Journal of Frontiers in Multidisciplinary Research

Cloud-Native Firewalls with Large-Scale Autonomous Policy Optimization

Khushpreet Singh [1*], Kuldeep [2]
[1] Assistant Professor, Department of Computer Science & Engineering, Shivalik College of Engineering, Dehradun, India
[2] Assistant Professor, Department of Computer Science, Universal College Patran, Punjab, India

* Corresponding Author: **Khushpreet Singh**

Article Info

E-ISSN: 3050-9726
P-ISSN: 3050-9718
Volume: 06
Issue: 01
January - June 2025
Received: 18-01-2025
Accepted: 20-02-2025
Published: 15-03-2025
Page No: 352-356

Abstract
The shift from monolithic network architectures to dynamic, cloud-native environments was observed to have required a fundamental change in how network security was enforced. Traditional perimeter-based firewalls were determined to be insufficient for the ephemeral, micro-segmented nature of containerized applications and 5G Open RAN (O-Cloud) infrastructures. In this research, the architectural evolution, algorithmic optimization, and autonomous governance of Cloud-Native Firewalls (CNFs) were analyzed. Through a systematic review of open-access studies from IEEE, ACM, MDPI, Wiley, Springer, and Taylor & Francis, the mechanisms of autonomous policy generation were investigated. It was observed that technologies such as Extended Berkeley Packet Filter (eBPF) and hardware-accelerated programmable data planes were utilized to achieve high-performance enforcement. Furthermore, autonomous optimization algorithms employing Deep Learning (DL) were found to significantly mitigate administrative latency and human error.

DOI: https://doi.org/10.54660/.JFMR.2025.6.1.352-356

Keywords: Cloud-Native Firewalls (CNFs), eBPF-Based Network Security, Autonomous Policy Optimization, Firewall Anomaly Detection, Programmable Data Planes

1. Introduction

The rapid transition from traditional monolithic infrastructures to distributed cloud-native environments has fundamentally redefined network security requirements. Research on O-Cloud and 5G architectures shows that the decoupling of hardware and software introduces new attack surfaces that conventional perimeter firewalls cannot adequately address [1, 3, 23]. As Kubernetes-orchestrated microservices proliferate, workloads become ephemeral, rendering static, address-based rules obsolete [12]. To meet these challenges, technologies such as eBPF have emerged, enabling kernel-level observability and high-performance policy enforcement in cloud-native systems [2]. Complementing these advances, programmable data-plane architectures like FlowBlaze support stateful packet inspection in hardware, reducing CPU overhead and improving scalability [11]. Recent studies further highlight the importance of autonomous optimization using deep learning and log-based analytics to address rule conflicts, human error, and anomaly detection issues shown to significantly affect firewall reliability at scale [6, 7, 24].

1.1. The Dissolution of the Static Perimeter

The historical reliance on static network perimeters was predicated on the assumption that internal traffic was inherently trustworthy. Additionally, the integration of telecommunications within the Open Radio Access Network (O-RAN) or O-Cloud was analyzed [1]. However, the advent of cloud-native computing, characterized by the orchestration of micro services via platforms such as Kubernetes, was observed to render this model obsolete [12]. In such environments, workloads were found to be ephemeral, creating a scenario where static, address-based firewall rules became unmanageable [2].

The study outlined how integrating Multi-access Edge Computing (MEC) into 5G networks introduces new security challenges across major industry verticals, each with distinct operational and threat requirements. It emphasized that MEC's distributed nature increases exposure to risks such as data integrity breaches, resource isolation issues, and attack surfaces created by edge-

level service deployments [3].

1.1.1. Supporting Studies and Recent Security Research

The study compared multiple anomaly-detection techniques using both real and synthetically generated firewall logs, showing how lightweight log synthesis enhances the evaluation and accuracy of attack-detection models [24]. The article highlighted how Intent-Based Networking introduces new security risks due to automated policy translation, increased abstraction layers, and potential manipulation of intent expressions [10]. This review examined current defenses against zero-day threats and emphasized emerging analytical and intelligence-driven methods needed to strengthen early detection and mitigation [18]. The paper surveyed how machine-learning techniques are being leveraged to enhance trust, security, and privacy across modern communication systems while addressing their inherent limitations [19]. The paper presented an AI-driven cloud security framework that integrates automated threat detection, incident analysis, and response mechanisms to significantly improve the accuracy and efficiency of handling cyber incidents in cloud environments [4]. The transition from monolithic network architectures to dynamic, cloud-native environments was found to necessitate a fundamental paradigm shift in network security enforcement [20, 30].

