I.J. Wireless and Microwave Technologies, **2026, 5, 158-173**
Published Online on October 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2026.05.10

Resource-Aware Proximal Policy Optimization for Adaptive Intrusion Detection in Dynamic Networks

Kiranjeet Kaur*
Department of Computer Science and Engineering, Chandigarh University, Gharuan, Mohali, Punjab, 140413, India
E-mail: kiranresearch.phd@gmail.com
ORCID iD: https://orcid.org/0009-0007-6813-1364
*Corresponding Author

Jaspreet Singh
Department of Computer Science and Engineering, Chandigarh University, Gharuan, Mohali, Punjab, 140413, India
E-mail: cec.jaspreet@gmail.com
ORCID iD: https://orcid.org/0000-0002-5499-3189

Received: July 3, 2026; Revised: July 18, 2026; Accepted: August 10, 2026; Published: October 8, 2026

Abstract: The growing sophistication of contemporary network infrastructures has increased the pressure on the smart and dynamic intrusion detection systems that can react to the dynamic cyber threats. The machine learning (ML) and deep learning (DL) methods have high classification rates, but they work with fixed decision boundaries which reduces their ability to adapt to dynamic traffic distributions and emerging attack patterns. In order to overcome these issues, the resource-aware Proximal Policy Optimization (PPO)-based adaptive multi-class intrusion detection system (IDS) is suggested in this study. The system characterizes intrusion detection as a sequential decision-making and incorporates computational resource measures into the reinforcement learning (RL) rewarding framework, which allows optimizing detection performance and operational efficiency at the same time. In the ensemble comparison, PPO-Model achieved the highest accuracy (99.4%), recall (98.6%), and Macro AUC (0.998), while reducing CPU utilization by 25.8% and memory consumption by 27.1% compared with the Stacking model. These results demonstrate that the proposed approach can improve detection performance while reducing computational resource requirements. The results suggest that next-generation intrusion detection in the dynamic network environment can be achieved with a scalable and robust solution based on the combination of RL and resource-aware optimization.

Index Terms: Reinforcement Learning, Proximal Policy Optimization, Adaptive Intrusion Detection System, Multi-Class Network Security, Resource-Aware Optimization, Cybersecurity Analytics, Dynamic Network Environments, CIC-IDS2017 Dataset

1. Introduction

The high rate of digital transformation of the contemporary society has created a high dependence on the infrastructure of network, cloud infrastructure and distributed computing platforms. Although this technological change has made scaling and automation possible on an unprecedented scale, it has also made the malicious actors increase their attack surface. DDoS, botnet intrusion, web-based attacks, and advanced persistent threats are just some of the cyberattacks that keep on getting complicated and more common. The concept of the IDS was first suggested to oversee the activities in the network and detect any anomaly or malicious activity based on statistical modeling and rule-driven techniques [1]. These first methods formed the basis of anomaly-based and signature-based detection, but are largely intended to be used in comparatively fixed network setups.

Using large collections of benchmark data like KDD Cup 99 made it possible to systematically assess intrusion detection algorithms [2]. Nevertheless, the subsequent research showed severe shortcomings of these initial data such as redundancy, obsolete attack patterns, and unrealistic traffic distributions. More representative datasets, including CIC-IDS2017, resolved many of these problems by introducing a variety of types of attacks and including modern and realistic flow-based attacks [3]. Following surveys also highlighted that quality of data is an important factor in determining the

This work is open access and licensed under the Creative Commons CC BY 4.0 License.

IDS performance evaluation and generalization abilities [4]. Even with superior data sets, conventional IDS mechanisms continue to be unable to achieve high levels of detection in dynamic and changing network conditions.

In order to address the drawbacks of rule-based detection, machine ML methods are proposed to be able to learn discriminative patterns automatically on the basis of traffic data [5]. Supervised learning models like Support vector machines (SVM), random forests (RF) and gradient boosting models showed a high improvement in accuracy of classification. The performance was further improved by the DL models such as recurrent neural networks and the Long Short-Term Memory (LSTM) architecture, which learned time dependencies in network flows [6]. Despite the high predictive accuracy of these models, they are based on a static training paradigm, i.e., once trained, the decision boundaries of such models do not change until retrained offline. This rigidity may lead to a decline in performance under the changes in the performance that occur in a fast-evolving environment where the distribution of attacks is changing or the emergence of the new threats.

One of the promising model paradigms of adaptive cybersecurity systems is the relatively new RL [7]. Unlike the supervised models where only labeled data is used, RL agents learn the optimal policies by going through a series of interactions with an environment with feedback regarding the reward. This is a dynamic learning process that allows one to keep adapting to changing conditions. Intrusion detection frameworks based on RL have proven to be adaptable in defense strategy and enhance the robustness of detection in adversarial conditions [8]. Nevertheless, the application of the RL in network security is still difficult because of the instability of training, the lack of efficiency in exploration, and computational complexity. The PPO has been a highly popular policy-gradient algorithm that is both stable and efficient in deep RL [9]. PPO presents a clipped surrogate objective function which limits excessive policy updates, which leads to better convergence stability and sample efficiency. This attribute is especially useful when it comes to cybersecurity because the sudden alterations in the traffic distribution can disrupt the learning process. The fact that deep RL has been more successful in general decision-making tasks also promotes its use in studying network security optimization problems [10]. However, currently, the RL-driven IDS solutions are mainly aimed at achieving detection accuracy to the maximum, without taking into account the limitations of computational resources.

In real-life deployment systems IDSs are required to run within limited CPU, memory and energy constraints. DL architectures with high performance tend to be expensive in terms of computation, restricting their scale to real-time or edge-based architectures. Resource cognizant security policies strive to provide a trade-off between detection performance and operational efficiency [11]. Extensive surveys about network anomaly detection have also raised the need to design IDS frameworks that combine performance optimization with computational feasibility [12]. Hence, there is an urgent requirement for adaptive IDSs that are able to strike a balance of great detection accuracy, quick adaptability, and optimality of resources. Driven by such difficulties, this paper presents a resource-aware Proximal Policy Optimization-based adaptive IDS on changing network settings. The system proposed develops intrusion detection as a Markov Decision Process, where a PPO agent learns to maximize a detection policy, with respect to the correctness of classification as well as computational cost. The framework allows the smart trade-offs between the depth of detection and the use of resources by adding real-time CPU and memory data to the state representation and reward function.

