International Journal of Artificial Intelligence, Data Science, and Machine Learning
Grace Horizon Publication | Volume 7, Issue 1, 71-76, 2026
ISSN: 3050-9262 | https://doi.org/10.63282/3050-9262.IJAIDSML-V7I1P114

Original Article

Reinforced Learning Based Firewall Architecture Leveraging Large Language Models for Adaptive, Unique Security Policy Synthesis

Sujay Kanungo
Independent Researcher Boston, USA.

Received On: 16/12/2025 ***Revised On: 18/01/2026*** ***Accepted On: 23/01/2026*** ***Published On: 30/01/2026***

Abstract - *In the rapidly evolving landscape of cybersecurity, the need for dynamic and adaptive security solutions has become paramount. This paper presents a novel firewall architecture that integrates reinforcement learning (RL) with large language models (LLMs) to enhance the synthesis of unique security policies tailored to specific network environments. By employing RL techniques, the architecture learns from real-time network traffic, adapting its defense mechanisms in response to emerging threats. Simultaneously, LLMs facilitate the interpretation and generation of security policies, allowing for a more intuitive interaction between security analysts and the system. The proposed architecture is evaluated through extensive simulations, demonstrating its effectiveness in reducing false positives and improving threat detection rates compared to traditional firewall systems. Our findings suggest that the synergy between RL and LLMs not only fosters more robust security postures but also streamlines the policy management process, offering a promising direction for future research in adaptive cybersecurity solutions.*

Keywords - *Reinforced Learning, Large Language Model, Firewall, Networking, Machine Learning*.

1. Introduction

The pervasive threat of cyber-attacks compromises crucial infrastructure, making the synthesis of adaptive and unique security policies an enduring challenge. Firewalls defend against external threats through controllable assets, yet existing works utilizing Reinforcement Learning (RL) to automate this process do not adequately assess or enhance safety and novelty. Policies remain vulnerable to covert adversarial manipulations that generate extensive, imperceptible changes to system configuration. A novel Reinforced Learning architecture enabling the synthesis of adaptive and unique firewall policies is proposed, leveraging Large Language Models (LLMs) to verify, complement, and generate security strategies [1].

Reinforced Learning enables security policy synthesis through interaction with environmental states to learn optimal configurations. An agent observes the environment, where an auto-generated description of current firewall rules articulates potential remediation points; it then selects specifications to fine-tune or modify the rule set. Temporal information frames past configurations as additional context—essential for institutions implementing coherent policies subject to rapid change—while other agents apply risk appraisal to assess safety before action. Each additional dimension within a multi-agent architecture provides distinct perspectives, enhancing adaptability through simultaneous exploration of varied strategies.

2. Background and Related Work

Reinforcement learning (RL) enables agents to learn optimal behaviors by interacting with their environment. Commercial software and hardware components producing abundant, heterogeneous data increase the need for adaptive security policies across various contexts. Consequently, network firewalls need to adapt their policies to diverse security objectives yet remain enforceable and machine-readable. Reinforcement-learning research has explored RL-driven multi-agent decision-making for network security [1]. It has also investigated automated policy generation and distribution without execution, based solely on policy files and narrative descriptions. Although large language models (LLMs) cannot yet accurately validate existing firewall policies [2] , they are valuable for reasoning about context and risk across diverse formats. Integrating RL-driven decisioning with LLMs for policy synthesis has substantial, largely unexplored potential. Large language models (LLMs) have revolutionized natural language processing through advanced tasks such as summarization, dialogue, translation, and reasoning. They are now being explored for their capacities in cybersecurity to process narratives about malicious behaviors, support professional training, and assist in code synthesis or completion. Extensive general pre-training enables LLMs to serve not only as generative models but also to verify or reason about generated code, a capability under initial investigation for generating machine-actionable cyber-intrusion-attack patterns.

3. Problem Formulation and Objectives

The synthesis of unique security policies for firewalls remains a vital yet challenging task, requiring periodic revision or entirely new formulations to counter evolving threats [3]. Reinforcement learning (RL) is a promising means of achieving autonomous, adaptive configuration.

