From Operational Efficiency to Financial Vulnerability: A Theoretical Framework for Mitigating Economic Denial of Sustainability

Azar Mamiyev

Doctoral School of Applied Informatics and Applied Mathematics, Óbuda University, Budapest, Hungary
azarmamiyev@stud.uni-obuda.hu

Prof. Dr. Kornélia Lazányi

Wekerle International University
lazanyi.kornelia@wsne.hu

Abstract: The event-driven billing model of Serverless Computing introduces a vulnerability known as Economic Denial of Sustainability (EDDoS), or Denial of Wallet (DoW), in which adversaries exploit auto-scaling to inflict financial damage without triggering conventional availability alerts. Recent supervised approaches have achieved detection accuracies up to 99.90%, yet depend on labelled attack data and typically operate on network-layer or system-layer metrics rather than billing-layer signals. This paper identifies the predominance of supervised, offline, non-billing-centric approaches as a gap and proposes a conceptual framework exploring unsupervised anomaly detection, specifically Isolation Forest applied to billing telemetry, as a complementary approach. The framework operates asynchronously and employs graduated response orchestration. Preliminary validation on a public DoW benchmark dataset demonstrates 91.47% precision and an AUC-ROC of 0.671 without exposure to attack labels during training. The trade-offs between supervised accuracy and unsupervised practicality are discussed.

Keywords: Economic Denial of Sustainability, Denial of Wallet, Serverless Computing, Function-as-a-Service, Cloud Security, Anomaly Detection, Machine Learning

1 Introduction

The paradigm of serverless computing has fundamentally restructured the economics of cloud infrastructure. Function-as-a-Service (FaaS) platforms such as AWS Lambda, Google Cloud Functions, and Microsoft Azure Functions enable

developers to deploy event-driven applications with a granular pay-per-use billing model, charging exclusively for compute resources consumed during function execution [1], [2], [3]. For enterprise decision-makers, this shift transforms infrastructure cost from a fixed budget line into a variable expense directly coupled to application demand. However, the very properties that deliver this efficiency (automatic horizontal scaling, event-driven invocation, and usage-based billing) simultaneously introduce financial vulnerabilities with no precedent in traditional cloud deployments. An adversary can trigger arbitrary financial expenditure on the victim's account simply by generating sufficient invocations [4]. This attack vector, termed Economic Denial of Sustainability (EDDoS) or Denial of Wallet (DoW), targets the victim's financial sustainability rather than service availability [5], [6].

The distinction has profound implications for defence. Traditional DDoS attacks produce observable symptoms such as service degradation, latency, and resource saturation, which trigger well-understood alerting mechanisms [7]. EDDoS attacks, by contrast, exploit the fact that the serverless platform continues to function correctly: the application remains responsive, SLAs are met, and no alarm is raised. The damage manifests solely in the billing statement, potentially days or weeks after the attack [4], [8].

Recent supervised approaches have achieved impressive detection accuracies: 97.98% for DoWNet [9] and 99.90% for Neural ODE models [10]. However, these methods share common characteristics: they require labelled attack data, operate on network-layer or system-layer metrics rather than billing-layer signals, and are generally designed for offline analysis [11], [12]. The "leech" attack variant, which operates at traffic volumes indistinguishable from legitimate usage [4], poses particular challenges for systems depending on pre-labelled patterns.

This paper contributes a critical analysis of the gap between current detection paradigms and the financial threat model of serverless EDDoS. The primary contribution is a conceptual framework arguing for a shift from availability-centric, network-layer monitoring toward cost-centric, billing-layer anomaly detection using lightweight unsupervised machine learning, in particular Isolation Forest [14]. The intent is not to displace high-accuracy supervised methods, but to investigate whether an unsupervised, cost-aware approach can provide useful detection capability for organisations whose production traffic diverges from available benchmark datasets. The remainder of this paper is structured as follows: Section 2 reviews the relevant literature, Section 3 presents the proposed framework, Section 4 reports preliminary validation, Section 5 discusses implications and limitations, and Section 6 concludes.