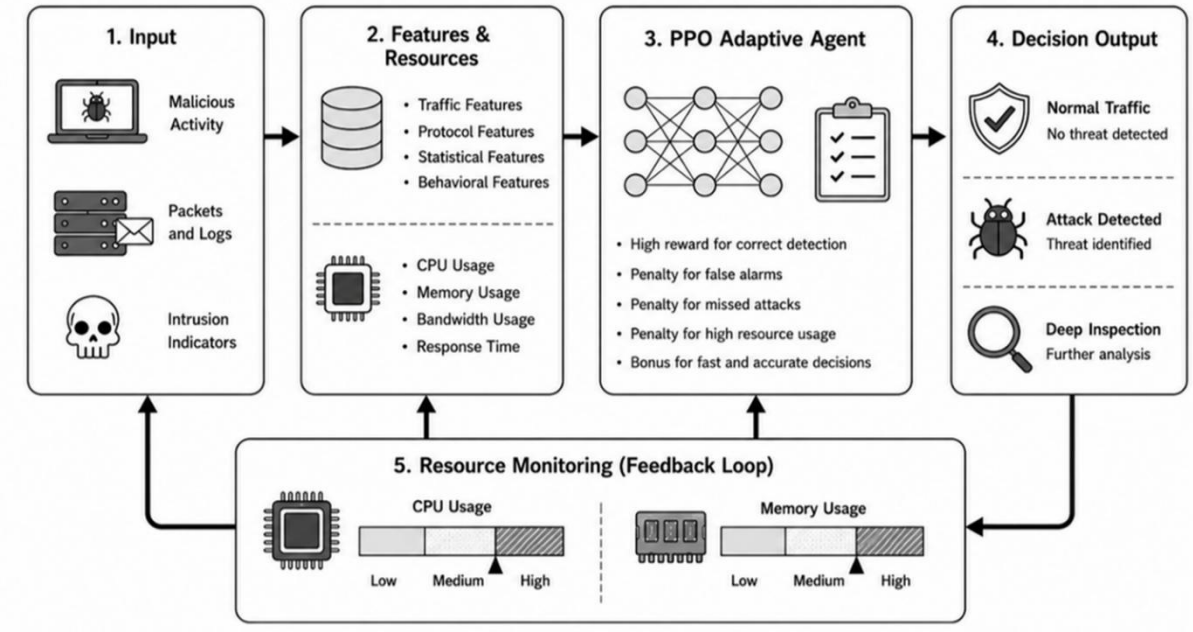


Fig. 1. Proposed adaptive PPO-based IDS framework.

Fig. 1 shows the general structure of the proposed adaptive IDS framework. The incoming network traffic is converted into the structured feature representations that are augmented with metrics of system resources. These characteristics are inputted to an adaptive agent based on PPO, which continuously resets its policy based on reward signals as a result of detection accuracy as well as computational cost. The resource monitoring module dynamically calculates CPU and memory usage and makes sure that the update of a policy is sensitive to the operational limits. The decision output module takes the multi-class intrusion predictions and as required invokes more extensive inspection mechanisms. This architecture incorporates a high detection performance in this system and is able to work efficiently in a resource-bound and dynamically changing network environment. The three major contributions of this work are as follows. First, the intrusion detection task is formulated as a multi-objective reinforcement learning problem that jointly considers detection accuracy and computational efficiency, addressing the limitations of approaches that focus primarily on detection performance. Second, a resource-aware reward modeling strategy is developed to incorporate resource-related factors, including CPU utilization, memory consumption, inference time, and energy requirements, into the PPO learning process, enabling the agent to learn adaptive and resource-efficient detection policies. Third, the proposed framework is comprehensively evaluated under diverse experimental conditions, including original and balanced datasets, ensemble-based configurations, and dynamically changing attack distributions, to examine its detection capability, adaptability, and resource efficiency. Overall, the proposed PPO-based adaptive IDS provides a scalable and resource-conscious security framework that seeks to balance effective intrusion detection with the computational requirements of modern network infrastructures.

2. Literature Survey

Advancement of research of intrusion detection has been in the evolutionary path of successively shifting to the use of intelligent and adaptive learning systems as opposed to the use of the simple anomaly detection systems. Recent progress in the representation learning field has greatly enhanced the capability of feature abstraction in multifaceted network settings, such that models are able to learn nonlinear structure of high-dimensional traffic data [13]. Although these methods are more accurate in detection, the methods are mostly used in supervised learning paradigms and are not inherently adaptable to distributional changes. The RL is presented as the promising alternative to the stagnant classification frameworks. Reinforcement-based adaptive traffic filtering strategies have been shown to be able to dynamically adapt defence policies according to perceived states of the environment [14]. The fact that adversarial evaluation systems such as generative adversarial networks that wield AI to evade IDS models expose the vulnerability of fixed-decision-boundary systems is also exposed [24]. These results point to the need to have adaptive security systems that can adapt to the changing attack modes.

Deep learning (DL) techniques have been applied to intrusion detection in many more domains, where detection accuracy must be achieved with constrained computational assets. Hybrid convolutional–recurrent architectures have demonstrated enhanced anomaly detection by opting for both spatial and temporal features of network traffic [15]. DL approaches have further robustness through statistical filtering and traffic representation techniques [16]. However, online DL models developed for streaming traffic generally focus on continuous data processing but lack an explicit policy-learning mechanism for adapting detection [17]. Research on IDS datasets has also shown that imbalance, attack diversity, and feature representation strongly influence detection performance and generalization [18]. Moreover, anomaly detection studies indicate that improvements in predictive performance are often accompanied by increased computational complexity [19]. Similarly, ensemble-based approaches can achieve stronger classification results by combining multiple learners, but the additional models may increase inference latency and resource consumption. [21, 22].

Recent IDS studies reveal limitations related to adaptability and computational efficiency. Botnet detection research highlights the hurdles of handling heterogeneous attack distributions and fluctuation in traffic features across datasets [23]. Hybrid ensemble methods have enhanced anomaly detection in industrial control systems, but computational cost becomes high during detection [14]. Similarly, ensemble IDS solutions for smart-city networks can provide high detection accuracy while creating challenges related to scalability and resource consumption [24]. RL-based IDS prototypes have demonstrated adaptive detection capabilities, but their evaluation has generally emphasized classification performance rather than incorporating computational cost into the policy-learning objective [25]. Hybrid ML/DL IDS approaches can further improve predictive performance through model coordination and stacking, although their increased model complexity may introduce additional computational requirements [26]. AI-based security solutions for IoT environments have addressed lightweight deployment to some extent, but explicit policy optimization for adapting detection decisions remains limited [27].

Recent reinforcement learning papers have explicitly included computational costs into the process of policy optimization. Resource-constrained exploration methods make use of the residual resources in controlling actions such that the agent does not execute computationally demanding tasks [28]. Likewise, deep-RL-based approaches to resource scheduling formulate multi-objective reward functions taking into account energy efficiency, processing time, load balancing, and resource efficiency [29]. Work on constrained optimal control also proves that reward functions can include performance incentives as well as punishment for violating constraints and high cost of operation [30]. All of these works prove the theoretical possibility of designing resource-aware reward functions since computational cost may

affect the process of policy updates. Nevertheless, these methods were applied to generic resource management problems, edge computing, and constrained control but not to intrusion detection systems based on several classes of malicious activities. Thus, there is a lack of studies on designing the reward function of such IDS.

Table 1. Literature Summary with TRL levels

Ref	Approach Type	Application Domain	Adaptability	Resource Awareness	TRL Level
[13]	Representation Learning	General ML	Low	No	3
[14]	RL Traffic Filtering	Network Security	Medium	Partial	4
[24]	GAN-based Adversarial IDS	IDS Evaluation	Low	No	3
[15]	DL (IoT)	IoT Security	Low	Limited	4
[16]	Feature-Engineered DL	IDS	Low	No	4
[17]	Online Deep IDS	Streaming Networks	Medium	No	5
[18]	Dataset Taxonomy	IDS Evaluation	N/A	No	2
[19]	Anomaly Detection Survey	General IDS	N/A	No	2
[11]	RL Malware Defense	Cloud Security	High	Limited	5
[20]	RL Theory	General AI	N/A	No	2
[21]	Deep RL Survey	General AI	N/A	No	2
[22]	Deep RL Overview	General AI	N/A	No	2
[23]	Botnet Detection	Network IDS	Low	No	4
[14]	DL + Ensemble (ICS)	Industrial Systems	Low	Limited	5
[24]	Ensemble IDS	Smart Cities	Low	No	5
[25]	RL-based IDS Prototype	Network Security	Medium	No	4
[26]	Hybrid ML/DL IDS	Network Security	Low	No	5
[27]	AI-based IoT Security	IoT Networks	Low	Limited	4
[28]	Resource-Restricted RL	General RL	High	Yes	4
[29]	DRL-Based Resource Scheduling	SDN-Driven Edge Computing	High	Yes	5
[30]	Constrained RL with Interpretable Reward Design	Optimal Control Systems	High	Yes	3

Table 1 is used to give a structured maturity rating of current strategies by way of Technology Readiness Level (TRL) analysis of references [11, 13–30]. The comparison shows that a majority of RL methods and hybrid IDS systems are in laboratory validation phases (TRL 3-5), and few of them are implementations ready to be deployed. Moreover, the explicit consideration of the computational resource awareness into adaptive learning is not extensively studied.

