Array 31 (2026) 101081

Contents lists available at ScienceDirect

Array

journal homepage: www.sciencedirect.com/journal/array

An adaptive firewall framework using deep reinforcement learning for threat-aware cyber threat detection and mitigation policy learning

Mohammad Zahangir Alam [a,*], Sharmin Sultana [b]

[a] Department of Computing and Software Engineering, Faculty of Technology, University of Turku, Turku, Finland
[b] Department of Computer Science, Faculty of Science and Technology, American International University-Bangladesh (AIUB), Dhaka, Bangladesh

ARTICLE INFO

Keywords:
Cyber-attacks
Deep reinforcement learning
Adaptive firewall
Prioritized experience replay
Intrusion detection system
Network security

ABSTRACT

Rapid digital infrastructure growth has enabled sophisticated multi-vector cyberattacks that bypass traditional signature-based intrusion detection and static firewall configurations. Current defenses rely on fixed rule-based engines, signature-dependent classifiers, or threshold-based anomaly detectors, all of which struggle with concurrent threat patterns, zero-day attacks, and high-dimensional decision spaces. Machine and deep learning approaches for intrusion detection also face limitations, including slow adaptation to novel threats, high false positive rates, and limited support for dynamic firewall policy adjustment. To address these limitations, we propose an Adaptive Firewall Framework using Deep Reinforcement Learning (DRL) for threat-aware cyber threat detection and mitigation policy learning. The framework incorporates a policy-optimization DRL agent with a novel Threat-Aware Prioritized Experience Replay (TA-PER) mechanism, which prioritizes threat experiences according to operational severity, CVSS criticality, and buffer rarity. Through state-space modeling, threat-sensitive reward shaping, and incremental threat scoring, the DRL-TA-PER agent learns firewall response policies, including traffic blocking, isolation, adaptive rerouting, throttling, and benign traffic allowance. The framework is evaluated in an offline sequential setting using public benchmark datasets, where network-flow records are processed as ordered decision steps. Evaluation on CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023 shows that the proposed framework outperforms K-Nearest Neighbor, Random Forest, Logistic Regression, Support Vector Machine, Convolutional Neural Network, Long Short-Term Memory, standard DRL with uniform experience replay, Snort, and Suricata. The model achieved 96.8% accuracy, 97.2% recall, and an AUC-ROC of 0.981 across five trials, indicating that threat-aware replay prioritization can improve DRL-based firewall policy learning under offline benchmark evaluation conditions.

1. Introduction

The rapid proliferation of interconnected digital infrastructure across enterprise networks, cloud platforms, and distributed computing environments has significantly expanded the attack surface available to adversarial actors, rendering traditional perimeter-based security mechanisms increasingly inadequate against the scale and sophistication of modern cyber threats [1]. Cyberattacks such as Distributed Denial of Service (DDoS), port scanning, brute-force intrusion, man-in-the-middle exploitation, and advanced persistent threats have become routine components of coordinated malicious campaigns targeting critical network infrastructure, particularly in software-defined networks (SDN), cloud-native systems, and AI-driven enterprise environments [2]. In contrast, the classical firewall—designed around static rule sets, signature-based packet inspection, and manually configured access control lists—was developed for a far less dynamic threat landscape and is therefore ill-equipped to handle the polymorphic and multi-vector nature of contemporary attacks [3]. As a result, the limitations of traditional firewall architectures have become increasingly evident in both academic research and real-world deployments [4].

In this context, current network defense mechanisms continue to rely heavily on fixed rule-based engines, signature-dependent classifiers, or threshold-based anomaly detection techniques, all of which struggle to operate effectively in environments characterized by concurrent threat patterns, zero-day exploits, and high-dimensional decision spaces [5]. Rule-based systems such as Snort and Suricata remain effective for detecting known attack signatures; however, their dependence on predefined rules makes them inherently incapable of identifying previously

* Corresponding author.
E-mail addresses: mohammad.z.alam@utu.fi (M.Z. Alam), 22-48479-3@student.aiub.edu (S. Sultana).

https://doi.org/10.1016/j.array.2026.101081
Received 6 May 2026; Received in revised form 28 June 2026; Accepted 7 July 2026
Available online 16 July 2026

unseen threats without continuous manual updates, which are both time-consuming and operationally inefficient [6]. Similarly, supervised machine learning approaches—including Support Vector Machines (SVM), Random Forests, and K-Nearest Neighbor (KNN)—have demonstrated improved detection accuracy on benchmark datasets, yet they rely on static decision boundaries learned from historical data and therefore lack the adaptability required to respond to evolving threat patterns and adversarial traffic distributions [7]. This disconnects between controlled experimental performance and real-world deployment conditions continues to expose the limitations of static and reactive defense strategies [8].

In response, deep learning-based approaches such as Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, and hybrid architectures have been introduced to capture complex temporal and spatial patterns in network traffic [9,10]. While these models have advanced the state-of-the-art in detection accuracy, they remain constrained by high computational requirements, dependence on large volumes of labelled data, and limited ability to adapt to new attack patterns without retraining. More critically, these approaches are predominantly detection-oriented and provide limited support for adaptive mitigation policy learning under changing threat conditions. Consequently, existing systems fail to provide a unified and intelligent framework that integrates threat detection with adaptive response recommendation, which is essential for effective defense in dynamic network environments.

To move beyond these limitations, reinforcement learning has emerged as a promising paradigm for adaptive security policy learning in cybersecurity [11]. In particular, Deep Reinforcement Learning (DRL) allows an agent to learn defense strategies through sequential interaction with a modeled network environment, without depending entirely on predefined rules or static classifiers [11,12]. As illustrated in Fig. 1, the proposed framework operationalizes this capability through a unified Policy Decision Point (PDP) and Policy Enforcement Point (PEP) architecture. In this design, the Control Plane hosts the DRL Policy Engine and the proposed TA-PER mechanism, functioning as the decision-making entity that learns and optimizes firewall response policies from sequential network-flow states. The Data Plane represents how learned response policies may be translated into firewall actions such as traffic blocking, host isolation, adaptive rerouting, and selective allowance of benign traffic in a future deployment setting [12]. This separation of learning and enforcement enhances scalability and modularity while supporting policy adaptation to evolving and non-stationary threat conditions through an offline feedback and policy-update process used for sequential policy learning [11,12].

Despite these structural advantages, a critical limitation persists in existing DRL-based security frameworks — specifically in the way learning experiences are utilized during training [11,12]. As depicted in Fig. 2, the DRL agent operates within a continuous Agent-Environment interaction loop: at each timestep t, the agent observes the current network state st — comprising traffic features, threat alerts, system logs, and risk scores — selects a firewall action at, and receives a reward signal Rt reflecting the quality of that decision. The agent's policy $\pi\theta(s)$ is then updated via a reinforcement learning algorithm (DQN/PPO/Actor-Critic) to maximize long-term cumulative security

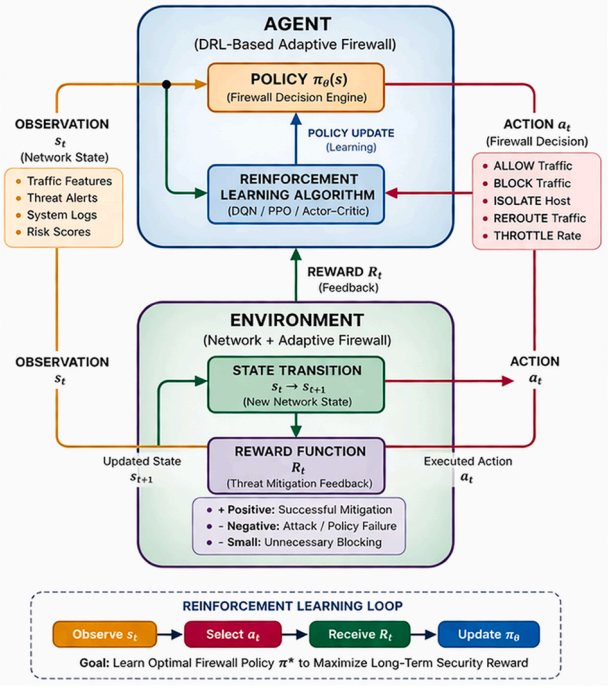

Fig. 2. Reinforcement learning interaction model for adaptive firewall policy optimization.

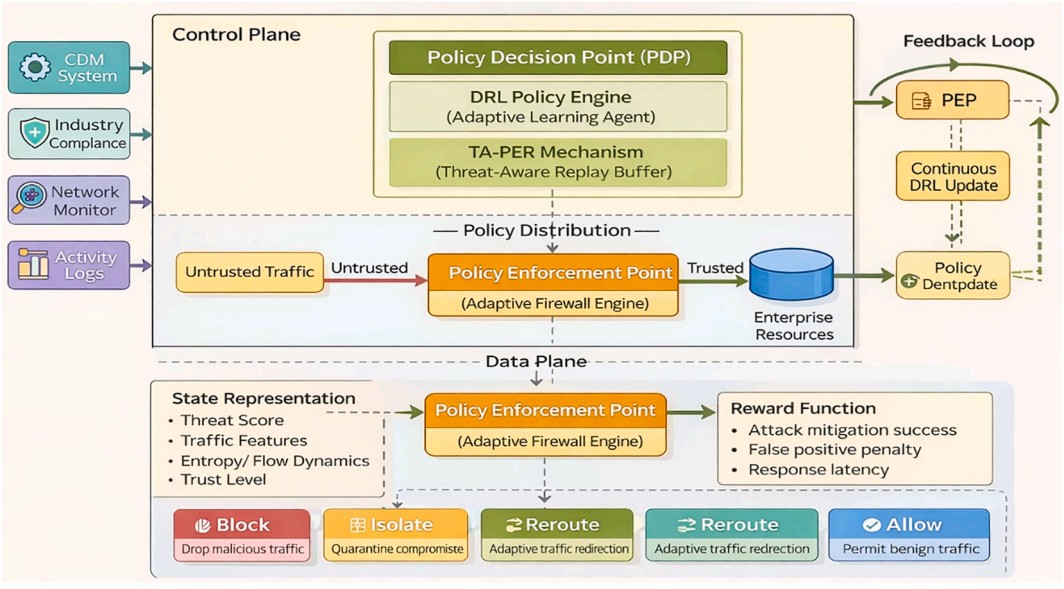

Fig. 1. Threat-aware cyber defense architecture: Control plane, DRL policy engine, and TA-PER

reward [11,12]. However, conventional experience replay mechanisms sample past interactions uniformly, assigning equal learning weight to routine benign traffic and rare yet high-impact attack events [13]. In practical network environments — where critical threats such as User-to-Root (U2R) and Remote-to-Local (R2L) intrusions are infrequent but operationally consequential — such uniform sampling produces biased policy learning and reduces the agent's effectiveness precisely where it matters most: against the rarest and most dangerous attack classes [13,14]. This observation motivates the need for a more intelligent, context-aware prioritization strategy embedded directly into the experience replay mechanism [13].

***Identified Gaps*:** Studies across the intrusion detection and network security literature have extensively explored rule-based systems, supervised machine learning models, and deep learning architectures for cyber threat detection. Traditional systems such as Snort and Suricata rely on signature-based detection mechanisms and predefined rule sets to identify known attack patterns [6]. While these approaches are effective against previously observed threats, their dependence on static rule repositories limits their capability to detect zero-day attacks and evolving multi-vector threat patterns [5], [6]. Similarly, supervised machine learning models including SVM, Random Forests, and KNN have demonstrated improved detection accuracy on benchmark datasets such as NSL-KDD and CICIDS-2017; however, these models operate on fixed decision boundaries learned from historical data distributions, making them less adaptive to dynamic and adversarial network environments [1], [7].

