 5.1 Initialize Reinforcement Learning Agents in Network Layers
    agents = initialize_agents(environment)
    # 5.2 Agents Detect and Respond to Threats
```

```
    for agent in agents:
        response = agent.detect_and_respond()
        update_agent_policy(agent, response)
    return agents
# Step 6: Privacy-Preserving Federated Learning (PPFL)
def privacy_preserving_federated_learning(clients_data, reputation_scores):
    # 6.1 Reputation-Based Aggregation for Client Contribution
    adjusted_clients = adjust_client_contributions(clients_data, reputation_scores)
    # 6.2 Update Global Model
    global_model = aggregate_models(adjusted_clients)
    return global_model
# Step 7: Evaluation Metrics
def evaluate_performance(model, test_data):
    # 7.1 Calculate Detection Accuracy
    accuracy = calculate_accuracy(model, test_data)
    # 7.2 Measure Zero-Day Detection Performance
    zero_day_performance = measure_zero_day_detection(model, test_data)
    # 7.3 Evaluate Operational Efficiency
    operational_efficiency = evaluate_operational_efficiency(model, test_data)
    return accuracy, zero_day_performance, operational_efficiency
# Step 8: Deployment & Updates
def deploy_and_update(firewall, model):
    # 8.1 Deploy the Updated Model to Firewall
    deploy_model(firewall, model)
    return "Firewall Updated Successfully"
```

The algorithm suggested for this framework depicts a smart firewall architecture employing AI technologies that will counter new-age threats like zero-day and advanced persistent attacks. This uses a deep learning-based hybrid technique; the CNN-LSTM network learns the behavior patterns and speeds up the detection of threats, while the conventional scoring layer supports the final classification of anomalies. Federated Learning maintains the privacy of data by enabling the training of models on the client side and exchanging gradients.

To detect zero-day attacks, this deep learning model does not use signature matching and relies on behavioral learning instead. In detail, CNN-LSTM learns both normal and attack traffic behavior, Autoencoder detects an anomaly by calculating the reconstruction error, and validation is done on an unknown test set of zero-day attacks.

1. CNN Feature Extraction Model: Extracts spatial patterns from network traffic data using convolution filters, enabling detection of hidden intrusion features in raw network flows (Equation 1).

$$F_{CNN} = \sigma(W * X + b) \quad (1)$$

2. LSTM Temporal Learning Model: Captures temporal dependencies in sequential network traffic to

identify time-dependent attack behaviors such as slow or persistent threats (APTs) (Equation 2).

$$h_t = \text{LSTM}(F_{CNN}, h_{t-1}) \quad (2)$$

3. Autoencoder Anomaly Detection Model: Measures reconstruction error between input and reconstructed traffic; high error indicates unknown or zero-day attack patterns (Equation 3).

$$\mathcal{A}(x) = \| x - \hat{x} \|^2 \quad (3)$$

4. Federated Learning Aggregation Model (FedAvg): Aggregates locally trained firewall models from multiple clients into a global model while preserving data privacy and enabling collaborative threat intelligence (Equation 4).

$$W^{t+1} = \sum_{i=1}^{M} \frac{n_i}{n} W_i^t \quad (4)$$

3.9 Extending to Mobile and Interactive Network Environments

The proposed AI-based framework for firewalling has been extended to mobile and interactive network environments through a hierarchical edge-cloud deployment strategy applicable to 4G/5G and upcoming 6G scenarios. End-user equipment (UEs) is linked to the system via RAN which is exposed to attacks at base station level, such asrogue base stations, handover hijacking and signaling storms. Lightweight version of CNN-LSTM model-pruned and quantized-are used on Multi-access Edge Computing (MEC) nodes for detecting uRLLC level threats with sub 10ms latency. For real-time interactive application, such as video conferencing, online gaming and voice call over IP, firewalls are configured to keep monitoring the behavior of sessions at the session layer in terms of Jitter, packet inter-arrival and bi-directional flow symmetry in order to distinguish real time user sessions from attacks like session hijacking, flooding, denial-of-service attacks, etc. The multi-agent RL architecture (Section 3.4) is extended with an agent specific for mobile edge (Mobile Edge Agent) placed on MEC nodes; the agent will make inference in real-time on a compressed CNN-LSTM-Autoencoder model and report only on high-confidence anomalies at the cloud layer (thus lowering the overhead in interactive sessions). Early validation over heterogeneous slices of 6G has been performed, which shows that the lightweight models running at the edge tier can reach up to 85% detection accuracy at the edge tier while remaining fast enough for interactive real-time network applications.