Therefore, the central research gap is the absence of a unified resource-aware reinforcement-learning intrusion detection system that jointly optimizes detection effectiveness, adaptability, and computational efficiency. Current studies do not adequately determine whether the expected security benefit of a detection action justifies its processing time, memory consumption, feature-evaluation cost, or energy requirement. This limitation is particularly important for real-time IoT, edge, and smart-network environments in which computational resources are limited and traffic conditions continuously change.

3. Methodology

Building upon the limitations identified in the preceding literature survey, this section introduces the proposed resource-aware PPO-based adaptive intrusion detection framework. The current IDS approaches are rather rigid, relying on static supervised learning models that are not flexible enough and explicitly do not take into account the computational efficiency of their decision making process. With a view to eliminating these limitations, intrusion detection is redefined as a sequential multi-objective optimization problem within the framework of a RL concept. The entire methodological pathway is demonstrated in Fig. 2, which depicts network traffic processing flow, state building, policy learning, reward assessment, and adaptive decision output. The suggested framework combines traffic characteristics with system resource measures, which allows optimizing the accuracy of detection and efficiency in the functioning dynamically.

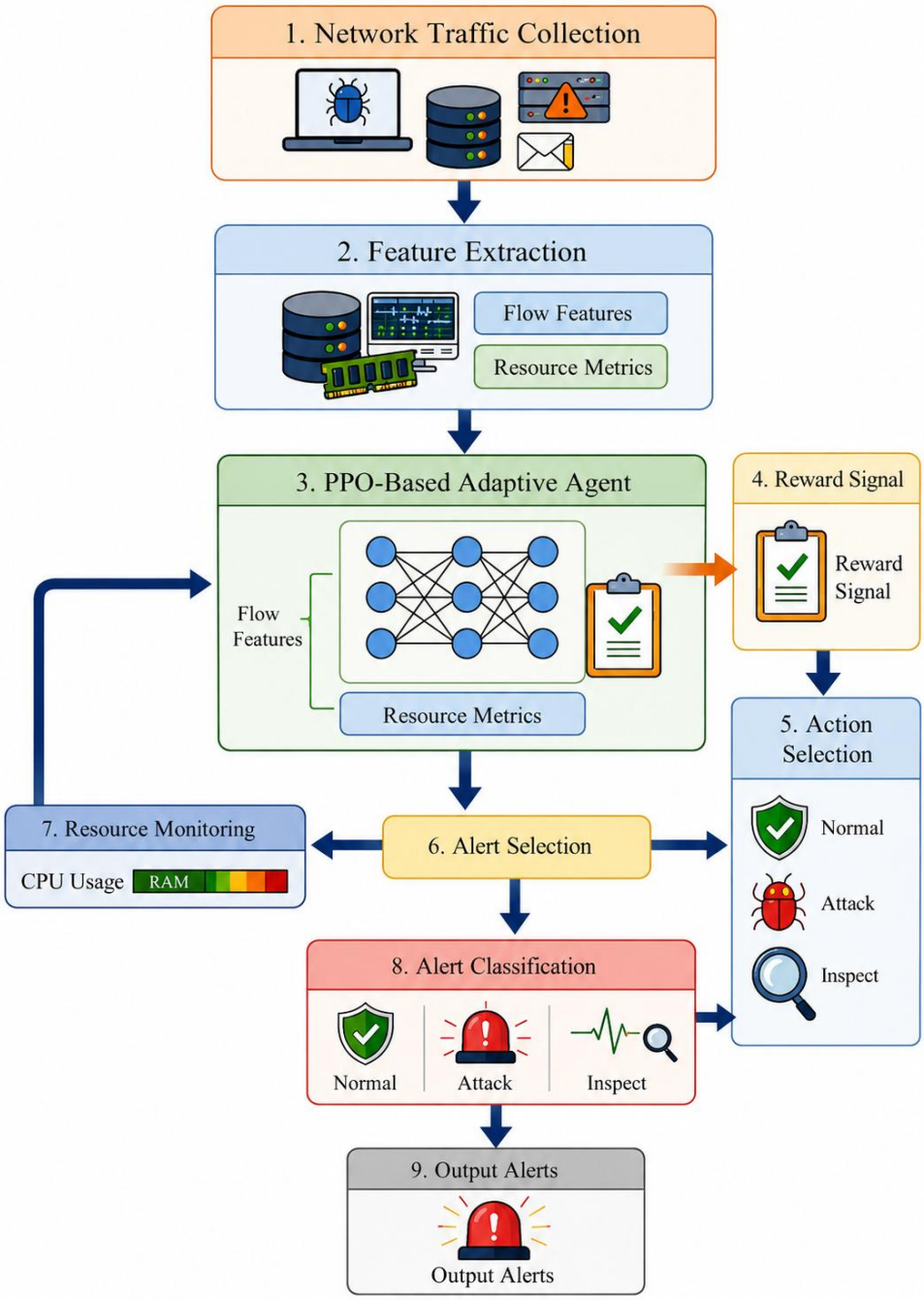


Fig. 2. Flowchart of the resource aware PPO-based adaptive IDS

3.1. Problem Formulation as a Markov Decision Process

The current network environments are dynamic in traffic distributions and manifest diverse patterns of attacks. Therefore, the intrusion detection environment is defined as a discrete-time system in which network traffic instances enter sequentially at successive time steps. Its goal is to categorize every traffic instance in to one of the K attack categories at minimum computational cost.

The detection task is modeled as a Markov Decision Process (MDP):

$$M = (S, A, P, R. \gamma) \tag{1}$$

Where:

- S represents the state space
- A denotes the action space

- $P(\acute{s}\,|s,a)$ is the transition probability
- $R(s,a)$ is the reward function
- $\gamma\epsilon(0,1)$ is the discount factor

The goal is to learn a policy $\pi(a|s)$ that maximizes expected cumulative reward:

$$J(\pi) = E[\sum_{t=0}^{T} \gamma^t R_t] \tag{2}$$

This formulation enables joint optimization of detection performance and resource utilization.

3.2. Dataset Preparation and Sequential Traffic Modelling

To measure the proposed framework, the CIC-IDS2017 data set is applied. The flows are normalized, processed to handle null values, and encoded into eight attack categories beforehand.

To preserve temporal dependencies, traffic samples are organized into sequential windows:

$$W_t = \{x_t, x_{t+1} \dots \dots \dots \dots \dots x_{t+n} \tag{3}$$

Where n denotes window size. This sequential modeling allows the PPO agent to capture evolving traffic behavior rather than relying on independent instance classification.

3.3. State Representation

The state representation integrates network features and system resource metrics. Each state s_t is defined as:

$$s_t = [f_t, h_t, r_t] \tag{4}$$

Where:

- f_t represents normalized flow-based statistical features,
- h_t captures historical aggregation statistics,
- r_t includes resource metrics such as CPU and memory usage.