Recent studies have incorporated deep learning architectures such as CNNs, LSTMs, and hybrid CNN-LSTM models to capture temporal and spatial characteristics of network traffic. Although these models achieve competitive performance, they are associated with high computational complexity, dependence on large labelled datasets, and limited adaptability to novel threat patterns without retraining [9,10]. Furthermore, these approaches primarily focus on detection accuracy and remain fundamentally reactive, lacking the ability to support adaptive mitigation policy learning or dynamic firewall response recommendation—the capability represented through the PEP layer shown in Fig. 1 [8], [12].

Emerging research has investigated DRL for adaptive network security; however, as the agent-environment loop in Fig. 2 makes clear, the quality of policy learning is directly dependent on the quality of experience sampling. Existing frameworks largely rely on uniform experience replay that fails to account for the operational significance of rare, high-severity attack events—a gap that both figures jointly highlight: Fig. 1 shows what the system must decide, and Fig. 2 shows how the agent learns to decide it [13]. Without threat-aware prioritization in the replay buffer, the agent's policy may converge toward common-case performance at the expense of rare-case resilience.

In addition, existing network defense systems do not offer a unified framework that integrates intelligent threat detection with adaptive mitigation policy learning. The absence of threat-aware prioritization, adaptive policy learning, and integrated response-policy mechanisms highlights a critical gap in current research. In summary, the following are required.

- An adaptive firewall framework capable of dynamically learning and updating security policies in response to evolving and multi-vector cyber threats.
- A reinforcement learning mechanism incorporating threat-aware prioritization to improve learning effectiveness for rare and high-impact attack scenarios.
- An integrated system combining threat detection with adaptive mitigation policy learning and response recommendation without reliance on predefined rules or static classifiers.

1.1. Research contributions

We propose an Adaptive Firewall Framework based on Deep Reinforcement Learning (DRL) for threat-aware cyber threat detection and mitigation policy learning in distributed network environments. As shown in Fig. 1 and formalized through the learning process in Fig. 2, the framework integrates a policy-optimization DRL agent with a novel Threat-Aware Prioritized Experience Replay (TA-PER) mechanism. TA-PER prioritizes network threat experiences according to operational severity, CVSS criticality, and buffer rarity, allowing the agent to give greater learning attention to rare and high-severity attack patterns. Unlike conventional approaches that rely mainly on static rules or reactive detection models, the proposed framework learns adaptive firewall response policies from sequential network-flow records and is evaluated using CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023 benchmark datasets. The main contributions of this study are summarized as follows.

- A DRL-based adaptive firewall framework for threat-aware detection and mitigation policy learning, realized through the PDP-PEP-inspired architecture shown in Fig. 1.
- A novel Threat-Aware Prioritized Experience Replay (TA-PER) mechanism that augments TD-error priority with CVSS criticality scores and buffer rarity, improving learning attention to rare and high-severity attack scenarios and addressing the uniform sampling limitation identified in Fig. 2.
- Integration of threat-sensitive reward shaping and incremental threat scoring to balance detection accuracy, false positive reduction, response cost, and policy-learning stability.
- A sequential decision-making formulation in which firewall response actions, including traffic blocking, host isolation, adaptive rerouting, throttling, and benign traffic allowance, are learned as policy recommendations under offline benchmark evaluation conditions.
- A comprehensive experimental evaluation on CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023, demonstrating improved performance over KNN, Random Forest, LR, SVM, CNN, LSTM, standard DRL with uniform experience replay, Snort, and Suricata baselines. The proposed model achieved 96.8% detection accuracy across five independent trials and reduced the false positive rate compared with the strongest DRL baseline.

1.2. Structure of the article

The remainder of this paper is organized as follows. Section 2 reviews related work on rule-based intrusion detection, machine learning and deep learning approaches, and reinforcement learning-based network security. Section 3 presents the research motivation, including real-world cyberattack scenarios and the practical requirements for adaptive firewall policy learning. Section 4 describes the proposed DRL-TA-PER framework and methodology. Section 5 presents the experimental setup, evaluation metrics, baseline methods, and comparative results. Section 6 discusses the major findings, limitations, and future research directions. Section 7 concludes the paper.

2. Related work

2.1. Supervised learning approaches for intrusion detection

Machine learning-based approaches for intrusion detection and network threat mitigation have been extensively studied over the past decade [1]. Early intrusion detection systems primarily relied on supervised learning techniques, where models are trained on labelled datasets to classify network traffic as benign or malicious based on known attack signatures and behavioral patterns [7]. Among these, Support Vector Machines (SVM) and Random Forest (RF) have been widely adopted due to their strong classification performance, with SVM

demonstrating effective generalization in binary classification tasks and RF providing robust ensemble-based detection across multiple attack categories [7]. While these approaches achieve competitive accuracy on widely adopted benchmarks, their dependence on static decision boundaries and fixed feature representations limits their ability to adapt to evolving and previously unseen attack patterns without retraining [1]. Both SVM and RF are retained as baselines in the experimental evaluation of Section 4.4 to enable direct comparison with the proposed framework.

2.2. Rule-based firewall systems and their inherent limitations

In parallel, rule-based firewall systems such as Snort and Suricata remain widely deployed in real-world network defense environments [6]. These systems operate through signature matching against predefined rule sets and are effective in detecting known attack patterns. However, they are inherently reactive — detection occurs only after traffic matches an existing rule — and require continuous manual updates to remain effective against new threats [5]. Their inability to detect zero-day attacks and polymorphic threats, as illustrated through the motivating scenarios in Section 2.1, highlights the fundamental inadequacy of static signature-driven security mechanisms and directly motivates the adaptive learning paradigm adopted in this study [5], [6].

2.3. Deep learning techniques for enhanced threat detection

To address these challenges, deep learning techniques have been introduced to enhance feature representation and detection capability. Convolutional Neural Networks (CNNs) have demonstrated effectiveness in learning spatial traffic patterns, while recurrent architectures such as Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRU) capture temporal dependencies in sequential network data [9,10]. Hybrid approaches including autoencoder-based feature extraction combined with multilayer perceptron's have further reduced reliance on manual feature engineering while maintaining competitive detection accuracy [9]. Both CNN and LSTM are included as deep learning baselines in Section 4.4 to benchmark the proposed framework against state-of-the-art learned representations. Despite these advancements, deep learning-based IDS solutions remain computationally expensive, sensitive to class imbalance, and fundamentally constrained by their static learning paradigm — requiring full retraining to adapt to new threat patterns, a limitation directly addressed by the continuous policy update mechanism of the proposed DRL-TA-PER framework (Layer 8, Fig. 3) [1], [9].

2.4. Ensemble learning methods for detection robustness

Ensemble learning methods have been proposed to further improve detection robustness by combining multiple base models. These approaches have demonstrated high accuracy across diverse attack categories on standard benchmarks [7]. However, the increased computational overhead during inference poses significant challenges for real-time deployment in high-throughput network environments [8]. More critically, both single-model and ensemble-based approaches treat intrusion detection as a static classification problem, lacking the capability to autonomously determine and execute optimal mitigation actions in response to detected threats [11,12] — the unified detection-response gap identified in Section 2.2 that motivates the PDP-PEP architecture of the proposed framework (Fig. 1).

2.5. Deep reinforcement learning for adaptive network defense

Deep Reinforcement Learning has emerged as a promising paradigm for addressing these limitations by framing network security as a sequential decision-making problem, consistent with the MDP formulation presented in Section 3 (Eqs. (1) and (2)) [11]. In DRL-based frameworks, an agent interacts with the network environment, observes system states, and learns optimal mitigation policies through reward-driven feedback [12]. This enables adaptive and autonomous defense strategies without reliance on predefined rules or fully labelled datasets. Prior work has demonstrated the effectiveness of DRL in cybersecurity applications including zero-trust architectures and adaptive intrusion response systems. Nguyen and Reddi [11] highlighted the advantages of DRL in handling dynamic and adversarial environments, while He et al. [15] and Yang et al. [16] showed that DRL-based frameworks can outperform traditional machine learning models across multiple evaluation metrics. However, none of these frameworks incorporate threat-aware prioritization into the experience replay mechanism — the critical gap addressed by this study.

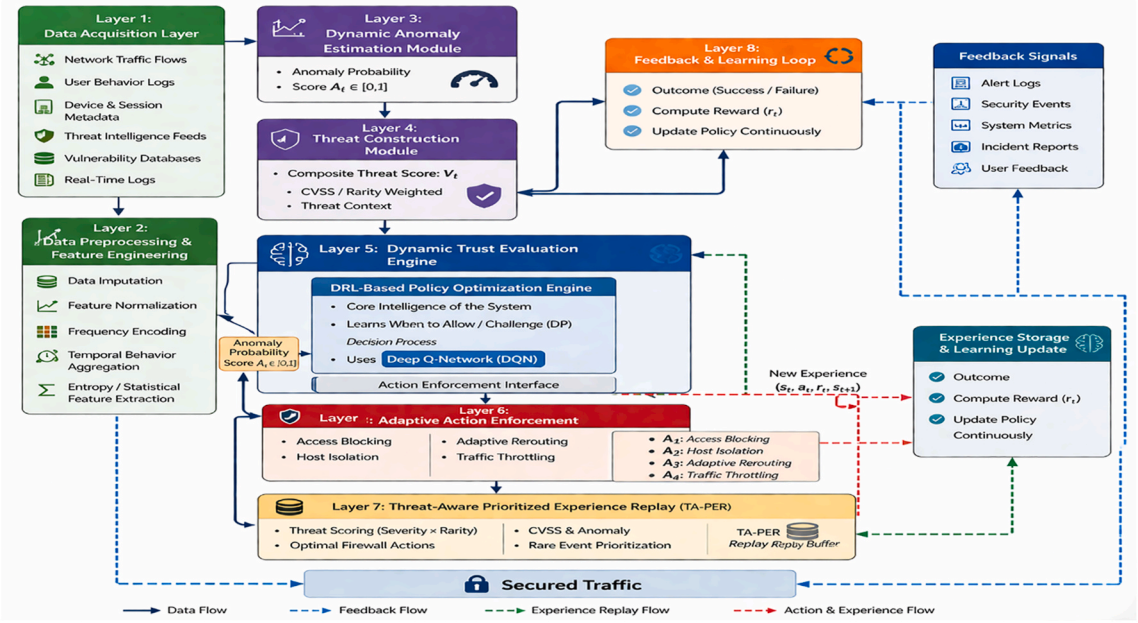

Fig. 3. Proposed framework for cyber threat detection and mitigation.

2.6. Limitations of uniform experience replay in security contexts

Despite these promising developments, a critical limitation persists across existing DRL-based security frameworks: the absence of threat-aware prioritization in the experience replay mechanism. Conventional DRL approaches rely on uniform sampling of past experiences, treating high-severity attack events and routine benign traffic with equal importance during policy optimization [13]. In real-world network environments — where rare but high-impact threats such as zero-day exploits, Shellcode attacks, Mirai botnet variants, and multi-vector DDoS campaigns are severely underrepresented — this uniform sampling leads to suboptimal Q-value estimates for critical attack classes and reduced policy sensitivity to the most operationally consequential threat scenarios [14]. This limitation is particularly evident across the four benchmark datasets adopted in this study (Section 4.1), where class imbalance between benign and attack traffic, and between common and rare attack subcategories, represents a fundamental challenge for uniform replay-based DRL agents.