However, most existing approaches generate generic rules across multiple environments, rendering successive policies insufficiently distinct [2]. Providing these models with explicit information regarding the policy's novelty could expedite convergence while also enabling orderly freshness and compliance verification.

A clear formal definition of the relevant environment is a prerequisite for RL policy generation. In this domain, action and state spaces vary widely among the available firewall decisioning functions, necessitating additional specification. Safety considerations add further constraints: regarding both the networked assets involved such as hosts, applications, and protocols and the need to avoid violation of pre-established governance and compliance constructs.

4. Architecture Overview

The proposed architecture comprises three layers. In the top layer, the environment captures all configurations and information required for firewall policy decisioning; this information is represented by a state vector. The middle layer consists of a reinforced learning agent. This agent first predicts which firewall rule to modify, using an auxiliary policy and guided exploration, to navigate the trade-off between exploration and exploitation. A policy-distillation module enables policy transfer to a second agent, which generates the actual modification required for policy adaptation. Finally, in the bottom layer, an external LLM receives the proposed adaptation and analyses its compliance to formulate independent validation signals. It also derives rationales from prior observations that justify the adaptation, which simplifies the interpretation for a network administrator [1].

Incorporating reinforced learning to determine firewall adaptations distinguishes the proposed system from existing approaches. Previous systems exclusively leverage LLMs for composing adaptations or revising policy documents freely. Although these techniques enhance policy presentation for improved admin cognition, the manual selection of modifications remains cumbersome, demanding significant input from the administrator. In the proposed system, reinforced learning interprets the current state of the environment, assesses whether to adapt the policy, and selects the modification to apply. This end-to-end process promotes a relaxed overview of consequence, reduces the labor of the administrator, and mitigates human errors arising from unforeseen circumstances.

4.1. System Layering and Components

Adaptive Firewall Policy Synthesis Via Reinforced Learning and Large Language Models. High-level architecture delineates layers, components, and data flows. The system comprises sensing, decisioning, enforcement, and feedback tiers. Beginning with network-state acquisition from the sensing layer, these observations traverse the layers toward policy synthesis at the decisioning level. Resulting configuration commands propagate down in the opposite direction to the enforcement layer, instantiating the policy on the firewall. Feedback may return along either direction for policy audits or formal property verification.

The notation of "firewall" and "policy" encompasses diverse solutions and rulesets. Adaptive decisions may concern selection of a pre-existing rule, modification of an existing rule, or addition of a new rule to the current configuration. Firewall-state observations may specify general characteristics (e.g., network type), describe rules or specific candidates for adaptation, and identify traffic flows or exploitation attempts. [4]

4.2. Reinforcement Learning Paradigm for Firewall Decisioning

Reinforcement learning offers a framework to derive adaptive firewall policies, using actions based on the current network state and feedback on the impacts of those actions. A Markov decision process defines this framework, where the system observes the network and takes actions based on an agent's policy. A learning signal, derived from the environment's reward, drives adaptation. Action sequence history forms the action space, while various temporal and network metrics contribute to the state space. In accordance with network safety principles, adaptation does not compromise policy monotonicity [5].

Existing approaches rely on a single policy, potentially rendering adaptation under adversarial conditions vulnerable to evasion and misdirection [2]. A dual-policy structure addresses these constraints. Validation establishes compliance with network requirements and active response inhibits erroneous adaptation. Policy updates follow a ring-and-feedback architecture across temporal neighborhoods to balance adaptation and stability. The integration exploits the distinct grammar between raw flow records and synthesis behavior, allowing complementary and cross-domain verification.