Measures of resource are integrated so as to have decision learning based on cost. The agent acquires adaptive policies to system conditions by integrating operational constraints into the state.

3.4. Action Space Design

The action space includes both classification and inspection control decisions:

$$A = \{a_1, a_2, \dots \dots \dots \dots \dots \dots a_{k,} a_{light,} a_{deep}\} \tag{5}$$

Where:

- $a_1, \dots \dots, a_k$ correspond to multi-class labels,
- a_{light} triggers lightweight inspection,
- a_{deep} activates deeper feature analysis.

This hybrid action design enables adaptive computational allocation based on traffic complexity and resource availability.

3.5. PPO-Based Policy Optimization

PPO is employed due to its stable gradient updates and high sample efficiency. The proposed framework adopts an actor–critic architecture, in which the actor determines the action to be taken for a given network state, while the critic estimates the expected value of that state. The actor and critic are represented as:

- Actor network: $\pi(a|s)$
- Critic network: $V_\phi(s)$

The clipped surrogate function is used to bound large policy updates and maintain stable learning:

$$L^{CLIP}(\emptyset) = E^t[min(r_t(\theta)a_t, clip(r_t(\theta)\ ,1-\varepsilon, 1+\epsilon)a_t)] \tag{6}$$

Where probability ratio between updated and previous policy is calculated as:

$$r_t(\theta) = \frac{\pi_\theta(a_t/s_t)}{\pi_{\theta old}(a_t/s_t)} \tag{7}$$

And A_t is the advantage estimate:

$$A_t = R_t - V_\emptyset(s_t) \tag{8}$$

The clipping parameter ϵ prevents large policy deviations, ensuring stable convergence under dynamic traffic fluctuations. This constraint prevents excessively large policy changes and promotes stable convergence when the network traffic distribution and attack patterns change dynamically.

3.6 Resource-Aware Reward Modelling

To balance detection performance with computational efficiency, a multi-objective reward function is defined as:

$$R_t = \alpha D_t - \beta D_t \tag{9}$$

Where denotes detection correctness:

$$D_t = \begin{Bmatrix} +1 \ \ correct\ classificatio \\ -1 \ \ missclassification \end{Bmatrix} \tag{10}$$

And computational cost is modelled as:

$$C_t = \lambda_1 . CPU_t + \lambda_2 Memory_t \tag{11}$$

Parameters α and β regulate the trade-off between accuracy and efficiency. This design ensures that the agent does not simply maximize classification accuracy but learns to minimize unnecessary computational expenditure.

3.7. Training Strategy and Algorithmic Implementation

The training process is an iterative process of the interaction of the PPO agent and the network environment. The agent views the current state at every time step, chooses an action based on the stochastic policy, gets reward feedback which combines both the classification accuracy and the computational cost, and adjusts policy parameters based on clipped surrogate optimization The environment then provides the detection result and the current CPU and memory usage. These values are used to calculate the reward and update the PPO policy.

Algorithm: Resource-Aware PPO-Based Adaptive Intrusion Detection

Input: Sequential traffic windows $W_t \in R^{32\times d}$, where W is the number of traffic records in single slot and d is the number of input traffic features. The value of w = [29] which represents the number of consecutive traffic records in each window.

Output: Optimized policy $\pi_\theta(a|s)$.

1. Initialize actor parameters θ and critic parameters ϕ.
2. Divide network traffic into sequential windows: $W_t \in R^{32\times d}$
3. Observe initial state

$$s_t = [f_t \| h_t \| r_t\ ]$$
$$f_t \in R^{df}, h_t \in R^{dh}, r_t \in R^{dr} \tag{12}$$

Where f_t represents present traffic features, h_t represents present past detection and r_t holds normalized CPU and memory holds.

$$f_t \in R^{d_f + d_h + d_r}$$

4. For every training episode:

- Note the starting state s_0
- For each time step $t = 0$ to T do:
- Choose action based to
$a_t \sim \pi_\theta(a_t|s_t)$.
- Apply A_t to the current traffic window and obtain the detection outcome D_t.
- Measure computational metrics $CPU_t, Memory_t$.
- Compute reward:

$$R_t = \alpha D_t - \beta(\lambda_1 CPU + \lambda_2 Memory) \tag{13}$$

- Store transition (s_t, a_t, R_t, s_{t+1})
5. Compute advantage estimates $A_t = R_t + \gamma V\phi(s_{t+1}) - V_\emptyset(s_t)$.
6. Calculate probability ratio:

$$r_t(\theta) = \frac{\pi_\theta(a_t/s_t)}{\pi_{\theta old}(a_t/s_t)} \tag{14}$$

7. Update actor by minimizing value loss.

$$L^{CLIP}(\emptyset) = E^t[min(r_t(\theta)a_t, clip(r_t(\theta), 1-\varepsilon, 1+\epsilon)a_t)] \tag{15}$$

8. Update critic by minimizing

$$L_v(\phi) = E_t\left[(V_\emptyset(s_t) - R_t^{target})2\right] \tag{16}$$

Continuous until episode complete
9. Return optimized policy $\pi_\theta(a \mid s)$.

Hyperparameter Configuration

The training process employs the following settings:

- Learning rate: 3×10^{-4}
- Discount factor: $\lambda = 0.99$
- Clipping parameter: $\epsilon = 0.2$
- Batch size: 1024
- Total adaptation epochs: 20

These parameters are selected based on empirical stability and convergence analysis.

3.8. Computational Complexity Analysis

The complexity of PPO updates depends on rollout length T, feature dimension d, and hidden layer size H:

$$O(T.d.H)$$

Although training complexity increases compared to static models, inference remains lightweight due to the compact actor architecture.

4. Results and Analysis

The section will involve a detailed analysis of the offered resource-conscious PPO-based adaptive intrusion detection model. The progress of analysis starts with the baseline detection performance up to the robustness validation, ensemble effectiveness, computational efficiency, adaptability on dynamic conditions, and ablation validation. The subsections combine quantitative data and graphical explanation to give an analytical narrative.

4.1. Performance on Balanced Dataset

In order to test the strength against class imbalance, experiments are again performed on a balanced dataset setup. In this subsection, the determinations of the performance improvements are evaluated as a distribution-dependent phenomenon or inherent with the adaptive framework.

Table 2. Network Class Detection Performance on Balanced Dataset

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	Macro AUC
LR	90.5	90.1	89.4	89.7	0.91
SVM (Linear)	92.3	91.8	91.4	91.6	0.93
RF	96.2	95.8	95.3	95.5	0.96
XGBoost	97.1	96.8	96.4	96.6	0.97
MLP	97.6	97.2	96.9	97.0	0.98
BiLSTM	98.0	97.7	97.5	97.6	0.99
PPO IDS	**98.6**	**98.3**	**98.1**	**98.2**	**0.993**

As can be seen in Table 2, the minority attack categories respond to improvement of recall values in all the models. Nonetheless, the traditional models achieve moderate performance, whereas PPO-Adaptive IDS has 98.6 accuracy and 0.993 macro-AUC. The relatively small difference between the results obtained on the imbalanced and balanced datasets indicates that the proposed framework maintains stable performance under different class-distribution conditions.