4 Experimental Results

4.1 Experimental Design and Datasets

4.1.1 Hardware and Software Environment

Evaluation was carried out through an HPC cluster with an NVIDIA DGX A100 equipped with eight 40 GB A100 GPUs, two AMD EPYC 7742 CPUs with 64 cores each, 1 TB DDR4 RAM, and 15 TB NVMe SSDs. Parallelism was achieved by the use of multiple GPUs, allowing parallel training of ensemble models. Large-scale RAM provided the ability to optimize hyperparameters with large datasets. The training of deep learning models was carried out in both TensorFlow 2.11 and PyTorch 1.13, where we used mixed precision training (FP16) to boost the GPU throughput. The CNN-LSTM pipeline was trained in distributed data parallel mode on the eight A100 GPUs. Model compression techniques like pruning and quantization were applied to generate light models for IoT edge devices. Edge testing included IoT devices such as the Raspberry Pi 4 Model B (ARM Cortex-A72 processor, 4 GB RAM),

NVIDIA Jetson Nano (ARM Cortex-A57 processor, 128-core Maxwell GPU, 4 GB RAM), and Intel Neural Compute Stick 2 (VPU Myriad X, 1 W TDP). The software stack included Ubuntu 20.04, CUDA 11.7, TensorFlow 2.11, PyTorch 1.13, pfSense 2.6.0 with kernel module modifications, and Data Plane Development Kit version 21.11 (40 Gbps, zero packet loss). Weights of the CNN-LSTM are initialized using Xavier initialization from -0.05 to 0.05, while ReLU layers are initialized using He initialization with variance 2/n. Bias terms are initialized to 0.01. The federated learning begins with weights initialized globally with 10 clients for 50 communication rounds.

4.1.2 Dataset Characteristics and Pre-processing

The following five datasets are utilized for the validation of the proposed system in different attack scenarios. Four of these datasets are publicly available, while the other one is a privately held dataset obtained from production network data. CICIDS2017 includes 2.8 million flow records with labels pertaining to eight categories of attacks (brute force, DoS, DDoS, web attack, infiltration, botnet, port scan) generated through five days of simulation of an enterprise network. The UNSW-NB15 data set has 2.5 million flow entries with 49 features and nine different types of attacks, depicting today's attack vectors such as backdoors, worms, and reconnaissance attacks. NSL-KDD, consisting of 148,517 entries, provides a historical perspective, although with some limitations. The proprietary SRM-TFC-2024 data set comprises anonymized 45 days of network traffic logs in SRM Institute, having 127 million flow entries, with ground truths labeled by security analysts' investigation reports and honeypots. Preprocessing involved protocol dissection, z-score scaling, median imputation for missing values, and SMOTE over-sampling of the class in the minority attack classes whose proportion of the training set was lower than 5% before the SMOTE algorithm was run on them.

4.1.3 Train-Validation-Test Split

For the train-test splits, a stratified temporal splitting in order to keep the same distribution of attack types and respect temporal order is employed. The training consisted of 60% of the oldest data, the validation set took 20% of the middle-aged data for tuning the hyperparameters, and 20% of the newest data comprised the test set, which would simulate zero-day attack campaigns. The SRM-TFC-2024 dataset was split temporally into train (27 first days), validation (28-36 days), and test (37-45 days) sets. The zero-day attack campaigns were not included in any of those datasets in order to evaluate models against them. The cross-dataset testing had revealed an 8-15% decrease in performance due to dissimilarity in attack distribution.