4.3. Large Language Model Integration for Policy Synthesis

Large language models (LLMs) exhibit human-like capabilities, prompting exploration of their use in material-policy synthesis. LLM-assisted frameworks have emerged for synthesizing packet-classification rules and natural-language security policies. Adapting similar techniques for reinforcement-learning (RL)-based firewall policy generation demands careful orchestration. LLMs require extensive contextual information to generate meaningful responses, complicating safe integration with established RL paradigms. Policy synthesis hinges on action, state, and observation definitions that diverge from traditional RL configurations, introducing further complexity. Moreover, harnessing LLMs for effective synthesis without supervised data remains challenging.

An integrated architecture is proposed to generate and reason about packet-filtering rules for adaptive firewall deployment. The approach combines a dual-policy controller with cross-modal validation, a self-referential-policy-verification module, and a context-aware-threat-reasoner. The

controller exploits LLMs to generate rules from high-level descriptions and assesses suitability based on packet-level scenarios. The verification module enables auto-checking of policy compliance against safety criteria and captures formal-verification signals, bolstering confidence and auditability. The threat-reasoner analyzes context information and infers critical attack vectors, outputting justifications that clarify choices for both generation and rejection. Finally, a zero-gradient privacy-preserving distillation mechanism permits LLM integration without exposing gradients, complemented by a temporal-policy web-of-trust for decentralized governance.

5. Unique Architectural Elements

A unique architectural element of the proposed design is the dual-policy controller enabling cross-modal validation. While existing approaches often generate only a single policy, this framework produces a high-level, context-aware guiding policy along with a detailed, low-level decision-making policy. The two modalities can be validated against each other to ensure coherent contextual understanding and consistent specification of security goals. Miscellaneous conflicts between the guiding and decisioning policies can compromise a desired level of adaptivity and safety. A variety of resolution techniques including priority settings, decision simulation, and uncertainty modeling have been devised to align the two policies more closely.

Another distinctive module is the self-referential policy-verification component, in which the agent exercises self-checking during policy synthesis. The controller loops back to the high-level adaptation policy to verify whether the actions taken still honor the high-level objectives delineated in the guidance prompt. Such checks provide a rough understanding of compliance, which can be complemented with additional formal-verification signals if more stringent guarantees are sought. To promote auditability, an extensive logging mechanism keeps track of the synthesized policy, the associated guidance, the self-check results, and the validation status obtained through external auditing methods.

The contextual threat-reasoning unit conducts inference on potential attack scenarios concerning the current state of the network, leveraging the policy model as a knowledge base. By integrating this scene-understanding capability, the system enables a justification chain to accompany proposed policy adaptations. Substantial justifications reinforce the credibility of change proposals, while links to actual industrial use cases enable practitioners to retrace the rationale and verify that the modifications align with organizational security objectives.

A zero-gradient privacy-preserving distillation strategy accommodates policy extraction from the agent without collecting explicit data samples that would breach user confidentiality. Contemporary data-free techniques allow the distilled models to be shaped solely on the basis of broader behavior-related information such as output statistics, selection patterns, and inductive biases rather than on the explicit data. The underlying strategy remains gradient-free, ensuring that the policy remains undisclosed throughout the whole distillation process and a bound on the privacy budget.

Finally, a temporal-policy web-of-trust mechanism facilitates the establishment of trust among distinct organizational policies when a time dimension is involved. Trust is in general preserved throughout the propagation process in different time perspectives, which localizes any trust decay to the change duration and allows a multi-stage solution. Consequently, the additional trust metric does not significantly increase the analysis complexity while enriching the graph with wider interpretation capability and greater flexibility. [4]

5.1. Dual-Policy Controller with Cross-Modal Validation

The cornerstone of the proposed architecture is a dual-policy controller, which sustains the interaction with a large language model (LLM) through network-aware and LLM-adapted policies. When a network or service change occurs, both policies are re-evaluated, leveraging cross-modal channels to validate and, if necessary, rectify conflicting decisions. An LLM fuses the two policies, temporarily disallowing a direct connection and enabling internal cross-checks. This configuration guarantees complementary intelligence across heterogeneous system layers, ensuring the distilled policy matches security criteria articulated in natural language thereby upholding the architecture's overarching aim of adaptive yet unique policy learning [1].