2 Literature Review

2.1 Serverless Computing and the FaaS Billing Model

Serverless computing abstracts infrastructure management to an unprecedented degree, characterised by automatic scaling, pay-per-use billing, and built-in fault tolerance [1], [2]. The billing granularity of major FaaS platforms operates at the millisecond level; each function invocation generates a discrete billable event regardless of whether it serves a legitimate user or an attacker. Taibi et al. [15] and Shafiei et al. [16] identify this billing model as both the primary economic advantage and the principal attack surface of serverless architectures. Cold start latency optimisations [17], [18] further reduce the cost barrier for attackers by ensuring malicious invocations are processed more efficiently.

2.2 Economic Denial of Sustainability: Attack Taxonomy

The concept of EDDoS was first articulated for general cloud computing [19], with early mitigation efforts (EDoS-Shield [20], EDoS Armor [21], in-cloud scrubbing [22], EDoS-ADS [23]) focusing on IaaS environments using cryptographic puzzles, CAPTCHAs, or IP-based filtering [24], [25]. The transition to serverless-specific Denial of Wallet (DoW) attacks was formalised by Kelly et al. [4], who established a critical taxonomy: flood attacks (high-volume, short-duration bursts) and leech attacks (low-rate, long-duration invocations indistinguishable from legitimate usage).

Subsequent work has expanded the attack landscape. Bremler-Barr et al. [5] demonstrated the Yo-Yo attack exploiting auto-scaling oscillation, Xiong et al. [6], [26] introduced the Warmonger attack via shared egress IP pools, and Bremler-Barr and Czeizler [27] extended this with the Tandem attack on microservices. Mileski and Mihajloska [8] characterised distributed DoW amplification across multiple endpoints, while Joshi et al. [28] contributed a high-fidelity simulated dataset addressing the scarcity of real-world DoW data.

2.3 Limitations of Existing Detection Mechanisms

Current detection approaches can be categorised into threshold-based monitoring, supervised machine learning, and hybrid approaches. Trinh Dinh and Park [13] proposed LSTM-based dynamic thresholds for EDoS detection in SDN-based clouds, later extending this with GANs [29], though both operate at the network level and carry substantial computational overhead for serverless contexts.

Among supervised approaches, Kelly et al. [9] proposed DoWNet, converting serverless traffic logs into heat map representations classified by CNN, achieving 97.98% accuracy. However, the approach requires labelled data from their DoW Test Simulator [28] and aggregates one month of traffic into a single image, potentially limiting sensitivity to short-duration attacks. Mitra et al. [10] achieved 99.90% accuracy using Neural ODEs with Liquid Time Constants, though computational requirements may challenge latency-sensitive deployments. Hossain et al. [11] demonstrated effective feature selection with Random Forest classifiers (99.45% recall on UNSW-NB15), targeting general cloud EDoS rather than serverless-specific patterns. Additional supervised approaches include deep wavelet networks [12], ML-based feature engineering [30], attribute reduction with deep learning [31], and comparative evaluations of supervised algorithms [32].

Sommer and Paxson [33] provide a foundational critique arguing that the "closed world" assumption, where training data represents all possible traffic patterns, is violated in real-world deployments. In serverless environments, where workload patterns are non-stationary and the cost asymmetry between false negatives and false positives is acute, this limitation warrants consideration regardless of the learning approach chosen.

The dominant detection approaches share three characteristics: they predominantly employ supervised learning, operate on network-layer or system-layer metrics, and are evaluated in offline settings. To the best of our knowledge, no existing work applies unsupervised anomaly detection directly to billing-layer telemetry in serverless environments. This gap is significant: billing signals are the most direct indicator of EDDoS impact, and unsupervised methods could serve organisations whose production workloads differ from available benchmarks. The framework proposed in Section 3 addresses this gap.

2.4 Anomaly Detection and Unsupervised Learning

Chandola et al. [34] provide a comprehensive taxonomy of anomaly detection techniques (point, contextual, and collective anomalies), all relevant to EDDoS detection. Isolation Forest [14], based on the principle that anomalous data points are easier to isolate through random partitioning, offers a compelling foundation: its linear time complexity $O(t \cdot n \cdot \log \psi)$, where ψ is the sub-sampling size, and low memory footprint make it suitable for deployment alongside serverless monitoring infrastructure, unlike density-based methods requiring computation of pairwise distances.

3 Proposed Theoretical Framework

3.1 Design Principles

The proposed framework is grounded in four design principles, intended not to replace supervised approaches [9], [10] but to define a complementary direction for scenarios where labelled data is unavailable.