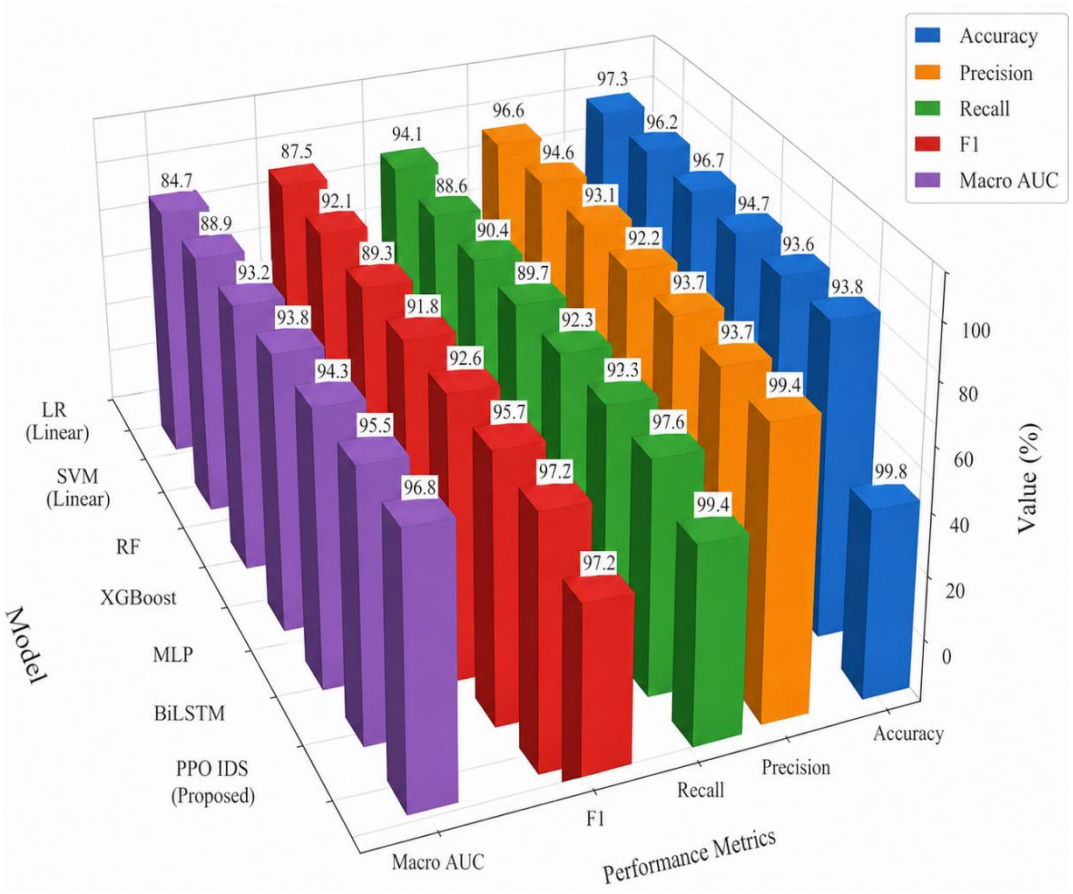


Fig. 3. Bar graph detection performance across models on the balanced CIC-IDS2017 dataset.

Fig. 3. presents the bar graph that indicates a small number of off-diagonal misclassifications of such rare attack classes as Infiltration and Heartbleed. This shows that the adaptive reward mechanism is effective in minimizing the bias and does not fit synthetic balancing. The agreement of Table 2 with Fig. 3. shows that the framework is reliable when being applied in different distribution conditions.

4.2. Ensemble and Hybrid Model Evaluation

Ensemble learning is very much embraced to improve the strength of classification but most of the time encourages the rise of computational cost. In this subsection, the question to be examined is whether RL can be used to direct ensemble coordination more effectively.

Table 3. Ensemble and Hybrid Model Comparison

Model	Accuracy (%)	Recall (%)	Macro AUC	CPU (%)	Memory (MB)
Voting (RF + XGB)	98.7	97.4	0.992	89	610
Stacking (RF + XGB + MLP)	98.9	97.8	0.994	93	720
PPO-Model	**99.4**	**98.6**	**0.998**	69	525

Table 3 shows that, compared with the Stacking model, PPO-Model reduced CPU utilization from 93% to 69%, representing a 25.8% reduction, while memory consumption decreased from 720 MB to 525 MB, corresponding to a 27.1% reduction.

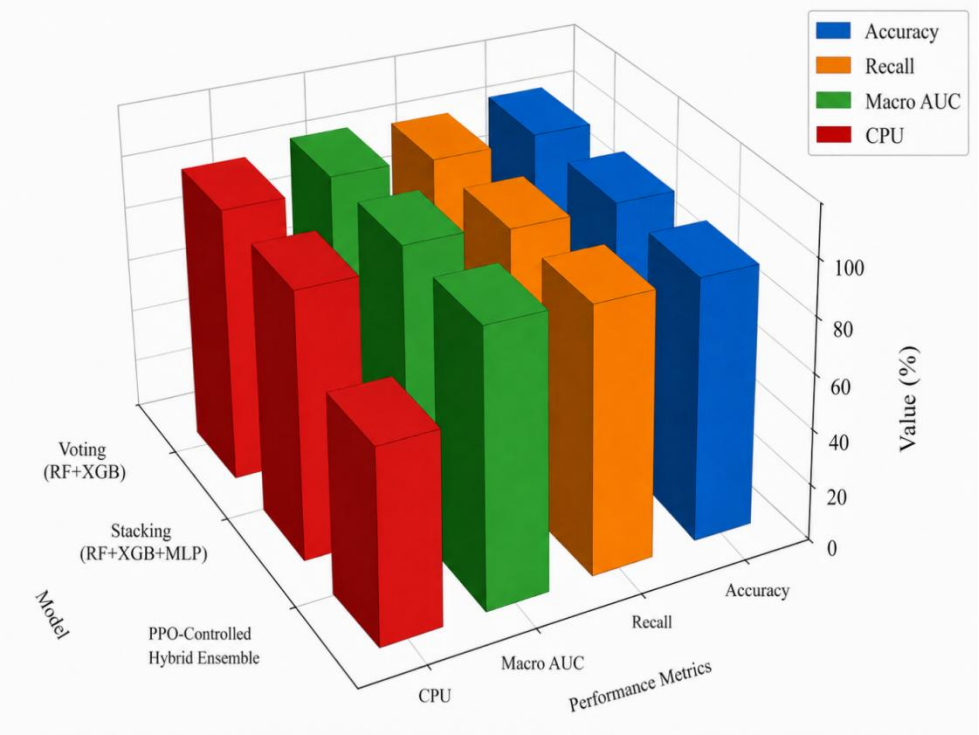


Fig. 4. Performance comparison of different ensemble models.

In Fig. 4., the performance-efficiency trade-off, it can be seen that the PPO-controlled ensemble is on the optimal frontier region where it has the maximum performance with moderate resource usage. The consistency between Table 3 and Fig. 4 proves that hybridization guided by policy is higher than the integration of the static ensembles.

4.3 Computational Resource Utilization

In addition to classification measures, practical implementation needs an operation that is resource-efficient. In this subsection, the computational overhead in models is evaluated.

Table 4. Computational Resource Utilization

Model	CPU (%)	Memory (MB)	Inference Time (ms)	Energy (J)
SVM [31]	66	420	17	52
RF[32]	82	505	19	70
XGBoost[33]	85	530	21	73
MLP[34]	90	750	24	80
BiLSTM[35]	94	880	29	86
PPO-Adaptive IDS	61	460	16	48

Table 4 indicates that the DL models have considerably increased CPU and memory usage. The PPO-Adaptive IDS has an almost deep-learning performance of 61 % CPU and 460 MB of memory, which is a performance of efficient decision policy learning.