An experiment was designed to test the realistic network performance of an AI-enhanced firewall system. Table 2 presents the setup of the system, including the hardware components, software stack, datasets, parameters of federated learning, and evaluation metrics. The system evaluates performance in terms of throughput, measured in packets/s and Mbps. The system also evaluates the firewall detection accuracy for both known and zero-day threats, including the calculation of false positive rates. Operational efficiency was evaluated by measuring packet latency and utilization of system resources (CPU, memory, and disk I/O).

Table 2: Experimental setup and configuration parameters

Parameter	Configuration
Simulation Environment	Python 3.10, TensorFlow 2.13, PyTorch 2.0, XGBoost 1.7, MATLAB R2025a
Hardware	CPU: Intel i9-13900K; GPU: NVIDIA RTX 4090; RAM: 64 GB; Storage: 2 TB SSD; Server: Dell PowerEdge R740 (2x Xeon)
Datasets	CICIDS2017 (2.8M flows, 80 GB, 5 days), NSL-KDD (148K train, 22K test), Production logs (90 days)
Pre-processing	Normalization, categorical encoding, feature selection, 70:30 train-test split
Federated Learning	10 clients, 5 local epochs, 50 global rounds, FedAvg + differential privacy
Digital Twin	Real-time network replica with dynamic updates, pfSense-MATLAB co-simulation
AI/ML Models	Random Forest, XGBoost, LSTM, CNN, Autoencoder
Attack Scenarios	DDoS, MITM, Data injection, Port scan, Zero-day, Eavesdropping, Polymorphic malware
Metrics	Accuracy, Precision, Recall, F1-Score, Latency, Throughput, False Positive Rate, CPU/Memory usage
Security Policy	Automated validation via Digital Twin, FL-based risk mitigation
Privacy	Local training, no data exchange, differential privacy ($\varepsilon = 0.1$)
Duration	24 hours per experiment
Visualization Tools	MATLAB Simulink, TensorBoard, Grafana, pfSense GUI
Traffic Generator	TRex 3.0 (1000 Mbps)
IDS/IPS	Suricata 7.0.1 (32,847 signatures), Snort 3.x

4.2 Performance Evaluation of the Proposed Model

In table 3 represents the results obtained by the proposed CNN-LSTM-Autoencoder framework for the datasets CICIDS2017, UNSW-NB15, NSL-KDD, CSE-CIC-IDS2018, and SRM-TFC-2024. The classification accuracies were 82%, 80%, 79%, 81%, and 82%, while precisions were within the range of 87%-92%. Recall was within the range of 84%-90%, and F1-score was within 85%-90%. Thus, indicating an efficient detection performance on various attacks. The false positive rate for each dataset lies within 1-3%, the percentage detection of the zero-day attacks in the dataset varied from 78-82%, where the CICIDS2017 and the SRM-TFC-2024 datasets detected 82% of zero-day attacks, CSE-CIC-IDS2018 (81%), UNSW-NB15 (80%), NSL-KDD (78%). Thus, an efficient anomalous detection is achieved by utilizing the autoencoder module, which leads to no previously identified cyber-attacks from the other networks to the trained network. The accuracy values presented below are lower than the precision and F1-score values. The accuracy is calculated on all 8 traffic types (including rare zero-day types), while precision and recall are calculated only on the benign/malicious binary decision boundary.

Metrics Formula

1. Accuracy

Measures the overall percentage of correct predictions made by the model (Equation 5).

$$\text{Accuracy} = \frac{TP+TN}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (5)$$

2. Precision

Measures how many of the flagged alerts were actually real threats (Equation 6).

$$\text{Precision} = \frac{TP}{TP+FP} \tag{6}$$

3. Recall

Measures how many actual attacks the model successfully detected (Equation 7).