The primary novelty lies in validation across disparate modalities, re-assessing the rationale behind policy adjustments issued by the high-level LLM when discrepancies with the low-level automation policy arise. The LLM first interrogates the low-level policy to elicit justifications and subsequently scrutinizes these against the trajectory generated by its high-level policy. If the high-level modification fails to satisfy foundational precepts outlined in prior reasoning, or if the low-level refutation remains unaddressed, the LLM delineates limitations that warrant further attention.

5.2. Self-Referential Policy Verification Module

Reinforced Learning Based Firewall Architecture Leveraging Large Language Models for Adaptive, Unique Security Policy Synthesis Aiming to further enhance the uniqueness and safety of the synthesized firewall policies, a self-referential policy verification module is incorporated into the architecture to perform exhaustive self-checks on the synthesized security policies before deployment. The verification module generates a set of formal verification signals and precisely quantifies their validity based on the generated security policy, providing valuable information about the safety of the policy. Moreover, two independent and traceable auditing trails are recorded, allowing users to review the formal verification details and enhancing the interpretability of the system. By utilizing these self-checks, the number of synthesized security policies requiring additional inspections is significantly reduced, consequently augmenting the safety of the automated synthesis system [1].

5.3. Context-Aware Threat Reasoner

In rapidly evolving network environments, new threats or modifications of existing ones can emerge at any time. Such contextually driven threats can use legitimate channels to compromise critical assets, resulting in consequences such as data leakage, theft of intellectual property, or operational disruption. Real-time identification of intrusion signals that signify these new threats happening on the fly and the policy modifications required for timely defense have hence become an important direction of adaptive firewall reinforcement learning (RL) research [5].

An innovative approach to tackle this is through the reasoning of the contextual threats of ensemble triggers taking place at different points in time. For any policy modification done on an existing policy under the influence of an ensemble of triggers, whether the new policy can still defend against the previous ones while allowing some form of defense against the latest addition, is a fundamental question of security preservation. To accurately assess this is non-trivial, as the same set of triggers posed at different times may lead to different policy constraints. The key lies in understanding the scene in which those signals appear and what threats consist of that scene. By extracting current scene understanding representations together with the latest temporal trigger information, temporal contextual understanding and trigger deriving are therefore conducted, followed by thorough exploration of the triggering chains in question. A justification chain in natural language [6] will then be generated automatically to explain what change in scene pose what constraint to the firewall decision, where policy adaptation specifies how to modify the current policy accordingly.

5.3. Zero-Gradient Privacy-Preserving Distillation

Most modern policies governed by Large Language Models (LLMs) are sensitive in public settings. To prevent sensitive information from leaking and to satisfy institutional privacy compliance, privacy-preserving methods become necessary. Distillation methods transfer knowledge from a teacher model to a student model. When utilizing an LLM as a teacher, no gradient information in the student model can be exploited to obtain intelligence about private training data, which establishes a zero-gradient solution under differential privacy (DP) constraints [7]. To guarantee compliance with privacy budgets, leveraging other prior knowledge still requires careful design of the learning framework.

Fine-tuning is a process whereby an already trained model is further trained on additional similar data for specific tasks. The degrees of freedom of the student model can be severely restricted to obtain knowledge without gradient information. With well-trained prior knowledge distilled to initialize the student model, adaptive additional training upon the new domain permits either a drastic reduction of training samples or preserving generalization performance with fewer gradient queries from the public LLM.

5.4. Temporal-Policy Web of Trust

Temporal-Policy Web of Trust. The time-aware temporal-policy web of trust enables the propagation of trust across time-varying firewall policies. A policy at a specific time point can propagate trust backward or forward in time to other policies on the temporal-policy web, depending on policy provenance, external endorsements, and time-varying threat information. The accumulated trust across the temporal-policy web serves as time-aware evidence for the real-time verification of the temporal-policy web. External policies can propagate trust not only to the primary policy but also to the intermediaries on the backward temporal-policy web that have received a trust endorsement, leading to a cascading effect that further illustrates the propagation mechanism. To decide whether the zero-gradient privacy-preserving policy distillation is trustworthy, the accumulated temporal trust is exploited as a time-dependent endorsement. By tracing back to check the governance of the suspended distilled policy, the zero-gradient privacy-preserving policy distillation has conditional access to guidance on what auxiliary information is safe to sample, thus maintaining a functional distilled policy while inspecting privacy.