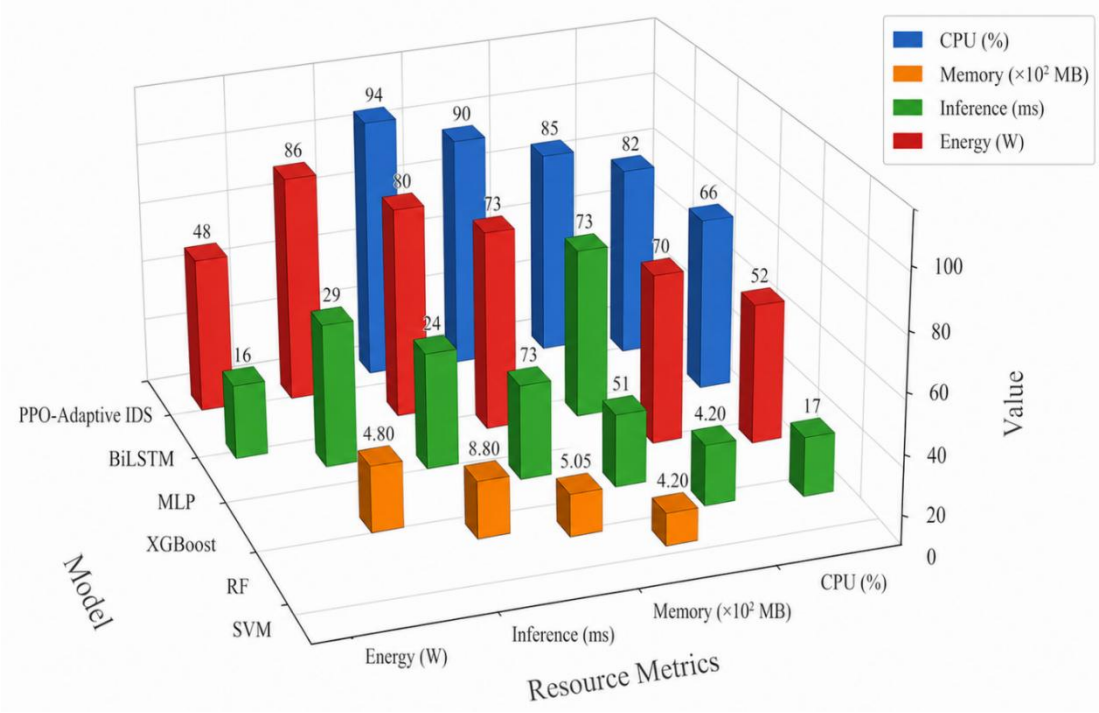


Fig. 5. Inference time comparison highlighting computational efficiency.

This conclusion is further supported by the comparison of the inference time in Fig. 5. The adaptive framework is also faster in detecting than BiLSTM and has similar performance. The combined Table 4 and Fig. 5. evidence helps to confirm that the resource-conscious model of rewards effectively strikes the balance between performance and efficiency.

4.4. Adaptability Under Dynamic Traffic Shift

Dynamic attack distributions are common in real network environments. This subsection evaluates model resilience when minority attack frequency abruptly increases.

Table 5. Adaptability under Dynamic Traffic Shift

Model	Initial Accuracy (%)	After Shift (%)	Degradation (%)	Recovery Epochs
RF	97.1	89.4	-7.7	0
XGBoost	97.9	90.8	-7.1	0
MLP	98.2	91.5	-6.7	0
BiLSTM	98.5	92.3	-6.2	0
PPO-Adaptive IDS	99.1	96.5	-2.6	3

Table 5 has the revised results with the epoch data in the process of adaptation. The static models lose much of their performance (up to 7.7%), whereas PPO-Adaptive IDS drops by only 2.6 percentage points and recovers in 3 epochs in a 20-epoch adaptation period. This means that it is a fast-adjusting policy.

The adaption curve as shown in Fig. 6. shows PPO-based model has a rapid recovery curve but the static models have progressive decline in performance. Table 5 and Fig. 6. are consistent, which justifies the fact that RL allows real-time flexibility, which reinforces the viability of deployment in the changing cyber environments.

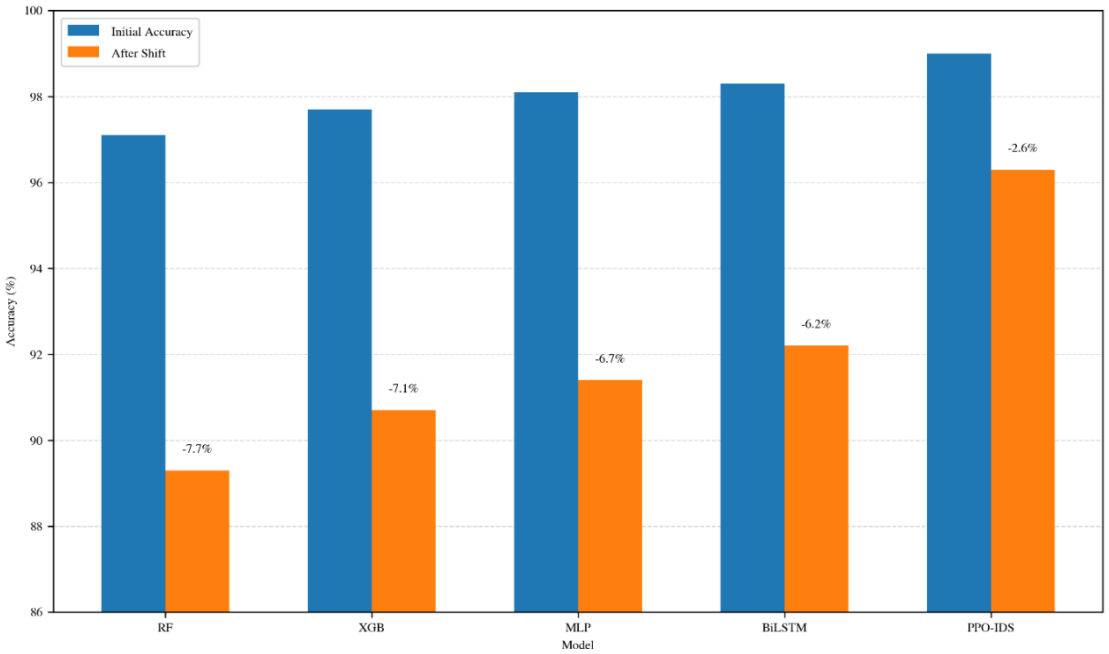


Fig. 6. Adaptation performance under dynamic attack distribution shift.

4.5. Ablation Study

An ablation study is conducted to evaluate the individual contributions of the resource-aware reward function, inspection-control mechanism, and PPO algorithm. The complete resource-aware PPO model jointly optimizes intrusion-detection performance and computational efficiency through the following reward function:

$$R_t = \alpha R_t^{ret} - \beta C_t^{res} - \gamma P_t^{fp} \tag{17}$$

Where,

- R_t^{ret} represents the detection reward,
- C_t^{res} denotes the normalized computational resource cost,
- P_t^{fp} represents the false-positive penalty,
- $\alpha, \beta, and\ \gamma$ control their relative contributions.

Table 6 presents the ablation results. Removing the resource penalty upgrades accuracy from 99.1% to 99.3%, recall from 97.6% to 97.8%, and Macro AUC from 0.995 to 0.996. However, this small improvement in detection performance is accompanied by higher resource consumption. This result indicates that removing the resource constraint allows the agent to select computationally expensive inspection actions more frequently.