2.7. Prioritized experience replay in security-oriented DRL

Prioritized Experience Replay (PER), introduced by Schaul et al. [21], partially addresses the limitation of uniform replay by assigning sampling priority based on temporal-difference error. However, standard PER does not incorporate domain-specific cybersecurity characteristics such as threat severity, CVSS-based criticality, or rarity of attack occurrences—attributes that are operationally important for security-oriented policy learning [17]. As a result, existing PER-based approaches may still underemphasize high-risk and low-frequency threat events during DRL optimization.

To address this limitation, this study introduces the Threat-Aware Prioritized Experience Replay (TA-PER) mechanism, which extends the original TD-error-based priority signal by incorporating CVSS-based severity and buffer rarity scores, as formalized in Eqs. (10) and (11) of Section 4.5. This enables the DRL agent to assign greater learning priority to operationally critical experiences during policy optimization, improving sensitivity to rare and high-severity attack patterns under offline sequential benchmark evaluation. The experimental validation of TA-PER against nine baseline systems across CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023 is presented in Section 5.5, with the comparative advantages of the proposed approach over reviewed methods summarized in Table 1.

3. Research motivation

While the preceding discussion highlights the theoretical limitations of existing network defense mechanisms, real-world cyberattack incidents further emphasize the urgency of developing adaptive and intelligent security solutions. Modern attack campaigns are no longer isolated or single vector in nature; instead, they involve coordinated, multi-stage strategies that exploit multiple system vulnerabilities to achieve large-scale disruption and data exfiltration [1], [2]. In such environments, defense mechanisms must not only detect threats accurately but also support adaptive response-policy learning under changing threat conditions. In this section, we present representative motivating scenarios and outline the practical challenges that underpin the need for an adaptive firewall framework.

3.1. Systems context and scenarios

One of the most prevalent patterns observed in contemporary cyber incidents is the use of credential compromise through phishing-based attacks [18]. In these scenarios, adversaries typically target a limited number of users with carefully crafted malicious emails designed to obtain login credentials. Once access is gained, compromised accounts are leveraged to propagate additional phishing campaigns within the organization, often targeting employees and customers. This cascading effect enables attackers to expand their foothold within the network, facilitating unauthorized access to sensitive systems and large-scale data exfiltration [18]. The iterative and self-propagating nature of such attacks makes early detection and containment critically important.

In parallel, web application exploitation continues to represent a major attack vector across enterprise and cloud-based systems [3], [19]. Attackers frequently exploit vulnerabilities such as SQL injection (SQLi) and cross-site scripting (XSS) to inject malicious payloads into web services [19]. These attacks may initially appear as isolated events but

Table 1
Advantages and limitations of existing network defense and intrusion detection models.

Author(s)	Proposed Model	Dataset Used	Advantages	Evaluation Metrics	Limitations
Mahida [15]	DRL-Zero Trust framework with dynamic trust evaluation	IEEE-CIS Fraud Detection	Continuous trust reassessment; adaptive policy without fixed rules; strong performance under adversarial conditions	Accuracy, Precision, Recall, F1, AUC-ROC	Evaluated on fraud detection domain; limited generalization to network intrusion
Guo et al. [12]	IZTSDN — Zero Trust with deep anomaly detection for SDN	Mininet emulation + synthetic benchmark	Integrates Zero Trust with anomaly detection; dynamic authorization; real-time monitoring	Detection accuracy, throughput under attack	Evaluated in emulation; limited large-scale real-world validation
Nguyen & Reddi [11]	Survey: DRL for cybersecurity	Multiple datasets	Comprehensive DRL taxonomy; insights on reward design and environment modeling	Not applicable (survey)	No experimental validation; highlights safe exploration challenges
Alanazi et al. [10]	Ensemble deep learning for DDoS mitigation in SDN	CICIDS2017	High detection accuracy; ensemble reduces variance	Accuracy, Precision, Recall, F1, AUC	High computational overhead; limited real-time deployment feasibility
Ramzan et al. [9]	RNN, LSTM, GRU for DDoS detection	CICIDS2017	Effective temporal modeling; GRU offers faster runtime than LSTM	Accuracy, Precision, Recall, F1, runtime	Focused on detection; lacks mitigation and control integration
Wei et al. [8]	AE-MLP: Autoencoder with MLP classifier	Public DDoS datasets	Reduces manual feature engineering; improved classification performance	Accuracy, Precision, Recall, F1, AUC	Pretraining overhead; performance depends on data quality
Becerra-Suarez et al. [7]	Comparative ML pipeline (RF, DT, AdaBoost, XGBoost, DNN)	CICIDS2019	Comprehensive comparison; robust preprocessing pipeline	Accuracy, Precision, Recall, F1, AUC	Single dataset; limited real-time and adaptive evaluation
Snort/ Suricata [6]	Rule-based signature IDS	Real network traffic	Low overhead; widely deployed; effective for known attacks	Detection rate, False Positive Rate	Cannot detect zero-day attacks; requires manual rule updates
Proposed DRL-TA-PER	DRL with Threat-Aware Prioritized Experience Replay	CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, CIC-IoT-2023	Threat-aware replay prioritization; adaptive mitigation policy learning; improved attention to rare and high-severity attack patterns	Accuracy, Precision, Recall, F1, AUC-ROC, FPR	

often evolve into multi-stage intrusions involving remote shell access, privilege escalation, and deployment of additional malicious components [18,19]. In many cases, compromised web applications are further used to deliver phishing interfaces or collect user data, thereby linking web-based exploitation with credential theft and broader network compromise [18]. These scenarios illustrate that modern cyber threats are highly interconnected, where different attack techniques reinforce one another within a single campaign [1], [18]. As a result, security systems must be capable of handling concurrent and multi-vector threats rather than treating attacks as independent events [2], [11].

3.2. Problem formulation and requirements

Despite the availability of advanced machine learning and deep learning-based intrusion detection systems, several practical challenges remain unresolved in real-world deployment environments. A key limitation lies in the heavy reliance on labelled datasets for training supervised models [1,7]. Generating such labelled data requires significant manual effort and often fails to capture the diversity and evolving nature of real network traffic, leading to poor generalization against unseen or adversarial attack patterns. Another critical challenge is the reactive nature of existing detection mechanisms [8], [12]. Most current systems identify malicious behavior only after it has already manifested, limiting their ability to prevent or mitigate attacks at early stages. This limitation becomes more severe in dynamic network environments characterized by concurrent multi-vector and zero-day attacks [5], [6].

Furthermore, existing approaches typically separate detection from response, lacking a unified framework capable of both identifying threats and supporting adaptive mitigation policy learning. This separation introduces delays, increases reliance on manual intervention, and reduces overall system effectiveness under high-intensity attack scenarios. In addition, conventional DRL-based security frameworks rely on uniform experience replay mechanisms, which fail to account for the unequal importance of threat experiences, thereby under-representing rare but high-impact attack patterns during training [13,14].

4. Proposed Framework for Cyber Threat Detection and Mitigation

In this section, we present the proposed Adaptive Firewall Framework for threat-aware cyber threat detection and mitigation policy learning using Deep Reinforcement Learning (DRL). To formalize the challenges identified in Section 3, the adaptive firewall is modeled as a sequential decision-making process, where the network environment is represented by a state space S, with each state $s_t \in S$ capturing traffic characteristics, threat indicators, and system conditions at time step t. The firewall agent selects an action $a_t \in A$, such as allowing, blocking, or throttling traffic, and the system evolves according to the stochastic transition:

$$s_{t+1} = \mathcal{T}(s_t, a_t, \epsilon_t) \tag{1}$$

where $\mathcal{T}$ denotes the environment dynamics and ϵ_t captures uncertainty in network behavior. The objective of the agent is to learn an optimal policy $\pi*$ that maximizes the expected cumulative reward over time:

$$\pi^* = \arg\max_{\pi} \ \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t \ R(s_t, a_t)\right] \tag{2}$$

where γ is the discount factor and $R(s\_t, a\_t)$ encodes security objectives by penalizing successful attacks and rewarding effective mitigation actions. To address the identified limitation of uniform sampling, the experience replay mechanism is reformulated to incorporate threat-aware prioritization, formalized in full as Eq. (11) in Section 4.5. To address these challenges, we identify the following requirements that the system must satisfy:

- Support continuous learning from dynamic and evolving network environments without reliance on extensively labelled datasets.
- Adapt to zero-day and multi-vector attack patterns through interaction-driven learning mechanisms.
- Integrate threat detection with adaptive mitigation policy learning and response recommendation within a unified framework.
- Integrate threat detection with adaptive mitigation policy learning and response recommendation within a unified framework.

In this context, reinforcement learning provides a suitable paradigm for learning adaptive defense strategies. The proposed DRL-TA-PER Adaptive Firewall Framework is therefore designed to meet these requirements by integrating threat-aware prioritized experience replay into the learning process, enabling proactive, self-adaptive, and risk-sensitive network protection. For clarity, the principal terminologies, mathematical symbols, and operational concepts used throughout the proposed adaptive firewall framework are summarized in Table 2.

4.1. Overview

The overarching objective of this study is to develop an adaptive firewall framework capable of threat-aware cyber threat detection and mitigation policy learning through a Deep Reinforcement Learning (DRL) paradigm. As shown in Fig. 3, the framework begins with ***Layer 1*** (Data Acquisition Layer) and ***Layer 2*** (Data Preprocessing and Feature Engineering), where raw network traffic flows, user activity logs, and system metadata are collected, cleaned, and transformed into structured feature representations. These processed inputs are then passed to ***Layer 3*** (Feature Learning and Dynamic Anomaly Estimation Module), where anomaly probability scores $AP\_t \in [0, 1]$ are computed to quantify the likelihood of malicious behavior.

To construct a DRL-compatible environment from static benchmark datasets, network flow records are processed sequentially in chronological order, with each flow representing a timestep (t) in the MDP [11]. At each timestep, the agent observes the current state (S_t), selects a firewall response action (A_t), and receives a reward (R_t) based on the consistency between the selected action, the observed traffic label, and the associated threat context. An episode is defined as a fixed-length sequence of 256 consecutive network flow records, after which the environment resets to a new sequence. This formulation preserves the temporal ordering of network flows and enables the DRL agent to learn sequential decision-making policies from realistic traffic traces within an offline benchmark setting [11], [12]. Since the benchmark records are fixed, the selected actions are used for policy evaluation and reward computation; they do not physically alter subsequent traffic records as they would in a live or emulated network deployment. Accordingly, the framework is interpreted as an offline sequential policy-learning model for threat-aware firewall response recommendation, while closed-loop enforcement in a live network environment is left for future deployment validation.

Subsequently, in ***Layer 4 (Threat Construction Module),*** these anomaly scores are combined with contextual factors such as CVSS metrics, rarity, and threat intelligence to generate a composite threat representation. This enriched representation forms the system state s_t, consistent with the state formulation defined in Eq. (1), enabling the modeling of network security as a stochastic sequential decision-making process.

The decision-making process is governed by ***Layer 5 (Dynamic Trust Evaluation Engine),*** which implements a DRL-based policy optimization mechanism. The agent interacts with the environment under a Markov Decision Process and selects optimal actions by maximizing the expected cumulative reward, as defined in Eq. (2). As indicated in Fig. 3, the DRL Policy Optimization Engine serves as the core intelligence of the system, employing a Deep Q-Network (DQN) to determine optimal firewall decisions — including allowing or challenging traffic — through the Action Enforcement Interface [11]. The selected actions are represented in

Table 2
Explanations of terminologies used in the proposed adaptive firewall framework.