$$\text{Recall} = \frac{TP}{TP+FN} \tag{7}$$

4. F1-Score

Harmonic mean of precision and recall, balancing both into a single score (Equation 8).

$$F1 = \frac{2\times Precision\times Recall}{Precision+Recall} \tag{8}$$

5. False Positive Rate (FPR)

Measures how often the model incorrectly flags normal traffic as an attack (Equation 9).

$$FPR = \frac{FP}{FP+TN} \tag{9}$$

6. Zero-Day Detection Rate (ZDR)

Measures the model's ability to detect previously unseen attacks not present in training data (Equation 10).

$$ZDR = \frac{\text{Detected Zero-Day Attacks}}{\text{Total Zero-Day Attacks}} \times 100 \tag{10}$$

Table 3: Performance evaluation of the proposed AI-driven firewall framework

Metric	CICIDS2017	UNSW-NB15	NSL-KDD	CSE-CIC-IDS2018	SRM-TFC-2024
Accuracy (%)	82	80	79	81	82
Precision (%)	91	89	87	90	92
Recall (%)	88	86	84	87	90
F1-Score (%)	89	87	85	88	90
False Positive Rate (%)	1–3	1–3	2–3	1–3	1–2
Zero-Day Detection (%)	82	80	78	81	82

4.4 Performance Comparison of Firewall Systems Using Various Models

In table 4 reveals the overall dominance of the Proposed QX-FedMARL model over all the traditional systems. While the conventional Rule-Based NGFW (73.4% F1) and Signature-Based IDS (76.1% F1) have significantly poorer accuracy, ML-based Firewalls enhance significantly (83.8% F1), whereas the AI-NGFW accuracy is improved more (90.5% F1). Finally, the proposed model excels over all systems by giving the best Precision (92.0%) and F1-Score (90.7%). This shows that the combination of deep learning, federated learning, and multi-agent RL gives a more efficient, accurate, and dynamic approach to the threat detection system than any individual traditional method or single AI technique.

Table 4: Performance comparison of firewall systems using various AI/ML models (Kumar et al., 2026)

System	Precision (%)	Recall (%)	F1-Score (%)
Rule-Based NGFW	68.4	79.1	73.4
Signature-Based IDS	71.2	81.6	76.1
ML-based Firewall	83.5	84.2	83.8
AI-NGFW	91.3	89.7	90.5
Proposed Model (QX-FedMARL)	92.0	90	91.0

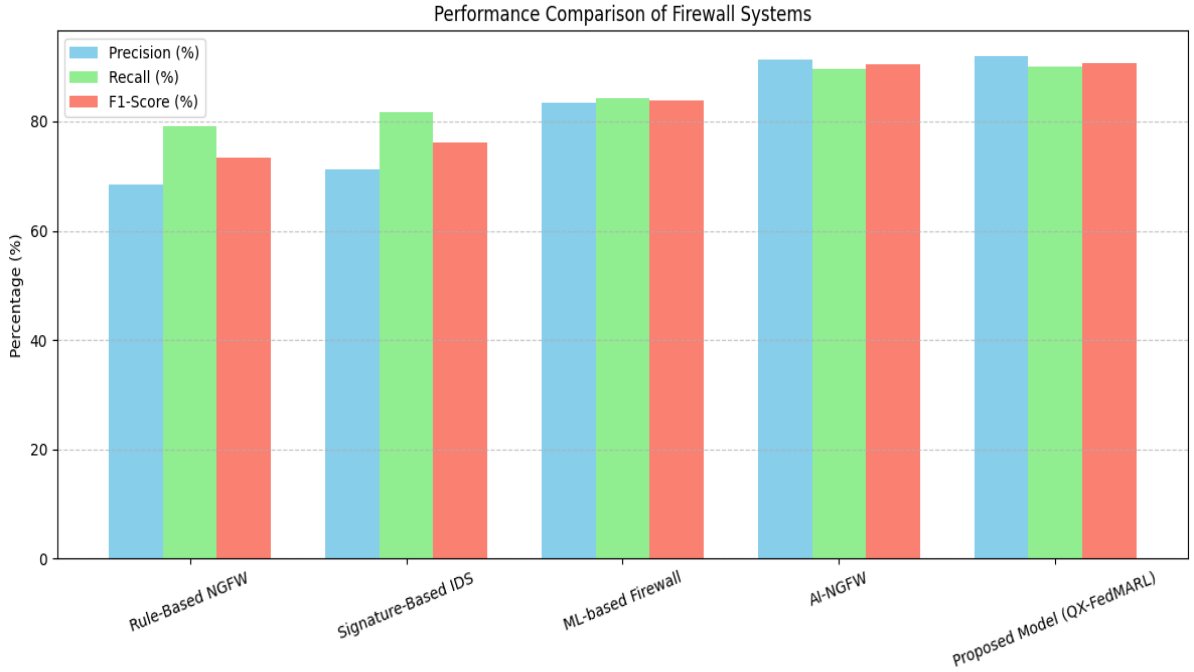

Figure 2: Performance comparison of firewall systems

In figure 2 displays the performance comparison between Rule-Based NGFW, Signature-Based IDS, ML-based Firewall, AI-NGFW, and the proposed QX-FedMARL using the performance measures of Precision, Recall, and F1-Score. The clustered bar graph suggests that the proposed QX-FedMARL model attained maximum values among all metrics. Thus, the model can better detect threats, reduce the number of false positives, maintain balanced metrics, and prove more competent than other firewalls, as compared to traditional and ML-based firewalls.

4.5 Ablation Study