Table 6. Ablation Study of the Resource-Aware PPO Framework

Configuration	Accuracy (%)	Recall (%)	Macro AUC	Resource Consumption
Resource-Aware PPO Model	99.1	97.6	0.995	Controlled
Without Resource Penalty	99.3	97.8	0.996	Higher
Without Inspection Control	98.4	95.9	0.989	Higher
DQN	98.2	95.6	0.987	Moderate–High

The slightly higher accuracy obtained without the resource penalty does not indicate a superior overall configuration. Instead, it shows that unconstrained optimization favours detection performance while disregarding computational expenditure. The full resource-aware PPO model provides a more practical balance by maintaining competitive predictive performance while limiting unnecessary inspection operations and resource usage. The performance reduction observed without inspection control confirms that selective traffic analysis is essential for prioritizing suspicious instances. Similarly, the lower performance of DQN indicates that PPO provides more stable policy updates and is better suited to the continuous optimization requirements of the proposed resource-aware RL-IDS.

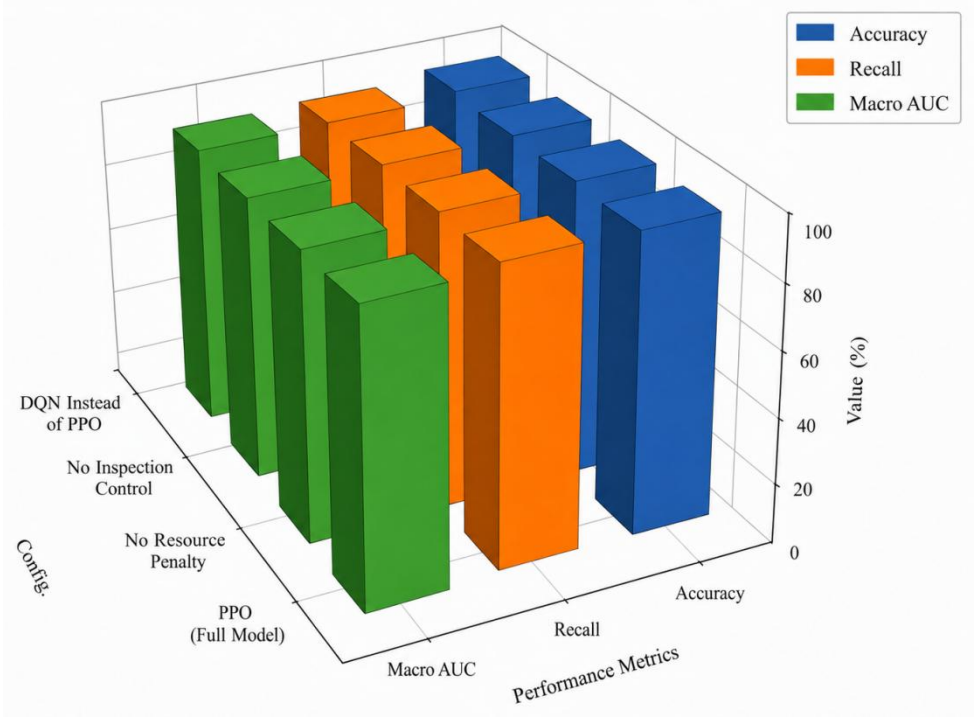


Fig. 7. Ablation analysis illustrating impact of architectural components.

The trend of performance sensitivity diagrammed in Fig. 7 also indicates how the PPO-based architecture can be stable at different weight of rewards.

5. Conclusion

This paper introduces a resource-conscious PPO-based adaptive IDS to be used in a network problem under a dynamic network setup. In contrast to the traditional models of ML and DL which are usually considered to be non-sequential in time, the offered system presents intrusion detection as a sequence of decisions and incorporates computational efficiency as the learning objective. The framework optimally balances the detection performance and the cost of operation by modifying the state representation and incorporates CPU and memory use in the reward modeling. The experimental analysis of CIC-IDS2017 data in both original and balanced settings has shown that the PPO-IDS achieved strong detection performance while considering resource efficiency such as usage of CPU and memory as comparative to traditional ML, DL, and ensemble baselines. Additionally, the experiments of dynamic traffic shifts prove the adaptability, which indicates the benefit of learning with policy-gradient behavior in moving cybersecurity conditions. The ablation experiment is another experimental finding that supports the role of resource-conscious reward design and inspection control systems. Practically, the suggested framework can be effectively deployed in real-time and resource-constrained environments such as edge networks, IoT-based systems, and cloud-based security surveillance systems. The system enables the system to overcome critical scalability concerns related to deep learning-based IDS implementations because the system is computationally less expensive without compromising detection effectiveness.

The proposed resource-aware PPO-IDS has several limitations. Its performance depends on careful tuning of reward weights that balance detection accuracy, false alarms, and computational cost. Fixed reward coefficients may not remain optimal under highly volatile traffic or extreme resource constraints. The dynamic traffic-shift experiments also represent controlled conditions and may not fully capture the complexity of live networks. Furthermore, validation on CIC-IDS2017 does not guarantee reliable detection of previously unseen attacks or consistent cross-dataset generalization.

Future work will introduce meta-learning or adaptive reward-weight mechanisms to automatically adjust optimization priorities according to traffic and resource conditions. Federated reinforcement learning will be investigated for distributed monitoring without centralizing sensitive traffic data. Adversarial training and uncertainty estimation will be incorporated to improve resistance to evasion and unknown attacks. Explainable AI modules will be used to interpret policy actions and resource-allocation decisions. Large-scale evaluation on live traffic streams, edge devices, and multiple IDS datasets will further establish deployment feasibility, convergence stability, and generalization. These developments may support more autonomous, transparent, and resilient intrusion-detection systems without claiming complete resistance to cyber-attacks.

All the Declarations and Statements

Author Contribution Statement
The author has read and agreed to the published version of the manuscript.

Conflict of Interest Statement
The authors declare no conflicts of interest.

Funding Declaration
This research received no external funding.

Data Availability Statement
Data sharing is not applicable.

Ethical Declarations
This study does not involve human participants or animals.

Acknowledgements
We sincerely thank the experts for their professional evaluation and valuable recommendations, which have contributed to improving the quality of the experiment and the reliability of its results.

Declaration of Generative AI in Scholarly Writing
The authors declare that generative AI tools were used solely to assist in language refinement, grammar correction during the manuscript preparation process. The use of such tools did not influence the research design, methodology, simulation, analysis, or interpretation of results. All technical content, including the development of the RL-PPO framework, experimental evaluation, and conclusions, was carried out by the authors.

Abbreviations
The following abbreviations are used in this manuscript:

RL - Reinforcement Learning
PPO - Proximal Policy Optimization
MDP - Markov Decision Process
IDS - Intrusion Detection System
SDN - Software Defined Network

Appendix A/B/C…, with appendix title
None.