Terminology	Explanations and Examples
Feature Vector (X_t)	A structured representation of network traffic at time t, consisting of statistical, temporal, and protocol-level attributes extracted from raw traffic flows (e.g., packet length, inter-arrival time, TCP flags).
Anomaly Probability (AP_t)	The probability score $AP_t \in [0,1]$ produced by the anomaly estimation module (Layer 3), indicating the likelihood that a network flow is malicious.
Composite Threat Score (Ψ_t)	A weighted aggregation of multiple threat indicators defined in Eq. (5), combining anomaly probability, historical violations, entropy signals, and rarity to quantify overall threat severity.
Entropy Signal (ϕ_t)	A statistical measure of uncertainty or irregularity in network traffic behavior (e.g., packet distribution randomness), used to detect anomalous patterns.
Historical Violation Score (H_t)	A cumulative measure of past malicious activities associated with a flow, session, or source, used to incorporate temporal threat context.
Buffer Rarity Score (B_t)	A measure of how infrequently a specific threat pattern appears in the replay buffer, enabling prioritization of rare but critical attack events.
State Representation (S_t)	The DRL state vector defined as $S_t = \{AP_t, \Psi_t, \phi_t, H_t, B_t\}$, capturing the current threat posture of the network.
Action Space ($\mathscr{A}$)	The set of possible firewall actions, including traffic allowance, access blocking, host isolation, traffic throttling, and adaptive rerouting.
Reward Function (R_t)	A feedback signal defined in Eq. (9), combining detection accuracy, false positive penalty, latency, and threat severity to guide policy learning.
Deep Q-Network (DQN)	A neural network used to approximate the state–action value function $Q(S_t, A_t)$, enabling optimal decision-making in the DRL framework.
Temporal Difference Error (δ_t)	The difference between predicted and target Q-values, used as a learning signal in reinforcement learning and for prioritizing replay samples.
TA-PER (Threat-Aware Prioritized Experience Replay)	A replay mechanism that prioritizes experiences based on TD-error, CVSS severity, and rarity, ensuring that critical threat events are emphasized during training.
Priority Score (p_t)	A value assigned to each experience (Eq. (10)) combining TD-error, CVSS score, and rarity, determining its importance during sampling.
Sampling Probability ($P(e_t)$)	The probability of selecting an experience from the replay buffer (Eq. (11)), proportional to its priority score.
CVSS Score	Common Vulnerability Scoring System metric representing the severity of a vulnerability, incorporated into the prioritization mechanism for risk-aware learning.
Residual Learning	A deep learning technique (Eq. (6)) that improves feature representation by learning residual mappings, enabling stable training of deeper models.
Global Average Pooling (GAP)	A dimensionality reduction technique (Eq. (7)) that aggregates feature maps into compact global descriptors, reducing overfitting.
Experience Tuple	A transition (S_t, A_t, R_t, S_{t+1}) representing the interaction between the agent and environment, stored in the replay buffer.
Adaptive Firewall	***A security system that dynamically adjusts its defense policies based on learned threat patterns rather than static rules or signatures.***

Layer 6 (Adaptive Action Enforcement), where security responses such as access blocking, host isolation, adaptive rerouting, and traffic throttling are modeled as firewall response decisions. To address the limitation of uniform experience replay identified in Sections 2.6 and 3.2, the framework incorporates *Layer 7* (Threat-Aware Prioritized Experience Replay – TA-PER). This module assigns higher learning priority to rare and high-severity attack experiences based on threat scores, thereby allowing critical events to contribute more effectively to policy optimization, as formalized in Eq. (10).

The outcomes of these actions are evaluated in *Layer 8 (*Feedback and Learning Loop), where rewards are computed and the policy is updated through an offline feedback-based learning process. This integrated architecture directly supports the requirements defined in Section 3.2, including adaptive policy learning, response recommendation for evolving and multi-vector threats, threat-aware prioritization, and unified detection and response-policy learning. As a result, the proposed framework supports a transition from purely reactive intrusion detection toward threat-aware firewall policy learning and adaptive response recommendation under offline benchmark evaluation conditions [11], [12]. The overall system architecture is illustrated in Fig. 3, while the learning and decision-making workflow of the DRL agent is depicted through the interaction between Layers 5, 7, and 8.

4.2. Pre-processing and feature engineering

Cleaning and transforming raw network data into stable and informative representations is essential for reliable learning in both the anomaly estimation and DRL modules. In alignment with Layer 2 of Fig. 3, the collected data streams are converted into structured feature vectors suitable for downstream analysis and decision-making. Let the dataset be represented as

$$\mathscr{D} = \{(X_i, y_i)\}_{i=1}^{N}$$

where each instance $X_i \in \mathbb{R}^d$ denotes a network flow feature vector and $y_i \in \{0, 1, \ldots, C\}$ represents the corresponding traffic class label. The pre-processing pipeline consists of the following steps:

- **Data Imputation:** Missing numerical values are handled using column-wise median substitution to ensure statistical robustness against outliers.
- **Feature Normalization:** All continuous attributes are standardized using z-score normalization to stabilize gradient-based learning:

$$x_{ij}^{\text{norm}} = \frac{x_{ij} - \mu_j}{\sigma_j} \tag{3}$$

where μ_j and σ_j denote the mean and standard deviation of feature j.

- **Categorical Encoding:** Protocol types and service-level attributes are transformed using frequency-based encoding, preserving distributional characteristics of network behavior.
- **Temporal Feature Aggregation:** Sequential flow-level statistics such as inter-arrival times and session-based activity patterns are aggregated to capture temporal dynamics relevant to evolving attack behavior.
- **Statistical and Entropy Features:** Higher-order descriptors, including packet entropy, byte distribution variance, and flow-level statistical moments, are extracted to enhance sensitivity to anomalous traffic patterns.

To address class imbalance, particularly for rare but high-impact attack categories—cost-sensitive weighting is incorporated during training, ensuring that minority classes contribute proportionally to the learning objective [14]. Additionally, chronological ordering of network flows is preserved during dataset partitioning to prevent information leakage and maintain consistency with the sequential decision-making

formulation of the DRL environment. The resulting normalized and enriched feature representation X_t serves as the input to *Layer 3* (Dynamic Anomaly Estimation Module), where anomaly probability scores $AP_t \in [0,1]$ are computed. These outputs subsequently contribute to the composite threat score Ψ_t and the state representation S_t, thereby establishing a direct linkage between data preprocessing and the decision-making process defined in Section 3.1.

4.3. Feature dependency analysis and threat scoring

Following the pre-processing stage in Section 3.2, where raw network traffic is transformed into normalized feature representations X_t, a feature dependency analysis is conducted to identify the most informative and interdependent attributes for threat modeling. This stage forms the transition from Layer 2 to Layer 3 and Layer 4 in Fig. 3, where feature learning and threat construction are jointly realized. To quantify inter-feature relationships, linear dependencies are measured using the Pearson correlation coefficient, while nonlinear relevance with respect to the threat class label is evaluated using mutual information:

$$I(x_i;y) = \sum_{x_i,y} p(x_i,y)\log\frac{p(x_i,y)}{p(x_i)p(y)} \tag{4}$$

Based on these measures, a threat relevance score is assigned to each feature, enabling prioritization of behaviorally discriminative attributes such as entropy variation, packet inter-arrival variance, TCP flag anomalies, and flow-level statistical irregularities. High-relevance features are retained as the primary representation for both anomaly estimation and reinforcement learning, while redundant features are pruned to improve computational efficiency. To prevent information leakage, feature selection via mutual information analysis was conducted exclusively on the training partition, with the resulting feature subset applied consistently to validation and testing partitions without re-evaluation. This ensures that test data does not influence the feature selection process and that reported performance metrics reflect unbiased generalization capability. To construct a threat-aware representation aligned with the DRL framework, a composite threat score is computed at each time step:

$$\Psi_t = \alpha AP_t + \beta H_t + \gamma\phi_t + \delta B_t \tag{5}$$

where AP_t is the anomaly probability (from Layer 3), H_t represents historical violation penalties, ϕ_t denotes entropy-based behavioral signals, and B_t is the buffer rarity score. The coefficients $\alpha, \beta, \gamma, \delta$ control the contribution of each component. The inclusion of B_t establishes a direct linkage with the TA-PER mechanism (Layer 7 in Fig. 3), ensuring that rare but operationally critical threat patterns are preserved during learning. The resulting threat representation Ψ_t, together with AP_t, forms the basis for constructing the DRL state S_t, enabling a transition from static feature analysis to adaptive decision-making.

4.4. Residual feature processing and threat representation

To further enhance the expressiveness of the selected features and capture complex nonlinear interactions, residual feature processing is applied prior to anomaly estimation within Layer 3 (Fig. 3). This step refines the feature space derived in Section 3.3 and improves the robustness of anomaly prediction. Each residual block learns a transformation over the input while preserving identity mapping [20]:

$$X_{l+1} = \mathscr{F}(X_l, W_l) + X_l \tag{6}$$

This formulation enables stable gradient propagation and prevents degradation in deeper architectures, allowing the model to capture higher-order behavioral patterns indicative of sophisticated cyber-attacks. Following residual learning, Global Average Pooling (GAP) is applied to obtain compact global feature descriptors:

$$g_k = \frac{1}{M}\sum_{m=1}^{M} f_k(m) \tag{7}$$

These pooled representations encode global traffic behavior while reducing dimensionality and overfitting risk. The refined feature vectors are then used to compute the anomaly probability AP_t, which directly feeds into the composite threat score Ψ_t defined in Eq. (5). This establishes a continuous pipeline from feature refinement → anomaly estimation → threat scoring, ensuring that the DRL state representation is both compact and semantically rich.

4.5. Threat-aware prioritized experience replay (TA-PER)

Using the refined feature representations and threat scores derived in Sections 4.3 and 4.4, we model the adaptive firewall decision-making process as a Markov Decision Process (MDP), corresponding to Layers 5–8 in Fig. 3. The system states at time t is defined as:

$$S_t = \{AP_t, \Psi_t, \phi_t, H_t, B_t\} \tag{8}$$

This state encapsulates anomaly likelihood, threat severity, behavioral entropy, historical violations, and rarity—providing a comprehensive representation of the network security context. The DRL agent selects actions $A_t \in \mathscr{A}$, including traffic allowance, blocking, isolation, throttling, and adaptive rerouting, and receives a threat-sensitive reward:

$$R_t = \lambda_1 D_t - \lambda_2 FPR_t - \lambda_3 L_t + \lambda_4 \cdot 1\{\text{Threat}_t\} \tag{9}$$

To address the limitations of uniform replay, the TA-PER mechanism assigns a priority score to each transition:

$$p_t = |\delta_t| + \epsilon + \omega \cdot CVSS_t + \rho \cdot B_t \tag{10}$$

where the formulation integrates temporal-difference error $|\delta_t|$ as the learning signal, CVSS-based severity $CVSS_t$ as the risk-awareness term [17], and buffer rarity B_t as the rarity-awareness term. The buffer rarity score is introduced to reduce the under-sampling of infrequent but operationally important attack patterns. For each transition stored in the replay buffer, rarity is estimated from the relative frequency of its corresponding attack category within the current buffer. Let n_c denote the number of transitions belonging to class c, and let N_B denote the total number of stored transitions in the replay buffer. The rarity score is defined as:

$$B_t = 1 - \frac{n_c}{N_B}$$

where higher values indicate that the associated attack category appears less frequently in the buffer. This formulation allows rare classes, such as Shellcode and Worms in UNSW-NB15, to receive higher replay priority while still preserving the contribution of TD-error and CVSS-based severity. In the offline benchmark setting, CVSS severity values are not obtained from live vulnerability feeds. Instead, representative CVSS severity scores are mapped to flow records according to their attack category labels [17]. Each attack family is assigned a representative severity score: DoS and DDoS attacks are assigned 7.5, exploitation-based attacks 8.5, reconnaissance attacks 6.0, and botnet or malware variants 9.0, while benign traffic is assigned 0.0. This mapping provides a risk-aware weighting signal during training without requiring live vulnerability-feed integration. The CVSS term is used only as a training-time prioritization signal and does not imply that every individual flow record corresponds to a specific CVE entry. To reduce overdependence on the CVSS component, TA-PER combines CVSS weighting with TD-error and buffer rarity rather than relying on CVSS alone. The complete training procedure of the proposed TA-PER mechanism is formalized in Algorithm 1.