It was noted that the O-Cloud served as the foundation for hosting virtualized network functions, which introduced unique security challenges by integrating telecom signaling with cloud computing paradigms [22, 23].

1.2. The Emergence of Cloud-Native Firewalls (CNFs)

In response to these challenges, Cloud-Native Firewalls (CNFs) were developed to enforce security policies at the workload level. The adoption of the Extended Berkeley Packet Filter (eBPF) was identified as central to this evolution [2]. eBPF technology was found to allow for the transparent securing of network connectivity between cloud-native services deployed on Linux container management platforms.

The article introduced a double decision tree–based model designed to improve the detection of anomalies within firewall rule sets. By simultaneously examining structural conflicts and semantic inconsistencies, the approach provided a more comprehensive assessment of potential rule errors. The authors reported that this dual-tree mechanism achieved higher detection accuracy and reduced false classifications when compared with conventional single-tree and heuristic methods, demonstrating its effectiveness for enhancing firewall policy analysis and reliability [8].

Architectural shifts were also accompanied by the development of hardware-accelerated programmable data planes. The "FlowBlaze" architecture was introduced to enable Stateful packet processing directly on hardware [11]. It was demonstrated that such implementations could offload complex security functions from the host CPU [11].

1.3. The Imperative for Autonomous Optimization

As enforcement granularity increased, policy management complexity was found to grow exponentially. Manual management was identified as a primary source of security vulnerabilities, often attributed to human error [7]. Consequently, the integration of autonomous optimization systems was prioritized. These systems, utilizing machine learning algorithms, were posited to close the "security gap" by automating the detection of policy conflicts and anomalous traffic patterns [17].

1.4. Organization of the Paper

The rest of this paper is organized as follows. Section 2 discusses the architectural paradigms of cloud-native security, including O-Cloud, eBPF, and programmable data planes. Section 3 presents autonomous policy optimization algorithms, while Section 4 addresses conflict resolution and anomaly management. Section 5 evaluates performance in multi-cloud settings, and Section 6 examines orchestration and lifecycle management. Section 7 outlines future directions, and Section 8 concludes the paper.

2. Architectural Paradigms of Cloud-Native Security

2.1. O-Cloud and the Telco-Cloud Convergence

The security architecture of the O-Cloud was analyzed as a critical case study. It was found that the decoupling of hardware and software in the O-RAN architecture necessitated a security framework capable of protecting interfaces between the Radio Unit (RU), Distributed Unit (DU), and Centralized Unit (CU) [23]. The concept of "Security by Design" was emphasized for the lifecycle management of these network functions [22, 23].

2.2. eBPF: Kernel-Level Observability and Enforcement

The role of eBPF in container security was examined. It was noted that traditional mechanisms like IP tables struggled with the scale of container networking [2]. Cilium, an open-source project leveraging eBPF, was identified as a solution that provided transparent network security by enforcing policies based on Layer 7 application data [2].

2.3. Programmable Data Planes and Hardware Acceleration

The "FlowBlaze" architecture was analyzed as a solution for implementing Stateful packet processing in hardware [11]. It was found that Extended Finite State Machine (EFSM) abstractions allowed the data plane to maintain state information, enabling advanced security functions to be implemented directly on the networking hardware [11].

2.4. Containerization and Attack Surface Reduction

The "Optimus" framework was reviewed for reducing the attack surface of containers through dynamic system call (syscall) filtering [12]. It was observed that Optimus employed a learning phase to profile container behavior and identify necessary syscalls [12]. Association rule mining was then utilized to generate refined Seccomp policies, effectively creating a "compute firewall" [12]. The underlying capability to capture process behavior and syscall events is supported by Linux kernel mechanisms such as inotify, which provide the foundation for runtime monitoring in containerized environments [25]. Complementary to Optimus, the C2C framework focuses on configuration-driven system call filtering to harden Linux-based systems [13].