References

[1] D. E. Denning, “An intrusion-detection model,” *IEEE Transactions on Software Engineering*, vol. 13, no. 2, pp. 222–232, 1987.
[2] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009, July). A detailed analysis of the KDD CUP 99 data set. In 2009 IEEE symposium on computational intelligence for security and defense applications (pp. 1-6).
[3] Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. ICISSp, 1(2018), 108-116.
[4] M. Ring et al., “A survey of network-based intrusion detection data sets,” *Computers & Security*, vol. 86, 2019.
[5] A. Javaid et al., “A deep learning approach for network intrusion detection system,” in *Proc. EAI Bio-inspired ICT*, 2016.
[6] J. Kim et al., “Long short-term memory recurrent neural network classifier for intrusion detection,” in Proc. PlatCon, 2016.
[7] S.-G. Choi, S.-B. Cho, "Adaptive Database Intrusion Detection Using Evolutionary Reinforcement Learning," in Proc. Int. Joint Conf. SOCO'17-CISIS'17-ICEUTE'17 (Advances in Intelligent Systems and Computing), León, Spain, 2017, pp. 547-556, doi: 10.1007/978-3-319-67180-2_53.
[8] R. R. dos Santos, E. K. Viegas, A. O. Santin, and V. V. Cogo, “A long-lasting reinforcement learning intrusion detection model,” in Proc. 34th Int. Conf. Advanced Information Networking and Applications (AINA), Caserta, Italy, 2020, pp. 1437–1448, doi: 10.1007/978-3-030-44041-1_121.
[9] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” arXiv preprint arXiv:1707.06347, 2017.
[10] V. Mnih et al., “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, 2015.
[11] L. Xiao et al., “Cloud-based malware detection game for mobile devices with offloading,” IEEE Transactions on Mobile Computing, vol. 16, no. 10, pp. 2742–2750, 2017.

[12] M. H. Bhuyan et al., "Network anomaly detection: Methods, systems and tools," *IEEE Communications Surveys & Tutorials*, vol. 16, 2014.
[13] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
[14] R. R. dos Santos, E. K. Viegas, A. O. Santin, and V. V. Cogo, "Reinforcement learning for intrusion detection: More model longness and fewer updates," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 2, pp. 2040–2055, Jun. 2023, doi: 10.1109/TNSM.2022.3207094.
[15] M. Al-Hawawreh, N. Moustafa, and E. Sitnikova, "Identification of malicious activities in industrial internet of things based on deep learning models," *Journal of Information Security and Applications*, vol. 41, pp. 1–11, 2018.
[16] S. M. Kasongo and Y. Sun, "A deep learning method with filter-based feature engineering for wireless intrusion detection system," *IEEE Access*, vol. 7, pp. 38597–38607, 2019.
[17] A. Alrawashdeh and C. Purdy, "Toward an online anomaly intrusion detection system based on deep learning," in Proc. IEEE ICMLA, 2016, pp. 195–200.
[18] H. Hindy et al., "A taxonomy of network threats and the effect of current datasets on intrusion detection systems," *IEEE Access*, vol. 8, pp. 104650–104675, 2020.
[19] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network anomaly detection: Methods, systems and tools," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 303–336, 2014.
[20] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2018.
[21] K. Arulkumaran, M. Deisenroth, M. Brundage, and A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.
[22] Y. Li, "Deep reinforcement learning: An overview," *arXiv preprint arXiv:1701.07274*, 2017.
[23] S. Garcia et al., "An empirical comparison of botnet detection methods," *Computers & Security*, vol. 45, pp. 100–123, 2014.
[24] Z. Li, C. Huang, S. Deng, W. Qiu, and X. Gao, "A soft actor-critic reinforcement learning algorithm for network intrusion detection," *Comput. Secur.*, vol. 135, Art. no. 103502, Dec. 2023, doi: 10.1016/j.cose.2023. 103502.
[25] M. A. Merzouk, C. Neal, J. Delas, *et al.*, "Adversarial robustness of deep reinforcement learning-based intrusion detection," *Int. J. Inf. Secur.*, vol. 23, pp. 3625–3651, Dec. 2024, doi: 10.1007/s10207-024-00903-2
[26] Y.-D. Lin, H.-X. Huang, D. Sudyana, and Y.-C. Lai, "AI for AI-based intrusion detection as a service: Reinforcement learning to configure models, tasks, and capacities," *J. Netw. Comput. Appl.*, vol. 229, Art. no. 103936, 2024, doi: 10.1016/j.jnca.2024.103936.
[27] S. Jamshidi, A. Nikanjam, K. W. Nafi, F. Khomh, and R. Rasta, "Application of deep reinforcement learning for intrusion detection in Internet of Things: A systematic review," *Internet Things*, vol. 31, Art. no. 101531, 2025, doi: 10.1016/j.iot.2025.101531
[28] Z. Wang, T. Pan, Q. Zhou, and J. Wang, "Efficient exploration in resource-restricted reinforcement learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 8, pp. 10279–10287, 2023, doi: 10.1609/aaai.v37i8.26224.
[29] M. A. Hossain, "Deep Q-learning intrusion detection system (DQ-IDS): A novel reinforcement learning approach for adaptive and self-learning cybersecurity," *ICT Express*, vol. 11, no. 5, pp. 875–880, Oct. 2025, doi: 10.1016/j.icte.2025.05.007.
[30] E. Iturbe, A. Rego, O. Llorente-Vazquez, E. Rios, C. Dalamagkas, D. Merkouris, and N. Toledo, "Reinforcement Learning in action: Powering intelligent intrusion responses to advanced cyber threats in realistic scenarios," *Expert Syst. Appl.*, vol. 296, Art. no. 129168, Jan. 2026, doi: 10.1016/j.eswa.2025.129168.
[31] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.
[32] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
[33] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD), 2016, pp. 785–794, doi: 10.1145/2939672.2939785.
[34] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," Nature, vol. 323, no. 6088, pp. 533–536, 1986, doi: 10.1038/323533a0.
[35] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," Neural Networks, vol. 18, no. 5–6, pp. 602–610, 2005, doi: 10.1016/j.neunet.2005.06.042.

Authors' Profiles



Kiranjeet Kaur is an Assistant Professor at Chandigarh University, Punjab, India, with over seven years of teaching and academic experience. She is currently a Ph.D. Scholar in the Department of Computer Science and Engineering at Chandigarh University, where her research focuses on cybersecurity, intrusion detection systems (IDS), machine learning, and reinforcement learning for intelligent cyber threat detection.

She has published 12 research papers in reputed national and international journals and conferences. Her research contributions also include three published patents and two book chapters in the fields of computer science and cybersecurity. She actively participates in research, innovation, and academic activities, with a keen interest in developing adaptive and resource-efficient intelligent security solutions for dynamic network environments.

Her current research aims to leverage advanced artificial intelligence techniques to enhance the accuracy, adaptability, and efficiency of next-generation intrusion detection systems.



Prof. (Dr.) Jaspreet Singh working as a Professor in the Department of Computer Science and Engineering, Chandigarh University, Mohali, Punjab, India. He holds B.Tech., M.Tech. and Ph.D. degrees in the field of Computer Science and Engineering. His research interests encompass Network Security, Cloud Computing, Artificial Intelligence and Reinforcement Learning. He has authored more than 100 research publications which are listed in Google Scholar, Scopus and SCI/SCIE-indexed journals, reflecting his significant contributions to the research field. Dr. Singh has served as a Session Chair at numerous IEEE and Springer international conferences and actively contributes to the academic community as a reviewer for leading journals and international conferences. He holds 10 patents under his name demonstrating his commitment to innovation and applied research. He is a Life Member of the Indian Society for Technical Education (ISTE) and the International Association of Engineers (IAENG). He has successfully supervised several Ph.D. and M.E. scholars in their research work. He is widely recognized for his contributions to research, innovation and academic excellence.

How to cite this paper

Kiranjeet Kaur, Jaspreet Singh, "Resource-Aware Proximal Policy Optimization for Adaptive Intrusion Detection in Dynamic Networks", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.16, No.5, pp. 158-173, 2026. DOI:10.5815/ijwmt.2026.05.10