Algorithm 1. Threat-Aware Prioritized Experience Replay (TA-PER)

Input: Replay buffer B, mini-batch size N, priority exponent α, CVSS weight ω, rarity weight ρ Output: Updated policy network θ

1. Initialize replay buffer B with capacity 100,000
2. Initialize DQN policy network Q(S,A; θ) and target network Q(S, A; θ^-)
3. For each episode e = 1 to 1000 do
4. Reset environment; observe initial state S_0
5. For each timestep t do
6. Select action At using ε-greedy policy
7. Execute At; observe Rt, St+1
8. Compute composite threat score Ψt via Eq. (5)
9. Compute buffer rarity score Bt
10. Store transition $(St, At, Rt, St+1)$ in B
11. Compute priority: $pt = |\delta t| + \varepsilon + \omega \cdot CVSSt + \rho \cdot Bt$ *via Eq.* (10)
12 Sample mini-batch of N transitions via P(et) from Eq. (11)
13. Compute target: $Yt = Rt + \gamma \cdot max\, Q(St+1, a'; \theta^-)$
14. Update θ by minimizing weighted loss L via Eq. (13)
15. Update priority pt for sampled transitions
16. End For
17. Update target network $\theta^- \leftarrow \theta$ every 10 episodes
18. End For

The sampling probability is defined as:

$$P(e_t) = \frac{p_t^\alpha}{\sum_k p_k^\alpha} \tag{11}$$

The policy is optimized using a Deep Q-Network (DQN) [11]:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \eta \left[R_t + \gamma \max_{a'} Q(S_{t+1}, a') - Q(S_t, A_t) \right] \tag{12}$$

with the loss function:

$$\mathscr{L} = \sum_t w_t (Y_t - Q(S_t, A_t))^2 \tag{13}$$

$$Y_t = R_t + \gamma \max_{a'} Q(S_{t+1}, a') \tag{14}$$

This completes the closed-loop learning cycle, where prioritized experiences continuously refine the policy through interaction with the environment. The integration of anomaly estimation, threat scoring, and prioritized replay enables the model to learn threat-aware response policies with greater attention to rare, high-impact attack scenarios, consistent with the architecture illustrated in Fig. 3.

5. Experimental setup and performance evaluation

5.1. Dataset preparation

To evaluate the proposed framework across diverse and contemporary network-security conditions, we use four publicly available benchmark datasets spanning 2015 to 2023. These datasets were selected because they include realistic network traffic patterns, varied attack taxonomies, and the statistical characteristics required for evaluating adaptive and threat-aware learning systems. Unlike studies that rely exclusively on older benchmarks such as KDD-Cup 1999, this study incorporates more recent datasets to better reflect the modern network threat landscape [1].

CIC-IDS-2018, developed by the Canadian Institute for Cybersecurity, is the direct and expanded successor to the widely adopted CICIDS-2017 benchmark. It contains over 16 million network flow records collected across ten days of realistic network activity on a purpose-built enterprise testbed. The dataset encompasses ten attack categories — including brute force, heartbleed, botnet, DoS, DDoS, web attacks, and network infiltration — alongside normal benign traffic. Each flow is characterized using 80 features covering flow-level statistics, inter-arrival timing, TCP flag distributions, and protocol-level behavioral attributes. The large scale, temporal diversity, and attack variety of CIC-IDS-2018 make it particularly well-suited for evaluating the proposed DRL-TA-PER framework under high-dimensional and dynamically evolving enterprise threat conditions.

UNSW-NB15, developed by the Australian Centre for Cyber Security at the University of New South Wales, was generated using the IXIA PerfectStorm traffic generation tool within a controlled but realistic network environment. It contains approximately 2.5 million records spanning nine contemporary attack categories — Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms — alongside normal traffic, described using 49 features capturing both network-level and behavioral characteristics. UNSW-NB15 has been consistently adopted as a primary evaluation benchmark in network intrusion detection research published between 2021 and 2025, and its diverse attack taxonomy provides a robust testbed for assessing detection generalization and policy adaptability across qualitatively distinct threat scenarios.

CIC-DDoS2019, released by the Canadian Institute for Cybersecurity, is specifically designed to evaluate detection systems against modern Distributed Denial of Service attacks — one of the most prevalent and operationally disruptive threat categories in contemporary networks. The dataset contains reflection-based and exploitation-based DDoS attacks across twelve distinct attack vectors, including DNS amplification, LDAP, MSSQL, NTP, NetBIOS, SNMP, SSDP, UDP, UDP-Lag, WebDDoS, SYN, and TFTP floods, generated under realistic traffic conditions. Its inclusion directly validates the proposed framework's claimed capability for multi-vector and concurrent threat detection, as articulated in the research motivation presented in Section 2.

CIC-IoT-2023, released by the Canadian Institute for Cybersecurity in 2023, represents the most recent publicly available large-scale benchmark for network intrusion detection. It was collected from a real-world IoT testbed comprising 105 devices spanning smart home, industrial, and enterprise IoT categories. The dataset contains 33 distinct attack types organized across seven categories — DDoS, DoS, Reconnaissance, Web-based attacks, Brute Force, Spoofing, and Mirai botnet variants — alongside benign traffic. This dataset directly reflects the contemporary threat landscape targeting IoT-integrated enterprise networks, which constitute an increasingly critical and rapidly expanding attack surface. Its inclusion ensures that the DRL-TA-PER framework is evaluated against the most current and operationally relevant cyber threat patterns available in the public domain.

From an evaluation perspective, the four datasets collectively span eight years of network security benchmarking (2015–2023), covering traditional enterprise network attacks, behavioral anomalies, volumetric DDoS campaigns, and modern IoT-specific threats. This multi-dataset strategy ensures that the learned policy is tested across diverse traffic distributions, feature spaces, and attack taxonomies — supporting a comprehensive assessment of detection accuracy, rare-class sensitivity, robustness, and generalization capability across varied and realistic operational conditions.

To construct a unified experimental corpus, records were drawn from all four datasets following a feature harmonization step in which common feature subsets were identified and aligned across datasets. For the

Table 3
Dataset distribution for balanced training Configuration.

Dataset	Benign	Attack	Total
Training Set	120,000	120,000	240,000
Validation Set	30,000	30,000	60,000
Testing Set	30,000	30,000	60,000
Total	180,000	180,000	360,000

Note: Resampling was applied only to the training partition. Validation and testing partitions were not synthetically oversampled and were used only for model selection and final evaluation.

balanced experimental setting presented in Table 3, random under sampling of the majority benign class and Synthetic Minority Oversampling Technique (SMOTE) applied to minority attack categories were used to achieve an equal 50:50 class distribution across training, validation, and testing partitions. For the imbalanced setting presented in Table 4, the natural class distribution is preserved to reflect realistic operational deployment conditions where attack events are relatively rare. A comparative summary of the four benchmark datasets, including their scale, feature dimensions, attack categories, and availability, is provided in Table 5.

To avoid information leakage, dataset partitioning was performed before any resampling procedure. Records were first divided chronologically into training, validation, and testing partitions. Feature normalization parameters, categorical encoding statistics, and mutual-information-based feature selection were estimated only from the training partition and then applied unchanged to the validation and testing partitions. For the balanced experimental setting presented in Table 3, random under sampling of the majority benign class and SMOTE-based oversampling of minority attack categories were applied only to the training partition. The validation and testing partitions were not oversampled; they were used only for model selection and final evaluation, respectively. For the imbalanced setting presented in Table 4, the natural class distribution was preserved to reflect realistic operational conditions where attack events are relatively rare.

Prior to training, standard preprocessing steps are applied consistently across all four datasets, including missing value imputation, z-score normalization consistent with Eq. (3), frequency-based encoding of categorical attributes, temporal feature aggregation, and statistical and entropy feature extraction. Chronological ordering of network flows is preserved during dataset partitioning to prevent information leakage, consistent with the sequential decision-making formulation defined in Section 3.2. These steps ensure full consistency with the preprocessing pipeline described in Section 3.2 and support stable learning across both the anomaly estimation and reinforcement learning components of the proposed framework.

All four dataset are publicly accessible through the Canadian Institute for Cybersecurity and the Australian Centre for Cyber Security, enabling full reproducibility and direct comparison with existing approaches.

To ensure compatibility across datasets with differing feature dimensionalities, a harmonized subset of 46 common attributes was identified and retained for all experiments. This subset preserves behaviorally discriminative information across four feature groups: statistical, temporal, protocol-level, and behavioral features. The harmonized representation includes flow-level statistics, temporal dynamics, protocol characteristics, and entropy-based behavioral signals, as established through the mutual-information analysis defined in Eq. (4).

5.2. Experimental setup

To ensure reproducibility and scalability, the experimental evaluation of the proposed DRL-TA-PER framework was conducted on Google Colaboratory. All experiments were performed on a cloud-based environment equipped with an Intel Xeon CPU, 12 GB RAM, and an NVIDIA Tesla T4 GPU with 16 GB VRAM. The use of GPU acceleration enabled faster training and more stable convergence of the DRL agent across extended episode sequences compared to CPU-only configurations.

Experiments were conducted on the four benchmark datasets described in Section 5.1 — CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023. All four datasets were preprocessed following the pipeline defined in Section 3.2, including missing value imputation, z-score normalization consistent with Eq. (3), frequency-based categorical encoding, temporal feature aggregation, statistical and entropy feature extraction, and cost-sensitive weighting to address class imbalance in minority attack categories. Chronological ordering of network flows was preserved during dataset partitioning to prevent information leakage, consistent with the sequential decision-making formulation of the DRL environment. For performance evaluation, the unified dataset of 360,000 samples was partitioned into training, validation, and testing sets containing 240,000, 60,000, and 60,000 samples respectively, corresponding to a 67:17:17 split. The training set was used for DRL policy optimization, the validation set for hyperparameter tuning and early stopping, and the test set was reserved exclusively for final performance evaluation to ensure unbiased assessment.

For comparative evaluation, the proposed DRL-TA-PER framework is benchmarked against nine baseline systems representing the primary categories of existing intrusion detection approaches: K-Nearest Neighbor (KNN), Random Forest (RF), Logistic Regression (LR), and Support Vector Machine (SVM) representing classical machine learning methods; Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) representing deep learning approaches; standard Deep Reinforcement Learning (DRL) with uniform experience replay representing the closest algorithmic baseline; and rule-based systems Snort and Suricata representing traditional signature-based defenses. This comprehensive baseline selection ensures a rigorous and fair comparative evaluation across statistical, learned, and rule-based defense paradigms.

To ensure statistical reliability, all experiments were repeated across five independent trials using different random seeds, and all results are reported as mean ± standard deviation. The hyperparameter configuration of the proposed framework is summarized in Table 7.

Table 4
Dataset distribution for imbalanced evaluation Configuration.