6. Learning framework and Safety mechanisms

Firewall tuning has been addressed within the framework of reinforcement learning (RL), where prior works emphasized performance and adaptability; however, this design prioritizes uniqueness, an essential yet underexplored aspect for real-world deployments facing evolving attack techniques. By integrating RL with large language models (LLMs), this approach enhances policy interpretation, synthesis, and cross-modal validation while governing both the learning process and policy credentials to ensure safe RL in networking contexts. The architecture commits to a limited set of options per time step, improving learnability without sacrificing adaptability or policy breadth. A balanced reward scheme supports exploration while penalizing unwarranted adaptations or performance degradation, and safety is enforced through a safety penalty. The exploration-exploitation dilemma in high-dimensional action spaces is addressed with an exploration budget and annealing strategy as training episodes increase. While RL has been successfully applied to automated network security, safety remains a crucial consideration to prevent jeopardizing system security, necessitating that RL policies remain within defined safety constraints. Three primary safety techniques for network environments include constraint methods that employ expert-defined safe states, barrier methods that use approximate models to ascertain safety, and occupancy measures that characterize vehicle information over time. Moreover, current neural firewalls are vulnerable to evasion attacks, where adversarial perturbations seek to bypass detection systems. Mitigation approaches currently require prior knowledge of attack types, while RL provides strategies with minimal packet alteration, alleviating the need for manual defense configuration. LLMs enhance policy generation by facilitating an in-depth understanding of underlying threats, yet the lack of formalism necessitates a grounded exploratory capability for threat extraction and policy identification. This

integrated approach aims to create a robust, protection-oriented design that safeguards against recognized threats.

7. Discussion of security, privacy and compliance

Reinforcement learning (RL) firewalls are vulnerable to manipulation by adversarial agents equipped with advanced reconnaissance capabilities, falling prey to attack categories such as extraction, evasion, transformation, and poisoning. These visibility-causing attacks exploit vulnerabilities to reconstruct easily extractable policies through low-cost sampling methods, degrading performance via data poisoning. Adversarial agents can inexpensively modify training rounds, thereby altering, gathering, or disseminating low-cost observations without loss. While the implementation of large language models (LLMs) bolsters the firewall's safety by preventing policy reconstruction and misuse, privacy concerns arise due to the localization of shared content, which reduces policy recordings and limits data access on external servers. Mitigation strategies include anonymization techniques like K-anonymity, which obscure sensitive data attributes, and privacy-preserving mechanisms that minimize the volume of sensitive information collected during firewall operations. Compliance with data privacy principles is assured through formal privacy audits, verifying that datasets do not leak sensitive information. Despite the advancements in adaptive firewall policy synthesis using RL, significant security threats persist, providing adversaries with multiple attack vectors that challenge effective deployment. The synthesis of unique security policies remains an area of active research, as there is currently no established process for creating adaptive, unique, and secure firewall decisions simultaneously.

8. Conclusion

Through this work, articulated a framework for deploying reinforced learning on firewalls and proposed a unique architecture aimed at enabling adaptive yet unique policy synthesis. Structured a reinforced-learning formulation adapted to firewall policy design, spanning agent actions, states, environment observables, and reward signals. Identified two constraints on reinforcement-learning approaches: policy novelty and safety of actions. Despite extensive research on reinforcement-learning for network security, no prior work was found addressing these criteria concurrently. Existing approaches based on large language models for reasoning and synthesis were also reviewed and found unable to address adaptivity, uniqueness, safety, or performance.