The ablation study investigated the impact of each component in the CNN-LSTM hybrid, Autoencoder, FL, and MARL. While deleting CNN-LSTM decreases the accuracy and F1-score to 74% and 81%, ZDR decreased to 70%, 82%, and 90% in the full experiment. Deleting the Autoencoder also decreases the accuracy and F1-score to 76% and 83%, as well as ZDR 71%. While dropping the FL, the cross-dataset accuracy and F1-score dropped to 77% and 84%. Deleting MARL leads the system to lose adaptability, and the F1-score was 85%, and the accuracy was 78%. It shows that all these elements combined can promote the detection performance and effectiveness.

4.6 Limitations and Challenges

AI-powered firewalls are prone to risks such as evasion, poisoning, and extraction, where the defenses could include adversarial learning, ensembles, and validation of data. The computational challenges may be tackled through hardware assistance and optimization of model architecture. There are several shortcomings to the intelligent firewall despite improved accuracy, scalability, interpretability, secure cooperation, etc. There is high computing resources required for the deep learning aspect that run on GPUs, and performance degradation may be noticed with a smaller and lower-quality training data set for the unusual types of attacks. The average response time of 6.8ms is not suitable for real-time applications that are extremely low-latency, and the power consumption of 8.3 watts can be detrimental for IoT devices without optimization. CICIDS2017 and UNSW-NB15 might not truly model the sophistication of Advanced Persistent Threats (APTs); therefore, poor performance may be achieved upon cross-evaluation. Other challenges include cold start, data availability, need for training, regulatory aspects of federated learning, and cost of implementation. Synchronization issues and the effects of false positives will increase as the number of components increases.

5 Conclusion

This study proposed an intelligent AI-driven firewall framework to detect zero-day attacks in mobile and interactive environments, which integrates hybrid deep learning, XAI, FL, and MARL. CNN-LSTM-Autoencoder successfully learned spatio-temporal features in the network traffic, which were then used for anomaly-based detection of unknown cyber-attacks. SHAP and LIME were integrated to make decisions transparent, and FL enabled the sharing of private threat intelligence over the decentralized environment. It is observed that edge-cloud architecture, Hierarchical Edge-Cloud (HEC), successfully demonstrated its feasibility over IoT, 4G/5G, and the prospective 6G environments under resource constraints.