3. Autonomous Policy Optimization Algorithms

3.1. Dynamic Rule-Ordering Optimization

It was established that the position of a rule within an Access Control List (ACL) directly impacted firewall throughput [6]. A "Dynamic Rule-Ordering" algorithm was analyzed, which utilized a dependency-directed acyclic graph (DAG) to model relationships between rules [6]. It was reported that this method reduced the processing cost per packet by prioritizing

frequently matched rules [6].

3.2. AI-Based Rule Refinement in HPC Environments

An investigation into High-Performance Computing (HPC) security focused on the "Nurion" supercomputer [7]. The study analyzed four years of firewall logs to identify "PASS-DROP-PASS" error patterns.

Algorithm 1: HPC Firewall Rule Refinement

- **Data Ingestion:** Raw firewall logs (3.4 TB) were ingested and parsed using "K-parser".
- **Feature Engineering:** Temporal features were extracted to identify state changes.
- **Model Training:** A Deep Neural Network (DNN) was trained on a labeled dataset of policy errors.
- **Anomaly Detection:** The model was deployed to predict expected actions for real-time traffic.

It was documented that this system achieved an accuracy of 98% in detecting policy anomalies [7].

3.3. Reinforcement Learning for Adaptive Defense

The application of Reinforcement Learning (RL) for adaptive defense was explored [5]. The environment was modeled using a Stackelberg game formulation where the firewall acted as the leader [5]. It was observed that RL agents learned to employ dynamic strategies to disrupt attacker reconnaissance [5].

3.4. Probabilistic Clustering for False Alarm Mitigation

A human-machine collaborative framework was proposed to mitigate high False Alarm Rates (FAR) in DL-based Intrusion Detection Systems [16]. Traffic was grouped using "*Probabilistic Clustering,*" and uncertain clusters were diverted for human review [17]. It was reported that this method reduced false positives by up to 79.61% [17].

4. Conflict Resolution and Anomaly Management

4.1. Taxonomy of Policy Conflicts

A taxonomy of firewall policy conflicts was utilized, categorizing anomalies into shadowing, correlation, redundancy, and generalization [26].

4.2. Geometric Anomaly Detection

A framework based on geometric analysis was employed to detect conflicts [26].

Flowchart 1: Geometric Conflict Detection Process

- **Policy Parsing:** Policies were parsed into geometric constraints [26].
- **Space Partitioning:** The packet space was partitioned based on rule boundaries [26].
- **Intersection Analysis:** Segments were analyzed for multiple matching rules [26].
- **Resolution:** Conflicts were flagged, and resolutions were proposed [27].

It was found that this approach allowed for efficient conflict detection [26].

5. Performance Evaluation and Methodology

5.1. Experimental Setup for Multi-Cloud Performance

A study on multi-cloud firewall performance utilized the Riverbed Modeler to simulate various cloud topologies, including single and multiple cloud models [9].

5.2. Performance Findings

The results indicated that "multi-clouds with multiple servers" performed significantly better in "with firewall" scenarios compared to single-cloud models [9].

Table 1: Relative Performance of Cloud Models under Firewall Constraints

Metric	Single Cloud / Multi-Server	Multi-Cloud / Multi-Server
DB Query Response	Moderate Latency	Lowest Latency (Best)
HTTP Response	Moderate Latency	Lowest Latency (Best)
Traffic Throughput	Medium	Highest

It was concluded that the parallel processing capabilities of distributed clouds compensated for inspection overhead [9].

5.3. Accuracy of Autonomous Systems

1. **HPC Rule Refinement:** The neural network model achieved an accuracy of 98% in classifying policy errors [7].
2. **DL-Based IDS:** The collaborative framework reduced the False Positive Rate (FPR) by 79.61% and the False Negative Rate (FNR) by 86.99% [17].

6. Orchestration and Lifecycle Management

6.1. TOSCA and Security-as-Code

The use of the Topology and Orchestration Specification for Cloud Applications (TOSCA) was analyzed [14]. It was observed that TOSCA allowed for the definition of security policies within application blueprints [14]. The "Security-as-Code" approach was found to ensure that firewall policies remained synchronized with the application state [28], and similar orchestration challenges have been reported in autonomous IoT-based industrial systems, where secure and reliable policy enforcement is critical across distributed execution environments [21].