The architecture integrates reinforced learning with large language models to tackle these challenges. A preliminary large language model opinion is solicited on the novelty and safety of proposed actions. Two prominent policies are therefore learned: one that maximizes safety and another that maximizes novelty. By contrasting the policies before decisioning, the system captures further stylistic variance and steers towards nodes that the large language model does not consider sufficiently novel or safe. Moreover, actions issued by the reinforced-learning agent are subject to large-language-model validation and either rejected or refined according to a co-participating synthesis prompt. Four additional modules enhance interpretability, promote compliance with security standards, and facilitate auditing while safeguarding proprietary information. The architecture hence synthesizes firewall policies that are at once unique, safe, performant, and interpretable.

References

[1] K. Hammar and R. Stadler, "Finding Effective Security Strategies through Reinforcement Learning and Self-Play," 2020.

[2] M. Wolk, A. Applebaum, C. Dennler, P. Dwyer et al., "Beyond CAGE: Investigating Generalization of Learned Autonomous Network Defense Policies," 2022.

[3] K. Hammar and R. Stadler, "Intrusion Prevention through Optimal Stopping," 2021.

[4] P. G. Clark, "Firewall Policy Diagram: Novel Data Structures and Algorithms for Modeling, Analysis, and Comprehension of Network Firewalls," 2013.

[5] J. Mern, K. Hatch, R. Silva, C. Hickert et al., "Autonomous Attack Mitigation for Industrial Control Systems," 2021.

[6] J. Jin, B. Tang, M. Ma, X. Liu et al., "Crimson: Empowering Strategic Reasoning in Cybersecurity through Large Language Models," 2024.

[7] T. Chen, L. Da, H. Zhou, P. Li et al., "Privacy-preserving Fine-tuning of Large Language Models through Flatness," 2024.

[8] X. Zhou, S. Yusuf Enoch, and D. Seong Kim, "Markov Decision Process For Automatic Cyber Defense," 2022.

[9] Y. He, J. Qiu, W. Zhang, and Z. Yuan, "Fortifying Ethical Boundaries in AI: Advanced Strategies for Enhancing Security in Large Language Models," 2024.

[10] I. Tsingenopoulos, V. Rimmer, D. Preuveneers, F. Pierazzi et al., "Adversarial Markov Games: On Adaptive Decision-Based Attacks and Defenses," 2023.

[11] N. Thomas McDermott, J. Yang, and C. Mao, "Robustifying Language Models with Test-Time Adaptation," 2023.

[12] E. Shayegani, M. Abdullah Al Mamun, Y. Fu, P. Zaree et al., "Survey of Vulnerabilities in Large Language Models Revealed by Adversarial Attacks," 2023.

[13] J. Nyberg and P. Johnson, "Training Automated Defense Strategies Using Graph-based Cyber Attack Simulations," 2023.

[14] N. Tihanyi, M. Amine Ferrag, R. Jain, and M. Debbah, "CyberMetric: A Benchmark Dataset for Evaluating Large Language Models Knowledge in Cybersecurity," 2024.

[15] A. Kumar, S. Singh, S. Vignesh Murty, and S. Ragupathy, "The Ethics of Interaction: Mitigating Security Threats in LLMs," 2024.

[16] A. Esmradi, D. Wankit Yip, and C. Fai Chan, "A Comprehensive Survey of Attack Techniques, Implementation, and Mitigation Strategies in Large Language Models," 2023.

[17] H. Li, Y. Chen, J. Luo, Y. Kang et al., "Privacy in Large Language Models: Attacks, Defenses and Future Directions," 2023.

[18] J. Clements, Y. Yang, A. Sharma, H. Hu et al., "Rallying Adversarial Techniques against Deep Learning for Network Security," 2019.

[19] A. M. Kassem, "Mitigating Approximate Memorization in Language Models via Dissimilarity Learned Policy," 2023.

[20] N. Schnepf, R. Badonnel, A. Lahmadi, and S. Merz, "Generation of SDN policies for protecting Android environments based on automata learning," 2018.