Principle 1: Cost-centric monitoring. Detection must operate on billing-layer signals (invocation count, execution duration, memory allocation, estimated cost) rather than network-layer metrics, aligning detection with the actual target of the attack.

Principle 2: Asynchronous analysis. Anomaly detection must be decoupled from the request-processing pipeline, consuming telemetry data asynchronously after function execution completes.

Principle 3: Unsupervised learning. The detection model should not depend on the transferability of externally generated labels. Models trained on simulated traffic [28] may not generalise to production workloads with different invocation patterns and seasonal demand cycles. Unsupervised learning establishes baselines directly from operational data, though this assumes the training period adequately represents normal behaviour.

Principle 4: DevOps compatibility. The framework must integrate into existing CI/CD pipelines and cloud-native monitoring stacks without requiring modifications to application code [35], [36].

3.2 Framework Architecture

The proposed architecture, depicted in Figure 1, consists of three logically distinct layers.

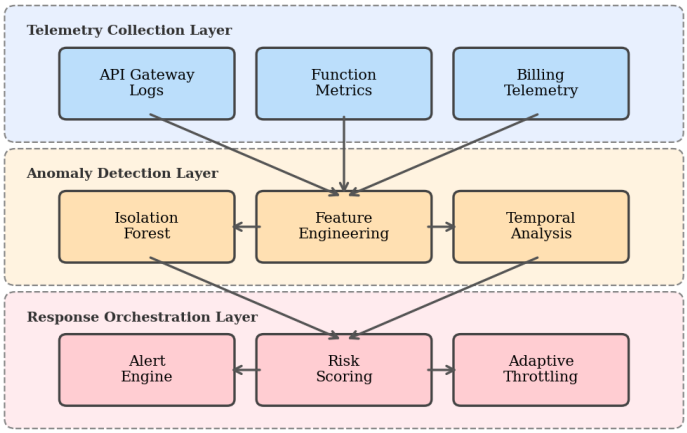

Figure 1
Proposed three-layer framework for cost-aware EDDoS detection.

The Telemetry Collection Layer aggregates three categories of metadata via existing cloud-native observability services (e.g., AWS CloudWatch, Azure Monitor): API Gateway logs (request source, method, path, timestamp), function execution metrics (invocation count, duration, memory utilisation, cold start frequency), and billing telemetry (real-time cost accumulation). Crucially, this layer does not inspect request payloads, preserving privacy and performance.

The Anomaly Detection Layer processes collected telemetry through a feature engineering module constructing temporal feature vectors: rolling averages of invocation rate, cost-per-invocation distributions, execution duration percentiles, and cross-function correlation matrices. Two complementary mechanisms are proposed. First, Isolation Forest [14] performs point anomaly detection, selected for its sub-linear time complexity and absence of density estimation requirements [34]. The model is trained exclusively on historical normal-operation data, establishing a baseline cost profile. Second, temporal pattern analysis detects slow-onset “leech” attacks [4] by computing derivatives of the cost accumulation curve and comparing against seasonal baselines.

The Response Orchestration Layer assigns composite risk scores integrating anomaly severity, estimated financial impact, and detection confidence, then applies graduated countermeasures: enhanced monitoring for low-risk anomalies, adaptive rate limiting for medium-risk, and automated budget circuit breakers for high-risk detections. This graduated approach avoids the binary accept/reject decisions of traditional WAF rules [4].

3.3 Comparison with Existing Approaches

Table 1 positions the proposed framework against existing approaches. The key differentiators are unsupervised learning, cost-centric monitoring, and asynchronous processing.

Approach	Learning Type	Target Layer	Labels Req.	Server-less	Perf.
EDoS-Shield [20]	Rule	Network	No	No	—
EDoS-ADS [23]	Rule	Network	No	No	—
Trinh & Park [13]	Sup.	Network	Yes	No	—
DoWNet [9]	Sup.	App.	Yes	Yes	97.98%
Neural ODE [10]	Sup.	System	Yes	Yes	99.90%
I-MPaFS [11]	Sup.	Network	Yes	No	99.45%†
Proposed	Unsup.	Billing	No	Yes	66.77%‡

Perf. = Accuracy unless noted. †Recall on UNSW-NB15. ‡Acc. at c=0.50; Prec. 91.47%, AUC 0.671.

Table 1
Characteristics of EDDoS/DoW detection approaches.