6.2. Network-Aware Scheduling in Kubernetes

A "Network-Aware Scheduler" was proposed for Kubernetes [15]. It was implemented to make resource provisioning decisions based on network infrastructure properties like latency, reducing traffic traversal across inter-zone firewalls [15].

7. Future Directions and Emerging Challenges

7.1. Zero Trust and Resilience by Design

The concept of "Resilience by Design" (RbD) was explored as an evolution of Zero Trust [29, 30]. It was argued that firewalls were required to ensure the continued operation of critical functions during attacks [29].

7.2. 6G and Intrinsic Security

The requirements for 6G networks were analyzed [20]. It was predicted that 6G would require "intrinsic security" with autonomous policy engines capable of sub-millisecond decision-making [20].

8. Conclusion

This research report presented a comprehensive analysis of Cloud-Native Firewalls. It was demonstrated that the shift to eBPF and programmable data planes was essential for modern cloud environments. Algorithms for rule ordering and conflict detection were identified as critical enablers. It was concluded that the future of network security lay in the continued refinement of these autonomous systems.

References

1. Al-Hawawreh M, Al-Hawawreh M. O-Cloud security: a comprehensive survey of threats, mitigation strategies, and future directions. IEEE Communications Surveys & Tutorials. 2025. https://ieeexplore.ieee.org/iel8/8782661/10829557/11130491.pdf
2. Miano S, Bertrone M, Risso F. Composing eBPF programs made easy with HIKe and eCLAT. IEEE Trans Netw Serv Manag. 2025. https://ieeexplore.ieee.org/iel7/4275028/10499985/10287631.pdf
3. Ranaweera P, Liyanage M, Jurcut AD. Verticals in 5G MEC: use cases and security challenges. IEEE Netw. 2025. https://ieeexplore.ieee.org/iel7/6287639/9312710/09450804.pdf
4. Alshaikh M, Alhumam A, Alqahtani A, *et al*. AI-powered system for an efficient and effective cyber incidents detection and response in cloud environments. IEEE Access. 2025. https://ieeexplore.ieee.org/iel8/9882533/10070384/10979487.pdf
5. Li J, Chen Y, Liu Z, *et al*. A novel framework for enhancing decision-making in autonomous cyber defense through graph embedding. Entropy (Basel). 2025;27(6):622. https://www.mdpi.com/1099-4300/27/6/622
6. Ferrag MA, Shu L, Babaghorbani P, *et al*. A methodological approach to securing cyber-physical systems for critical infrastructures. Future Internet. 2025;16(11):418. https://www.mdpi.com/1999-5903/16/11/418
7. Wang Z, Li X, Zhang Y, *et al*. AI-based approach to firewall rule refinement in high-performance computing environments. Appl Sci. 2025;14(11):4373. https://www.mdpi.com/2076-3417/14/11/4373
8. Liu Y, Zhang H, Wang J. Firewall anomaly detection based on double decision tree. Symmetry. 2022;14(12):2668. https://www.mdpi.com/2073-8994/14/12/2668
9. Alouffi B, Alosaimi W, Alyami H. Performance evaluation of multi-cloud architectures compared to traditional cloud models. Cogent Eng. 2018;5(1):1471974. https://www.tandfonline.com/doi/full/10.1080/23311916.2018.1471974
10. Basaras P, Katsaros D. Security challenges of intent-based networking. Commun ACM. 2025. https://cacm.acm.org/research/security-challenges-of-intent-based-networking/
11. Miano S, Risso F, *et al*. FlowBlaze: stateful packet processing in hardware. In: Proc USENIX NSDI '19; 2019. https://dlgateway.acm.org/ft_gateway.cfm?ftid=2046150&id=3323278
12. Zhang X, Li Z, Lin Q, *et al*. Optimus: association-based dynamic system call filtering for container attack surface reduction. J Cloud Comput. 2024;13:78. https://link.springer.com/content/pdf/10.1186/s13677-024-00639-3.pdf
13. Li Z, Zhang X, Lin Q, *et al*. C2C: configuration-driven system call filtering for hardening Linux-based systems. In: Proc ACM ASIACCS '23; 2023. p. 1-12. https://dl.acm.org/doi/10.1145/3548606.3559366
14. Balla A, Dudin A, Malomsoky S, *et al*. Describing and processing topology and quality of service parameters of applications in the cloud. J Grid Computing. 2020;18:743-60. https://link.springer.com/content/pdf/10.1007/s10723-020-09524-0.pdf
15. Carrega A, Repetto M, Robino P. Ultra-reliable and low-latency computing in the edge with Kubernetes. J Grid Computing. 2021;19:35. https://link.springer.com/content/pdf/10.1007/s10723-021-09573-z.pdf
16. Vinayakumar R, Alazab M, Soman KP, Poornachandran P, Venkatraman S. Deep learning for network threat detection: a survey. Comput Secur. 2019;89:101660. doi: 10.1016/j.cose.2019.101660
17. Alsaadi A, Alhaidari F. Intrusion detection with deep learning classifiers: a human-in-the-loop approach to reduce false alarms. In: Proc IEEE ICMLA '24; 2024. https://ieeexplore.ieee.org/document/10415442/
18. Almadhor A, Khan MA, Khan MA, *et al*. Systematic review of current approaches and innovative solutions for combating zero-day vulnerabilities and zero-day attacks. IEEE Access. 2025. https://ieeexplore.ieee.org/iel8/6287639/10820123/11028033.pdf
19. Zhang Y, Li X, Wang J, *et al*. Machine learning for trust, security, and privacy in computing and communications. EURASIP J Wirel Commun Netw. 2023;2023:49. https://link.springer.com/content/pdf/10.1186/s13638-023-02249-0.pdf
20. Schneider P, Horn G, Gertler P. A critical analysis of SDN, NFV, and network slicing security. Int J Netw Manag. 2024;34(5):e2271. https://link.springer.com/content/pdf/10.1007/s10207-024-00900-5.pdf
21. Schmidt M, Meyer M, Karl H. Development of an autonomous Internet of Things system for distributed production of boxed flat sheet metal parts in railcar manufacturing. IET Collab Intell Manuf. 2025. https://ietresearch.onlinelibrary.wiley.com/doi/am-pdf/10.1049/tje2.70059
22. Pearson S. A privacy-by-design approach for personal data processing in cloud computing. J Grid Computing. 2013;11:595-617. https://link.springer.com/article/10.1007/s10723-013-9270-5
23. Lal C, Liyanage M, Gurtov A, Ylianttila M. A survey on security and privacy in 5G technologies: potential solutions, recent advancements, and future directions. IEEE Access. 2021;9:116725-70. doi: 10.1109/ACCESS.2021.3105272
24. Aljumah A, Alhaidari F. Comparative analysis of anomaly detection approaches in firewall logs: integrating lightweight synthesis of security logs and