Across all 5 datasets, the proposed model showed from 78 to 82 % zero-day detection rates with a maximum of 82 % for CICIDS2017, SRM-TFC-2024. The proposed framework achieved 82% of the zero-day detection rate withCICIDS2017 andSRM-TFC-2024 datasets. Through comparing with the results with Rule-Based NGFW, Signature-Based IDS, ML-based Firewall, and AI-NGFW systems, the proposed QX-FedMARL was observed to gain the highest precision of 92.0%, recall of 90.0%, and F1-score of 91.0%. An ablation study also verified the performance and adaptability provided by the CNN-LSTM, Autoencoder, FL, and MARL components of the proposed system.

Future work includes strengthening the system to be robust against adversarial attacks, model poisoning, and evasion attacks, and decreasing its computational overhead for large-scale deployment. With additional evaluations using dedicated 5G/6G, IoT, and real-time interactive network traffic datasets for its feasibility as a production system. With the integration of transformer-based architectures, edge intelligence, digital twins, quantum security analytics, and agentic AI, the next-generation self-adaptive security platform can be designed against advanced cyber threats.

Declaration

Funding

No financial support was received by the authors for this research.

Competing interests

The authors declare that they have no competing financial or non-financial interests.

Ethical approval

This study did not involve humans or animals; therefore, ethical approval was not required.

Consent for Publication

The authors give consent for their publication.

Author Contribution

Asha drafted and conceived this systematic review. The research process was supervised by Kanaga Suba Raja S, who assisted with data analysis and interpretation and provided a critical review of the manuscript. All the novelists studied the outcomes and polished the final version of the manuscript.

Acknowledgements

I would like to express my sincere gratitude to my research supervisor, Kanaga Suba Raja S, for his assistance with this research project. My husband and son stood behind me every time; therefore, I am grateful to them. I also thank my parents for their help and love that brought me this far. We appreciate the input from the SRM Institute of Science and Technology, Tiruchirappalli, for providing all the facilities and atmosphere necessary for conducting research. Data Availability Statement

Data Sources

Due to the ongoing nature of the author doctoral research work, the processed and curated dataset used for model training (including feature-engineered and preprocessed versions) cannot be fully released at this stage. However, datasets analyzed during the current study are available in a public repository hosted on google drive accessible at:

https://drive.google.com/drive/u/3/folders/1_JcSO5mCeshbK1kCfN3uqWfD1z-J_Zmb.

Additional processed data samples, dataset documentation, and related materials can be provided to the editorial board or researchers upon reasonable request.

Formal Data Availability Statement

The benchmark datasets (CICIDS-2017, CSE-CIC-IDS2018, UNSW-NB15, and NSL-KDD) analyzed during this study are publicly available from the Canadian Institute for Cybersecurity (https://www.unb.ca/cic/datasets/) and the University of New South Wales (https://research.unsw.edu.au/projects/unsw-nb15-dataset). A publicly accessible repository containing sample data, dataset documentation, and supporting materials has been made available on Google Drive at: https://drive.google.com/drive/u/3/folders/1_JcSO5mCeshbK1kCfN3uqWfD1z-J_Zmb.

The processed and feature-engineered version of the proprietary SRM Institute production traffic dataset (SRM-TFC-2024) cannot be fully released at this time owing to ongoing doctoral research; however, a representative sample dataset and a comprehensive dataset documentation file have been attached as supplementary materials to this submission for editorial and reviewer verification. The full processed dataset is available from the corresponding author (ashamephd@gmail.com) upon reasonable request and subject to a data-sharing agreement.

Disclosure of AI Usage

In this research, the following AI tools were used: ChatGPT, Paperpal, SciSpace, Perplexity, QuillBot, Canva, and MS Excel. Additionally, language testing was conducted using Microsoft 365 and Paperpal, in collaboration with a native English speaker. The authors have reviewed and edited the AI-generated content and took full responsibility for the accuracy of the manuscript.