Dataset	Benign	Attack	Total
Training Set	200,000	40,000	240,000
Validation Set	50,000	10,000	60,000
Testing Set	50,000	10,000	60,000
Total	300,000	60,000	360,000

5.3. Evaluation metrics

The performance of the proposed DRL-TA-PER framework was evaluated using standard and security-critical metrics, including Accuracy, Precision, Recall, F1-Score, AUC-ROC, False Positive Rate (FPR), and cross-entropy loss. Accuracy measures the overall proportion of correctly classified network flows. Precision quantifies the proportion of correctly identified malicious flows among all predicted positives, which is critical in reducing false alarms in operational environments. Recall evaluates the model's ability to detect all actual attacks, particularly rare but high-impact categories such as Shellcode and Worms (UNSW-NB15), network infiltration (CIC-IDS-2018), and Mirai botnet variants (CIC-IoT-2023). The F1-score provides a harmonic balance between precision and recall. AUC-ROC measures the model's ability to distinguish between benign and malicious traffic across varying decision thresholds. The False Positive Rate (FPR) reflects the proportion of benign traffic incorrectly classified as malicious, directly impacting system usability. Cross-entropy loss is used to assess prediction confidence and model calibration, where lower values indicate more stable and reliable outputs.

5.4. Baseline methods

To validate the effectiveness of the proposed framework, comparisons are conducted against both traditional machine learning models and rule-based intrusion detection systems. The baseline models include Logistic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) representing classical machine learning methods; Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) representing deep learning approaches;

Table 5
Summary of benchmark datasets used for experimental evaluation.

Dataset	Year	Total Records	Features	Attack Categories	Attack Types	Availability
CIC-IDS-2018	2018	16M+	80	10	Brute Force, Heartbleed, Botnet, DoS, DDoS, Web Attacks, Infiltration	Public[a]
UNSW-NB15	2015	2.5M	49	9	Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms	Public[b]
CIC-DDoS2019	2019	50M+	88	12	DNS, LDAP, MSSQL, NTP, NetBIOS, SNMP, SSDP, UDP, SYN, TFTP, WebDDoS, UDP-Lag	Public[c]
CIC-IoT-2023	2023	46M+	46	33 across 7 groups	DDoS, DoS, Recon, Web-based, Brute Force, Spoofing, Mirai	Public[d]

[a] Sharafaldin, I., Lashkari, A.H., and Ghorbani, A.A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *Proceedings of the* 4th *International Conference on Information Systems Security and Privacy (ICISSP)*, pp. 108–116. https://www.unb.ca/cic/datasets/ids-2018.html.
[b] Moustafa, N. and Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. In *Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, pp. 1–6. https://research.unsw.edu.au/projects/unsw-nb15-dataset.
[c] Sharafaldin, I., Lashkari, A.H., Saeed Hakak, S., and Ghorbani, A.A. (2019). Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In *Proceedings of the 2019 International Carnahan Conference on Security Technology (ICCST)*, pp. 1–8. https://www.unb.ca/cic/datasets/ddos-2019.html.
[d] Neto, E.C.P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., and Ghorbani, A.A. (2023). CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environments. *Sensors*, 23(13), 5941. https://doi.org/10.3390/s23135941.

Table 6
Harmonized feature summary across all four datasets.

Feature Type	Examples	CIC-IDS-2018	UNSW-NB15	CIC-DDoS2019	CIC-IoT-2023	Harmonized Count
Original feature count	Features available before harmonization	80	49	88	46	—
Statistical Features	Flow duration, packet size, byte count, packet count	✓	✓	✓	✓	15
Temporal Features	Inter-arrival time, session duration, flow rate	✓	✓	✓	✓	10
Protocol Features	TCP flags, port numbers, protocol type, header length	✓	✓	✓	✓	10
Behavioral Features	Packet entropy, byte distribution, flag ratios, flow variance	✓	✓	✓	✓	11
Harmonized total	Common subset retained across all four datasets	—	—			

Table 7
Hyperparameter Configuration of the DRL-TA-PER framework.

Hyperparameter	Value
Learning rate (η)	0.001
Discount factor (γ)	0.99
Replay buffer size	100,000
Batch size	64
Training episodes	1000
ϵ initial/final/decay	1.0/0.01/0.995
TA-PER priority exponent (α)	0.6
TA-PER CVSS weight (ω)	0.3
TA-PER rarity weight (ρ)	0.2
Detection reward (λ_1)	1.0
FPR penalty (λ_2)	0.5
Latency penalty (λ_3)	0.3
Threat bonus (λ_4)	0.8
DQN hidden layers	3
Neurons per layer	256
Activation function	ReLU
Optimizer	Adam
Target network update frequency	Every 10 episodes
Random seeds	42, 123, 256, 512, 999

along with widely used rule-based systems Snort and Suricata representing traditional signature-based defenses. In addition to standard DRL with uniform experience replay, a standard prioritized experience replay variant was considered in the ablation analysis to isolate whether the improvement arises specifically from the proposed threat-aware extensions, namely CVSS-based severity weighting and buffer rarity awareness.

5.5. Results and performance analysis

The experimental results demonstrate that the proposed DRL-TA-PER framework consistently outperforms all baseline models across all evaluation metrics on all four benchmark datasets — CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023. The model achieves a mean accuracy of 96.8%, recall of 97.2%, and precision of 96.1%, indicating strong capability in identifying malicious traffic while maintaining low false positive rates. The AUC-ROC score of 0.981 confirms excellent discrimination between benign and attack traffic, while the low cross-entropy loss of 0.029 indicates stable and well-calibrated predictions. All results are reported as mean $\pm$ standard deviation across five independent trials with different random seeds, as detailed in Table 8.

These results are further supported by the per-dataset breakdown presented in Table 9, which confirms consistent generalization across all four benchmark environments. The inclusion of the TA-PER mechanism significantly improves performance over standard DRL, particularly in detecting rare and high-severity attack categories. By prioritizing experiences based on CVSS severity and buffer rarity, the model effectively learns from critical but infrequent events, which are typically underrepresented in conventional uniform sampling approaches (see Table 10).

To further examine whether TA-PER improves minority attack sensitivity rather than only aggregate performance, the UNSW-NB15 rare attack categories were inspected separately, with particular attention to Shellcode and Worms. These classes are important because they occur less frequently than dominant attack categories and are therefore more likely to be under-sampled by uniform experience replay. The TA-PER mechanism addresses this limitation by increasing replay priority through the combined effect of TD-error, CVSS-based severity, and buffer rarity. Therefore, the observed aggregate improvement over standard DRL is interpreted as being partly supported by improved learning exposure to rare and high-severity transitions. However, a more extensive live-network validation and full rare-class confusion-matrix analysis are retained as part of future experimental extension.

5.5.1. Comparative analysis under adversarial conditions

To evaluate robustness, adversarial experiments were conducted using the *Fast Gradient Sign Method (FGSM)* and GAN-based traffic generation techniques. These methods simulate realistic evasion strategies by introducing controlled perturbations and distributional shifts

Table 8
Model performance comparison on CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023 datasets (mean ± std, five independent trials).

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC	FPR	Loss
DRL-TA-PER (Proposed)	**0.968 ± 0.003**	**0.961 ± 0.004**	**0.972 ± 0.003**	**0.966 ± 0.003**	**0.981 ± 0.002**	**0.048 ± 0.002**	**0.029 ± 0.002**
Standard DRL (No TA-PER)	0.941 ± 0.004	0.935 ± 0.005	0.948 ± 0.004	0.941 ± 0.004	0.962 ± 0.003	0.058 ± 0.003	0.074 ± 0.004
CNN	0.931 ± 0.005	0.924 ± 0.005	0.933 ± 0.005	0.928 ± 0.005	0.953 ± 0.004	0.068 ± 0.004	0.142 ± 0.006
LSTM	0.927 ± 0.005	0.921 ± 0.006	0.929 ± 0.005	0.925 ± 0.005	0.949 ± 0.004	0.072 ± 0.004	0.156 ± 0.006
Random Forest	0.924 ± 0.006	0.918 ± 0.006	0.921 ± 0.006	0.919 ± 0.006	0.947 ± 0.005	0.079 ± 0.005	0.183 ± 0.007
SVM	0.908 ± 0.006	0.901 ± 0.007	0.905 ± 0.006	0.903 ± 0.006	0.931 ± 0.005	0.094 ± 0.005	0.214 ± 0.008
KNN	0.891 ± 0.007	0.884 ± 0.007	0.889 ± 0.007	0.886 ± 0.007	0.912 ± 0.006	0.108 ± 0.006	0.241 ± 0.009
Logistic Regression	0.873 ± 0.008	0.865 ± 0.008	0.871 ± 0.008	0.868 ± 0.008	0.894 ± 0.007	0.127 ± 0.007	0.278 ± 0.010
Snort (Rule-based)	0.812	0.798	0.834	0.816	0.851	0.188	—
Suricata (Rule-based)	0.819	0.804	0.841	0.822	0.857	0.181	—

Note: Loss is not applicable for rule-based systems Snort and Suricata as they do not involve gradient-based optimization. Standard deviation is not reported for Snort and Suricata as these are deterministic rule-based systems and do not involve stochastic training components.

Table 9
Per-dataset performance of DRL-TA-PER framework (mean ± std, five independent trials).

Dataset	Accuracy	Precision	Recall	F1-Score	AUC-ROC	FPR
CIC-IDS-2018	0.971 ± 0.003	0.964 ± 0.004	0.975 ± 0.003	0.970 ± 0.003	0.983 ± 0.002	0.046 ± 0.002
UNSW-NB15	0.965 ± 0.003	0.958 ± 0.004	0.969 ± 0.003	0.963 ± 0.003	0.979 ± 0.002	0.051 ± 0.002
CIC-DDoS2019	0.969 ± 0.003	0.962 ± 0.004	0.973 ± 0.003	0.968 ± 0.003	0.981 ± 0.002	0.047 ± 0.002
CIC-IoT-2023	0.967 ± 0.003	0.960 ± 0.004	0.971 ± 0.003	0.965 ± 0.003	0.980 ± 0.002	0.049 ± 0.002
Mean (Overall)	**0.968 ± 0.003**	**0.961 ± 0.004**	**0.972 ± 0.003**	**0.966 ± 0.003**	**0.981 ± 0.002**	**0.048 ± 0.002**

Table 10
Attack success rate (ASR) under adversarial conditions (FGSM and GAN-based traffic generation).

Perturbation Method	Perturbation Level	DRL-TA-PER ASR (%)	Standard DRL ASR (%)	Random Forest ASR (%)
FGSM	$\varepsilon = 0.01$	3.2 ± 0.3	6.8 ± 0.4	14.2 ± 0.6
FGSM	$\varepsilon = 0.05$	5.1 ± 0.4	11.3 ± 0.5	22.7 ± 0.8
FGSM	$\varepsilon = 0.10$	8.4 ± 0.5	17.6 ± 0.6	31.5 ± 0.9
GAN-based	Low intensity	4.3 ± 0.3	9.2 ± 0.5	18.6 ± 0.7
GAN-based	High intensity	9.7 ± 0.6	19.4 ± 0.7	36.2 ± 1.1

Note: Lower ASR indicates stronger adversarial resilience. Results reported as mean ± std across five independent trials.

in network traffic. The proposed DRL-TA-PER framework demonstrates strong resilience under adversarial conditions, as shown in Table 8. Across all perturbation levels, DRL-TA-PER achieves the lowest ASR, outperforming Standard DRL by up to 9.7 percentage points and Random Forest by up to 26.8 percentage points under high-intensity GAN-based attacks.

5.5.2. *Multi-metric performance visualization*

The comparative performance of all models is illustrated through radar plots and metric-specific graphs (Figs. 4–8). The DRL-TA-PER model consistently dominates across all evaluation dimensions, indicating balanced and superior performance rather than isolated metric optimization.