4 Preliminary Validation

A preliminary experiment was conducted using Isolation Forest on the publicly available DoW dataset generated by Kelly et al.'s DoWTS simulator [4], [28]. The dataset contains 187,087 records of serverless function invocations, labelled as legitimate (56,015 records, 29.9%) or attack traffic (131,072 records, 70.1%), with 12 system-level features spanning invocation timing, function execution, and infrastructure metrics. Six additional cost-centric features were engineered, including a billing cost proxy (FunctionDuration × vmmemorybucket), cost-per-round-trip-time, and invocation intensity. These 18 features were standardised prior to training.

The experiment followed a strictly unsupervised protocol: Isolation Forest was trained exclusively on legitimate traffic (70% of legitimate subset; n = 39,210), with no attack labels used. Evaluation was performed on the remaining 30% of legitimate traffic (n = 16,805) and all attack traffic (n = 131,072), totalling 147,877 test records. Table 2 summarises results across three contamination parameter settings (c). The model used 200 estimators and a sub-sampling size of 512.

c	Acc.	Prec.	Recall	F1	AUC	FPR
0.30	55.39%	93.37%	53.48%	0.680	0.671	29.6%
0.40	62.02%	92.38%	62.29%	0.744	0.671	40.1%
0.50	66.77%	91.47%	68.94%	0.786	0.671	50.2%

Table 2.
Isolation Forest detection performance on the DoW dataset [28] under unsupervised training (no attack labels used). c denotes the contamination parameter.

Precision remains consistently high (91–94%) across all configurations, indicating that flagged anomalies are correct over nine times out of ten, which is valuable where false positives carry direct business cost. The AUC-ROC of 0.671, while substantially below supervised accuracies (97–99%), represents detection meaningfully above chance (0.5) achieved without attack labels. The contamination parameter provides operators a tunable knob to balance missed detections against false alarms. Two caveats apply: the dataset contains system-level features rather than billing-layer telemetry, and the 70.1% attack ratio does not reflect production conditions where attacks constitute a small minority of traffic.

5 Discussion

The proposed framework explores an alternative detection paradigm at the intersection of cloud security and financial risk management. By shifting the detection target from network-layer metrics to billing-layer signals, it foregrounds the insight that EDDoS is fundamentally an economic attack [4]. The preliminary validation provides initial empirical support: an unsupervised Isolation Forest achieved 91.47% precision and an AUC of 0.671 without attack labels during training.

The unsupervised approach trades the high accuracy of supervised methods (97.98% for DoWNet [9], 99.90% for Neural ODE [10]) for independence from the assumption that externally generated labels transfer reliably to the target environment. The relevant question is not which paradigm is superior in absolute terms, but under what operational conditions each is preferable. Organisations whose workloads closely resemble available benchmarks [28] may be better served by supervised approaches; unsupervised methods may better suit organisations deploying novel or heterogeneous serverless applications where the gap between simulated training conditions and production reality is wide.

The asynchronous architecture preserves serverless performance characteristics by consuming telemetry after function execution rather than inspecting requests inline [17], [18]. However, this introduces detection latency: for slow-onset leech attacks,

significant financial damage could accrue before an alert is raised. The graduated response model recognises that blocking legitimate invocations has direct business impact, requiring risk-scoring that accounts for anomaly severity, financial exposure, and detection confidence.

From an enterprise management perspective, EDDoS represents a risk class between traditional IT security and financial governance. EDDoS attacks may only surface during end-of-month billing reconciliation, often well after mitigation is possible. The framework is deliberately lightweight, reusing existing cloud-native observability services to lower adoption barriers for organisations with limited security budgets. The choice between supervised and unsupervised paradigms is ultimately a risk management decision shaped by organisational context, workload maturity, and acceptable financial exposure.

A methodological consideration warrants explicit discussion. The preliminary validation employs system-level features from the DoW dataset rather than raw billing telemetry obtained from a cloud provider. Beyond the absence of publicly available billing-layer datasets for EDDoS research, this choice rests on a structural property of serverless pricing: billing in FaaS environments is a deterministic function of the system-level metrics already captured in the dataset. Invocation count, execution duration, and allocated memory map directly to cost through the provider's published pricing formula; for example, AWS Lambda charges $0.20 per million requests plus $0.0000166667 per GB-second of compute time. Consequently, anomaly patterns detected at the system level correspond to proportional anomalies in the resulting bill: a function exhibiting anomalous invocation frequency or duration will, by mathematical necessity, produce an anomalous cost signal. The system-level features therefore serve as a structurally valid upstream proxy for billing telemetry in this preliminary investigation.