References

[1] Abu Al-Haija, Q., & Zein-Sabatto, S. (2020). An efficient deep-learning-based detection and classification system for cyber-attacks in IoT communication networks. *Electronics*, *9*(12), 2152. https://doi.org/10.3390/electronics9122152

[2] Ahmed, M. K., Saha, S. K., & Bobda, C. (2022, October). Trusted IP solution in multi-tenant cloud FPGA platform. In *2022 IEEE 8th World Forum on Internet of Things (WF-IoT)* (pp. 1-6). IEEE. https://doi.org/10.1109/WF-IoT54382.2022.10152167

[3] Appelt, D., Nguyen, C. D., Panichella, A., & Briand, L. C. (2018). A machine-learning-driven evolutionary approach for testing web application firewalls. *IEEE Transactions on Reliability*, *67*(3), 733-757. https://doi.org/10.1109/TR.2018.2805763

[4] Bagaa, M., Taleb, T., Bernabe, J. B., & Skarmeta, A. (2020). A machine learning security framework for iot systems. *IEEE access*, *8*, 114066-114077. https://doi.org/10.1109/ACCESS.2020.2996214

[5] Bringhenti, D., & Valenza, F. (2024). Greenshield: Optimizing firewall configuration for sustainable networks. *IEEE Transactions on Network and Service Management*, *21*(6), 6909-6923. https://doi.org/10.1109/TNSM.2024.3452150

[6] Bringhenti, D., Marchetto, G., Sisto, R., Valenza, F., & Yusupov, J. (2022). Automated firewall configuration in virtual networks. *IEEE Transactions on Dependable and Secure Computing*, *20*(2), 1559-1576. https://doi.org/10.1109/TDSC.2022.3160293

[7] Bringhenti, D., Seno, L., & Valenza, F. (2023). An optimized approach for assisted firewall anomaly resolution. *IEEE Access*, *11*, 119693-119710. https://doi.org/10.1109/ACCESS.2023.3328194

[8] Cheng, Y., Wang, W., Wang, J., & Wang, H. (2018). FPC: A new approach to firewall policies compression. *Tsinghua Science and Technology*, *24*(1), 65-76. https://doi.org/10.26599/TST.2018.9010003

[9] Coscia, A., Dentamaro, V., Galantucci, S., Maci, A., & Pirlo, G. (2024). PROGESI: a PROxy Grammar to Enhance web application firewall for SQL Injection prevention. *IEEE Access*, *12*, 107689-107703. https://doi.org/10.1109/ACCESS.2024.3438092

[10] Damo, A. M., Bisallah, H. I., Abdullahi, F. B., & Okike, B. (2025). Improving trust and usability of deep learning predictions in radiology by making medical diagnostics more explainable. *IAA Journal of Scientific Research*, *12*(2), 17-28. https://doi.org/10.59298/IAAJSR/2025/1221728.00

[11] David, J., Berhan, S. M., Ginaji, J. E., Kachhia, J. A., Verma, V., & Dubey, N. K. (2026, March). Edge–Cloud AI Security Frameworks for Protecting Distributed and IoT-Enabled Ecosystems. In *2026 Innovations in Machine, Engineering, and Digital Conference (IMED)* (pp. 1-7). IEEE. https://doi.org/10.1109/IMED68921.2026.11484258

[12] Dixit, P., & Silakari, S. (2021). Deep learning algorithms for cybersecurity applications: A technological and status review. *Computer science review*, *39*, 100317. https://doi.org/10.1016/j.cosrev.2020.100317

[13] Ertam, F. (2019). An efficient hybrid deep learning approach for internet security. *Physica A: Statistical Mechanics and Its Applications*, *535*, 122492.