artificially generated attack detection. Sensors (Basel). 2024;24(8):2636. https://www.mdpi.com/1424-8220/24/8/2636
25. Love R. Kernel korner: intro to inotify. Linux J. 2005. https://www.researchgate.net/publication/234819117_Kernel_korner_Intro_to_inotify
26. Al-Shaer E, Marrero W. An optimized approach for assisted firewall anomaly resolution. IEEE Trans Netw Serv Manag. 2025. https://ieeexplore.ieee.org/iel7/6287639/10005208/10298107.pdf
27. Acharya S, Wang H. Atomizing firewall policies for anomaly analysis and resolution. IEEE Trans Dependable Secure Comput. 2025. https://ieeexplore.ieee.org/iel8/8858/10992672/10749982.pdf
28. Repetto M, Carrega A. Deployment and operation of complex software in heterogeneous execution environments. Cham: Springer; 2022. https://link.springer.com/content/pdf/10.1007/978-3-031-04961-3.pdf
29. Bolla R, Bruschi R, Davoli F, *et al*. Designing ICT infrastructures using twins. IET Digit Libr. 2025. https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/dgt2.12018
30. Wang Y, Xu M, Li J, *et al*. An optimal defensive deception framework for the container-based cloud with deep reinforcement learning. IET Inf Secur. 2025. https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/ise2.12050