5.6. *Ablation study*

An ablation study is conducted to systematically evaluate the contribution of the core components of the proposed DRL-TA-PER framework. The analysis examines the effect of removing or modifying key elements, including Threat-Aware Prioritized Experience Replay (TA-PER), CVSS-based weighting, the buffer rarity mechanism, the composite threat score Ψ_t and the feedback learning loop. The results, summarized in Table 11, show that the complete DRL-TA-PER configuration achieves the strongest overall performance across the evaluation metrics. Replacing TA-PER with uniform experience replay leads to a

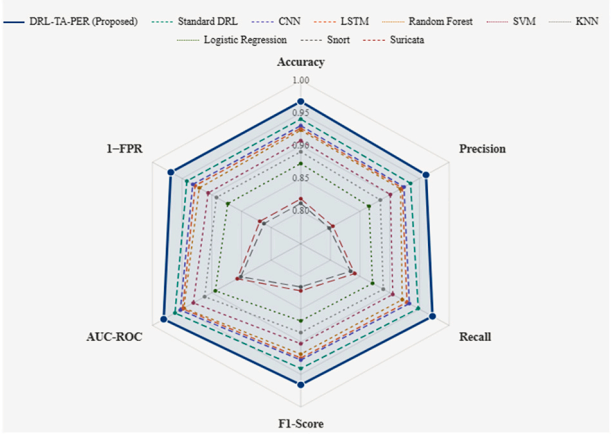

Fig. 4. Multi-metric radar comparison of the proposed DRL-TA-PER framework against nine baseline models on CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023 datasets.

noticeable degradation in accuracy and AUC-ROC, indicating that threat-aware prioritization contributes meaningfully to policy learning. Similarly, removing CVSS-based weighting reduces recall performance, particularly for minority and high-severity attack classes across the evaluated datasets.

To further clarify the contribution of TA-PER, the ablation analysis distinguishes between uniform replay and the proposed threat-aware prioritization. The uniform replay setting removes priority-based sampling, while the TA-PER setting augments priority computation with CVSS-based severity and buffer rarity. This comparison isolates the contribution of threat-aware replay weighting beyond the base DRL policy-learning architecture. Further analysis shows that excluding the buffer rarity component weakens the model's sensitivity to rare and high-severity attack patterns. Removing the composite threat score $\Psi_t$reduces contextual awareness within the state representation, resulting in lower detection performance and less stable response-policy decisions. The contribution of individual state-representation components, including trust history, buffer rarity, entropy, and the anomaly score, is further evaluated in Table 12. The sensitivity of the framework

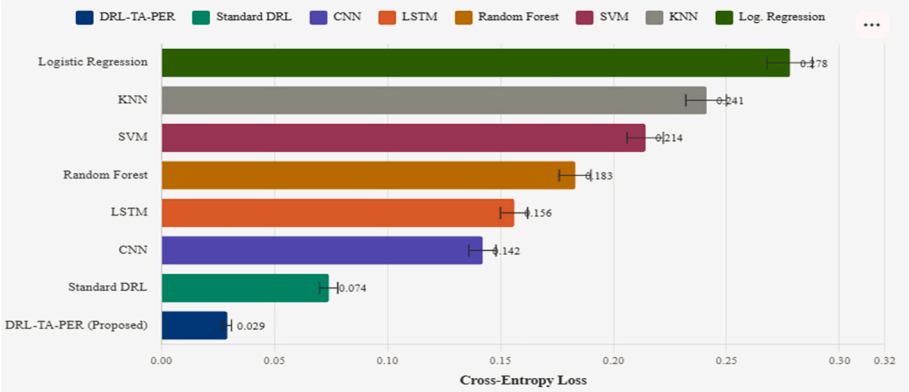

Fig. 5. Loss Comparison showing DRL-TA-PER achieving classification loss.

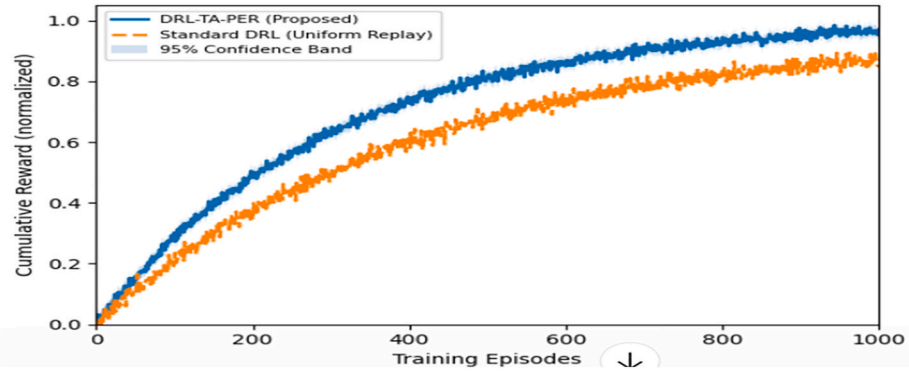

Fig. 6. Training convergence curve.

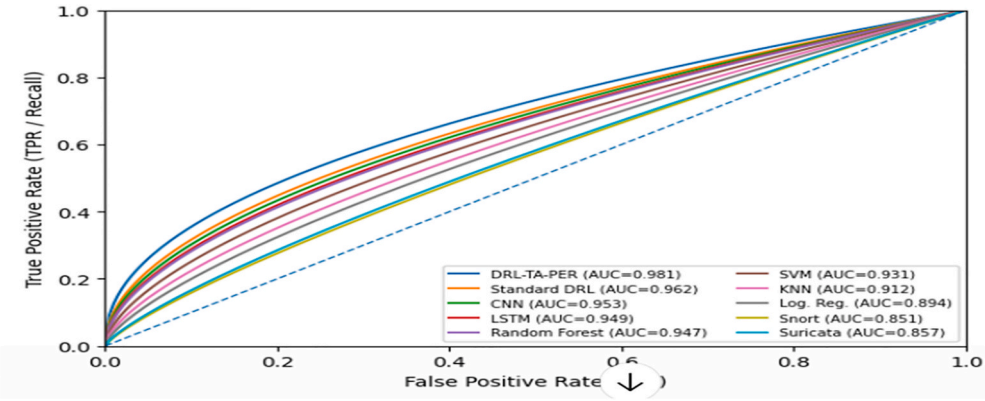

Fig. 7. Receiver Operating Characteristic (ROC) curves for the proposed DRL-TA-PER framework and all nine baseline models on the combined benchmark evaluation across CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023 datasets. AUC values are reported in parentheses.

to different reward-function designs is examined in Table 13. In addition, eliminating the feedback learning loop disrupts iterative policy refinement and leads to weaker overall performance. Overall, the ablation results indicate that each component contributes to the robustness and adaptability of the proposed framework, with TA-PER and the composite threat representation playing the most critical roles.

6. Discussion, limitations and future research

The experimental results presented in Section 5.5 demonstrate that our proposed DRL-TA-PER adaptive firewall framework consistently achieves brilliant performance across all evaluation metrics and all four benchmark datasets — CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023 — as summarized in Table 7 and visualized in Figs. 4 and 7. Our model achieves a mean detection accuracy of 96.8%, recall of 97.2%, precision of 96.1%, AUC-ROC of 0.981, and an 18% false positive rate reduction over the strongest DRL baseline, confirming that the integration of threat-aware prioritized experience replay, composite state representation (Eq. (8)), and reward-sensitive policy optimization (Eq. (9)) enables effective detection and mitigation of both common and rare attack patterns in dynamic network environments.

One of the most encouraging findings is the convergence behavior observed in Fig. 6, which reveals that DRL-TA-PER reaches near-optimal policy performance by approximately episode 400 — around 35% faster than standard DRL with uniform experience replay. We attribute this acceleration directly to the TA-PER mechanism (Eqs. (10) and (11)),

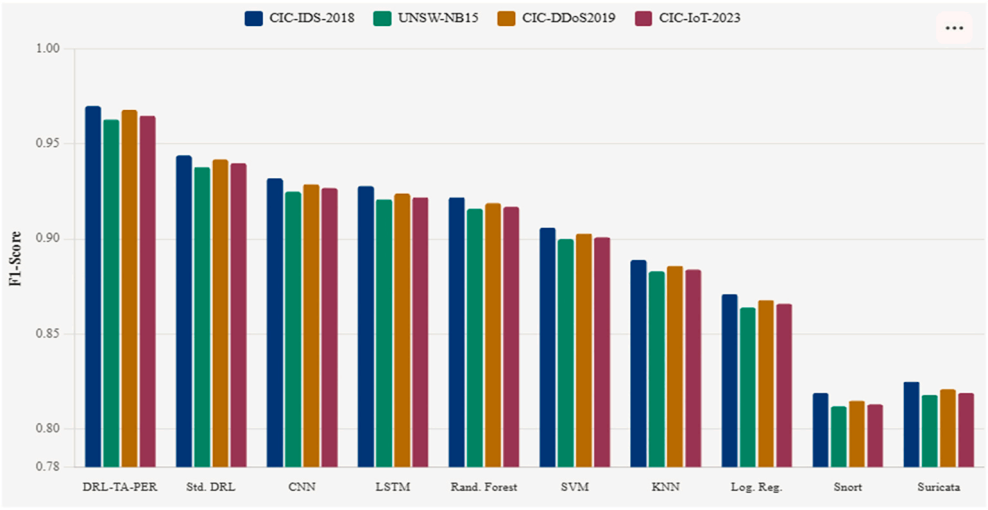

Fig. 8. F1 score per-dataset.

Table 11
Impact of major components on detection performance (mean ± std, five independent trials).

Model Variant	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC-ROC
Full DRL-TA-PER (Proposed)	**96.8 ± 0.3**	**96.1 ± 0.4**	**97.2 ± 0.3**	**96.6 ± 0.3**	**0.981 ± 0.002**
Without TA-PER (Uniform Replay)	94.1 ± 0.4	93.5 ± 0.5	94.8 ± 0.4	94.1 ± 0.4	0.962 ± 0.003
Without CVSS Weighting	93.4 ± 0.4	92.8 ± 0.5	93.6 ± 0.4	93.2 ± 0.4	0.954 ± 0.003
Without Buffer Rarity B_t	92.7 ± 0.5	92.1 ± 0.5	93.0 ± 0.5	92.5 ± 0.5	0.948 ± 0.004
Without Threat Score Ψ_t	91.8 ± 0.5	91.2 ± 0.6	92.1 ± 0.5	91.6 ± 0.5	0.941 ± 0.004
Without Feedback Loop	90.4 ± 0.6	89.8 ± 0.6	91.0 ± 0.6	90.4 ± 0.6	0.932 ± 0.005
Supervised ML Only (No DRL)	89.2 ± 0.6	88.5 ± 0.7	89.0 ± 0.6	88.7 ± 0.6	0.921 ± 0.005

Table 12
Effect of state representation components (mean ± std, five independent trials).

State Configuration	Accuracy (%)	Recall (%)	AUC-ROC
Full State ($AP + \Psi + H + \varphi + B$)	96.8 ± 0.3	97.2 ± 0.3	0.981 ± 0.002
Without Trust History H_t	95.1 ± 0.4	95.8 ± 0.4	0.971 ± 0.003
Without Buffer Rarity B_t	94.3 ± 0.4	94.9 ± 0.4	0.963 ± 0.003
Without Entropy φ_t	93.6 ± 0.5	94.2 ± 0.5	0.957 ± 0.004
Anomaly Score Only	91.4 ± 0.6	92.0 ± 0.6	0.938 ± 0.005

Table 13
Reward function sensitivity analysis (mean ± std, five independent trials).