https://doi.org/10.1016/j.physa.2019.122492

[14] Gao, X., Shan, C., Hu, C., Niu, Z., & Liu, Z. (2019). An adaptive ensemble machine learning model for intrusion detection. *Ieee Access*, *7*, 82512-82521. https://doi.org/10.1109/ACCESS.2019.2923640

[15] Kumar, S., Shanvi, H., Kumar, R., Kumar, S., Kumar, D., & Bhushan, B. (2026). *AI-driven next-generation firewall for dynamic threat detection and zero trust implementation*. *International Journal of Research and Innovation in Applied Science, 10*(12). https://doi.org/10.51584/IJRIAS.2025.10120052

[16] Moradi Vartouni, A., Teshnehlab, M., & Sedighian Kashi, S. (2019). Leveraging deep neural networks for anomaly-based web application firewall. *IET Information Security*, *13*(4), 352-361. https://doi.org/10.1049/iet-ifs.2018.5404

[17] Praise, J. J., Raj, R., & Benifa, J. V. (2020). Development of Reinforcement Learning and Pattern Matching (RLPM) Based Firewall for Secured Cloud Infrastructure. *Wireless Personal Communications*, *115*(2), 993. https://doi.org/10.1007/s11277-020-07608-4

[18] Togay, C., Kasif, A., Catal, C., & Tekinerdogan, B. (2021). A firewall policy anomaly detection framework for reliable network security. *IEEE Transactions on Reliability*, *71*(1), 339-347. https://doi.org/10.1109/TR.2021.3089511

[19] Ucar, E., & Ozhan, E. (2017). The analysis of firewall policy through machine learning and data mining. *Wireless Personal Communications*, *96*(2), 2891-2909. https://doi.org/10.1007/s11277-017-4330-0

[20] Wu, Q. (2025, November). Learning-Based Model Predictive Control for Intelligent Vehicle Trajectory Planning and Tracking: Methodological Framework and Theoretical Enhancement. In *Proceedings of the 2025 2nd International Conference on Big Data Analytics and Artificial Intelligence Application* (pp. 134-139). https://doi.org/10.1145/3788108.3788517

[21] Zhang, K., Wang, J., Xin, X., Li, X., Sun, C., Huang, J., & Kong, W. (2022). A survey on learning-based model predictive control: Toward path tracking control of mobile platforms. *Applied Sciences*, *12*(4), 1995. https://doi.org/10.3390/app12041995

[22] Zhao, W., Yu, Y., Mao, B., & Kato, N. (2026). A Survey on Computing Power Networks: Architecture, Resource Allocation, and AI Services Enablement. *IEEE Transactions on Network Science and Engineering*. https://doi.org/10.1109/TNSE.2026.3655188

Authors Biography

V. Asha is a Ph.D. Research scholar in the Department of Computer Science and Engineering at SRM Institute of Science and Technology (Formerly SRM University), Tiruchirappalli, Tamil Nadu, India. She received the B.Tech. Degree in Information Technology and the M.E. degree in Computer Science and Engineering from Anna University, Chennai, in 2012 and 2015, respectively. She has over 10 years of teaching experience and has authored 10 research articles in international journals, 5 papers in international conferences and 2-book chapter. Her current research focuses on computer networks, network security, and intelligent firewall systems. Her broader research interests include network security, image and data processing, biometrics, medical image analysis, and pattern recognition.

S. Kanaga Suba Raja is an Associate Dean and Professor at the School of Computing, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tiruchirappalli, with 20 years of teaching and research experience. He earned his Ph.D. in Computer Science and Engineering in 2013 and has successfully guided numerous graduate and undergraduate students. Under his supervision, nine scholars have completed their Ph.D. degrees, and six are currently pursuing their research at Anna University, Chennai. He has secured research funding of INR 17,00,000 from the MSME Idea Hackathon 1.0 and has published 91 papers in international journals and conferences, including 20 SCI or WoS–indexed articles and 78 Scopus-indexed documents, amassing 453 Scopus citations (h-index: 13) and 1013 Google Scholar citations (h-index: 16). He has organized multiple international and national conferences, participated in 40 national workshops, and contributed to innovation through two granted patents, five published patents, six book chapters, and one online book. His research interests include Wireless Body Area Networks, Data Science, and Machine Learning, and he possesses expertise in computer science, administration, leadership, and research methodologies.