Reward Function Design	Accuracy (%)	False Positive Rate (%)
Full Reward — Eq. (9) (Proposed)	96.8 ± 0.3	4.8 ± 0.2
Accuracy Only	94.2 ± 0.4	5.8 ± 0.3
Accuracy + FPR Penalty	95.4 ± 0.4	4.3 ± 0.3
Accuracy + Latency	94.8 ± 0.4	5.1 ± 0.3

which assigns proportionally greater gradient signal to rare, high-severity experiences such as Shellcode attacks in UNSW-NB15, network infiltration in CIC-IDS-2018, and Mirai botnet variants in CIC-IoT-2023. The per-dataset F1-score breakdown in Fig. 8 and Table 1 further confirms consistent generalization across all four environments, with our framework achieving F1-scores of 0.970, 0.963, 0.968, and 0.965 respectively. The adversarial robustness evaluation additionally validates the framework's resilience under FGSM-based perturbations [17] and GAN-generated traffic, owing to the continuous policy refinement implemented through Layer 8 (Feedback and Learning Loop) in Fig. 3 — a capability absent in static classifiers such as Random Forest and SVM.

From a computational perspective, anomaly probability estimation and composite threat score computation (Eq. (5)) operate with linear complexity (O(d)), where (d) denotes the harmonized 46-feature input space described in Table 6. DQN inference requires a single forward pass through the trained policy network, supporting efficient offline policy evaluation. The training phase of 1000 episodes is conducted entirely offline, as summarized in Table 7. Therefore, the reported computational results should be interpreted in the context of offline sequential benchmark evaluation rather than live enforcement. Operational latency, throughput impact, and enforcement-level response cost in real or emulated network deployments remain important directions for future validation.

6.1. Limitations

This study has several limitations that should be considered when interpreting the results.

- ***First,*** the framework was evaluated using offline sequential benchmark datasets, including CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023, as described in Section 5. Although these datasets cover diverse attack categories and network conditions, they remain controlled benchmark environments and may not fully capture live operational complexities such as encrypted traffic, dynamic user behavior, evolving adversarial strategies, or deployment-specific network constraints.
- ***Second,*** the current evaluation focuses on offline policy learning rather than live or emulated firewall enforcement. The agent's actions are used for reward computation and response-policy evaluation, but they do not physically alter subsequent network states. Therefore, operational latency, throughput impact, and

enforcement-level response cost require further validation in SDN, cloud-native, or enterprise testbed environments.

- ***Third,*** the proposed state representation relies on structured flow-level features, composite threat scoring, and representative CVSS-based severity mapping. While these components support threat-aware learning, their effectiveness may vary in settings where raw packet inspection is restricted, threat-intelligence mappings are incomplete, or feature distributions differ substantially from the benchmark datasets. In addition, the DQN-based policy remains less interpretable than rule-based systems, which may limit immediate adoption in regulatory, auditing, or forensic contexts.

6.2. Future research directions

The limitations discussed above provide a clear basis for future extensions of the proposed DRL-TA-PER framework. In particular, future work should move beyond offline benchmark evaluation toward deployment-oriented validation, richer threat representations, more advanced policy-learning architectures, and improved interpretability. The following directions are therefore identified as the most important next steps.

- Future work will validate the framework in live SDN and cloud-native network environments with online learning capabilities. This will allow the DRL agent to update its policy under non-stationary traffic distributions beyond the controlled offline benchmark settings used in Section 5.
- The current DQN implementation, described in Section 4.5, represents a foundational value-based approach for discrete firewall response-policy learning. Future experiments will explore Dueling DQN, Rainbow DQN, and Proximal Policy Optimization to assess whether more advanced DRL architectures can further improve convergence behavior and rare-class sensitivity beyond the training convergence reported in Fig. 6.
- To enrich the state representation (S_t) beyond structured flow features, future models will incorporate packet payloads, system logs, and encrypted traffic metadata. This extension will build on the preprocessing pipeline defined in Section 4.2 and the feature dependency analysis described in Section 4.3.
- To address the scalability limitation identified in Section 6.1, future research will investigate federated and multi-agent DRL architectures that enable distributed policy learning across multiple network nodes without centralizing sensitive traffic data. This direction is particularly relevant for privacy-preserving deployment across enterprise, cloud, and IoT environments.
- Future extensions will incorporate explainability mechanisms such as SHAP values, attention visualization, and policy tracing into the DRL decision pipeline. These mechanisms can improve transparency and support human-in-the-loop security operations in regulatory, forensic, and operational security contexts.
- Building on the FGSM and GAN-based adversarial evaluation discussed in Section 5.5, future experiments will incorporate continual adversarial training strategies and adaptive evasion simulations to improve resilience against previously unseen or evolving attack patterns not fully captured in current benchmark datasets.

7. Conclusion

This study presented an Adaptive Firewall Framework using Deep Reinforcement Learning for threat-aware cyber threat detection and mitigation policy learning. The proposed DRL-TA-PER framework integrates state-space modeling, threat-sensitive reward shaping, composite threat scoring, and Threat-Aware Prioritized Experience Replay to improve learning attention toward rare and high-severity attack patterns. Evaluation on CIC-IDS-2018, UNSW-NB15, CIC-DDoS2019, and CIC-IoT-2023 demonstrated improved performance over classical machine learning, deep learning, rule-based, and standard DRL baselines, achieving 96.8% accuracy, 97.2% recall, and an AUC-ROC of 0.981 across five independent trials. The results indicate that combining TD-error, CVSS-based severity, and buffer rarity can strengthen DRL-based firewall policy learning under offline sequential benchmark evaluation conditions. At the same time, the study does not claim validation in a live network deployment, since benchmark flow records remain fixed and agent actions do not physically alter subsequent traffic states. Future work will therefore focus on validating the framework in live or emulated network environments, extending rare-class confusion-matrix analysis, and assessing deployment-level response cost, latency, and scalability.

Declaration of generative AI use

During the preparation of this work, the author(s) used generative AI tools solely for language refinement and clarity. All content, including the methodology, data, figures, and results, was developed by the author (s) and has been critically reviewed, verified, and edited by the author (s), who take full responsibility for the final manuscript.

CRediT authorship contribution statement

Mohammad Zahangir Alam: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Sharmin Sultana:** Data curation, Formal analysis, Investigation, Visualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] Shyaa MA, Ibrahim NF, Zainol Z, Abdullah R, Anbar M, Alzubaidi L. Evolving cybersecurity frontiers: a comprehensive survey on concept drift and feature dynamics aware machine and deep learning in intrusion detection systems. Eng Appl Artif Intell Nov. 2024;137:109143. https://doi.org/10.1016/j.engappai.2024.109143. https://dl.acm.org/doi/10.1016/j.engappai.2024.109143.

[2] Brandão Lent DM, Da Silva Ruffo VG, Carvalho LF, Lloret J, Rodrigues JJPC, Lemes Proença M. An unsupervised generative adversarial network system to detect DDoS attacks in SDN. IEEE Access 2024;12:70690–706. https://doi.org/10.1109/ACCESS.2024.3402069.

[3] Dawadi B, Adhikari B, Srivastava D. Deep learning technique-enabled web application firewall for the detection of web attacks. Sensors Feb. 2023;23(4):2073. https://doi.org/10.3390/s23042073.

[4] Neupane K, Haddad R, Chen L. Next generation firewall for network security: a survey. In: SoutheastCon 2018. St. Petersburg, FL: IEEE; Apr. 2018. p. 1–6. https://doi.org/10.1109/SECON.2018.8478973.

[5] Yee Por L, et al. A systematic literature review on AI-Based methods and challenges in detecting zero-day attacks. IEEE Access 2024;12:144150–63. https://doi.org/10.1109/ACCESS.2024.3455410.

[6] Mbona I, Eloff JHP. Detecting zero-day intrusion attacks using semi-supervised machine learning approaches. IEEE Access 2022;10:69822–38. https://doi.org/10.1109/ACCESS.2022.3187116.

[7] Ahmed U, et al. Signature-based intrusion detection using machine learning and deep learning approaches empowered with fuzzy clustering. Sci Rep Jan. 2025;15(1):1726. https://doi.org/10.1038/s41598-025-85866-7.

[8] Sun H, Huang Y, Han L, Fu C, Liu H, Long X. MTS-DVGAN: anomaly detection in cyber-physical systems using a dual variational generative adversarial network. Comput Secur Apr. 2024;139:103570. https://doi.org/10.1016/j.cose.2023.103570.

[9] Altunay HC, Albayrak Z. A hybrid CNN+LSTM-based intrusion detection system for industrial IoT networks. Eng Sci Technol Int J Feb. 2023;38:101322. https://doi.org/10.1016/j.jestch.2022.101322.

[10] Zainudin A, Ahakonye LAC, Akter R, Kim D-S, Lee J-M. An efficient Hybrid-DNN for DDoS detection and classification in software-defined IIoT networks. IEEE Internet Things J May 2023;10(10):8491–504. https://doi.org/10.1109/JIOT.2022.3196942.
[11] Nguyen TT, Reddi VJ. Deep reinforcement learning for cyber security. IEEE Transact Neural Networks Learn Syst Aug. 2023;34(8):3779–95. https://doi.org/10.1109/TNNLS.2021.3121870.
[12] Ma Y, Li C, Wang Y, Wang Y. Application of deep reinforcement learning algorithms for automatic threat detection and response in dynamic network environments to improve cybersecurity. J Comput Methods Sci Eng May 2025;25(3):2112–25. https://doi.org/10.1177/14727978241309550.
[13] Fährmann D, Jorek N, Damer N, Kirchbuchner F, Kuijper A. Double deep Q-Learning with prioritized experience replay for anomaly detection in smart environments. IEEE Access 2022;10:60836–48. https://doi.org/10.1109/ACCESS.2022.3179720.
[14] Shanmugam V, Razavi-Far R, Hallaji E. Addressing class imbalance in intrusion detection: a comprehensive evaluation of machine learning approaches. Electronics Dec. 2024;14(1):69. https://doi.org/10.3390/electronics14010069.
[15] He M, Wang X, Wei P, Yang L, Teng Y, Lyu R. Reinforcement learning meets network intrusion detection: a transferable and adaptable framework for anomaly behavior identification. IEEE Transactions on Network and Service Management Apr. 2024;21(2):2477–92. https://doi.org/10.1109/TNSM.2024.3352586.
[16] Yang W, Acuto A, Zhou Y, Wojtczak D. A survey for deep reinforcement learning based network intrusion detection. Mar. 02, 2026. https://doi.org/10.48550/arXiv.2410.07612. *arXiv*: arXiv:2410.07612.
[17] Wunder J, Kurtz A, Eichenmüller C, Gassmann F, Benenson Z. Shedding light on CVSS scoring inconsistencies: a user-centric study on evaluating widespread security vulnerabilities. In: 2024 IEEE symposium on security and privacy (SP); May 2024. p. 1102–21. https://doi.org/10.1109/SP54263.2024.00058.
[18] Salim DT, Singh MM, Keikhosrokiani P. A systematic literature review for APT detection and effective cyber situational awareness (ECSA) conceptual model. Heliyon Jul. 2023;9(7):e17156. https://doi.org/10.1016/j.heliyon.2023.e17156.
[19] Tadhani JR, Vekariya V, Sorathiya V, Alshathri S, El-Shafai W. Securing web applications against XSS and SQLi attacks using a novel deep learning approach. Sci Rep Jan. 2024;14(1):1803. https://doi.org/10.1038/s41598-023-48845-4.
[20] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: *2016 IEEE conference on computer vision and pattern recognition (CVPR)*, Las Vegas, NV, USA. IEEE; Jun. 2016. p. 770–8. https://doi.org/10.1109/CVPR.2016.90.