Nevertheless, production billing data introduces complexities not captured by this proxy, including tiered pricing structures, reserved concurrency discounts, cross-service cost aggregation, and billing API reporting latency, all of which could affect detection performance in ways that system-level proxies cannot anticipate. Validating the framework against actual billing telemetry from a controlled testbed, using AWS Lambda or an open-source FaaS platform such as OpenFaaS, therefore constitutes the immediate priority for future empirical work and will form a central component of the authors' ongoing doctoral research programme.

Additional limitations must be acknowledged. The dataset's 70.1% attack ratio does not reflect production conditions where attacks constitute a small minority of traffic; performance under realistic base rates remains to be established. The framework assumes near-real-time billing telemetry availability and attack-free training periods; a contaminated training set would undermine the unsupervised baseline. Adversarial evasion strategies are not addressed. Future work should evaluate under realistic base rates and conduct a comparative analysis against supervised methods [9], [10] on the same dataset.

Conclusions

This paper has examined EDDoS attacks against serverless computing and identified the dominance of supervised detection methods operating at network and system layers, achieving up to 99.90% accuracy. A conceptual framework has been proposed exploring an alternative approach through cost-centric monitoring, asynchronous analysis, unsupervised learning, and DevOps compatibility. Preliminary evidence supports viability: unsupervised Isolation Forest achieved 91.47% precision and an AUC-ROC of 0.671 without attack labels.

Financial sustainability deserves recognition as a first-class security objective alongside availability. The detection literature would benefit from unsupervised approaches operating where supervised methods face transferability limitations. For enterprise stakeholders, the choice between paradigms is a risk management decision shaped by organisational context and acceptable financial exposure. Empirical validation on billing-layer telemetry is the necessary next step.

References

[1] E. Jonas, J. Schleier-Smith, V. Sreekanti, C.-Y. Tsai, A. Khandelwal, Q. Pu, V. Shankar, J. Carreira, K. Krauth, N. J. Yadwadkar, J. E. Gonzalez: Cloud Programming Simplified: A Berkeley View on Serverless Computing, arXiv Technical Report, arXiv:1902.03383, 2019

[2] J. Schleier-Smith, V. Sreekanti, A. Khandelwal, J. Carreira, N. J. Yadwadkar, J. E. Gonzalez: What Serverless Computing Is and Should Become, Communications of the ACM, vol. 64, no. 5, 2021

[3] P. Castro, V. Ishakian, V. Muthusamy, A. Slominski: The Rise of Serverless Computing, Communications of the ACM, vol. 62, no. 12, 2019

[4] D. Kelly, F. G. Glavin, E. Barrett: Denial of Wallet—Defining a Looming Threat to Serverless Computing, Journal of Information Security and Applications, vol. 60, p. 102843, 2021

[5] A. Bremler-Barr, E. Brosh, M. Sides: DDoS Attack on Cloud Auto-scaling Mechanisms, in Proceedings of IEEE INFOCOM, 2017

[6] J. Xiong, M. Wei, Z. Lu, Y. Liu: Warmonger: Inflicting Denial-of-Service via Serverless Functions in the Cloud, in Proceedings of ACM CCS, 2021

[7] J. Mirkovic, P. Reiher: A Taxonomy of DDoS Attack and DDoS Defense Mechanisms, ACM SIGCOMM Computer Communication Review, vol. 34, no. 2, 2004

[8] D. Mileski, H. Mihajloska: Distributed Denial of Wallet Attack on Serverless Pay-as-you-go Model, in Proceedings of 2022 30th Telecommunications Forum (TELFOR), IEEE, 2022

[9] D. Kelly, F. G. Glavin, E. Barrett: DoWNet—Classification of Denial-of-Wallet Attacks on Serverless Application Traffic, Journal of Cybersecurity, vol. 10, no. 1, p. tyae004, 2024

[10] D. Mitra, N. Chaki, M. Bhuyan: Detecting Denial of Wallet Attacks in Serverless Computing: A Neural ODE-LTC Approach, in Proceedings of 2025 IEEE International Conference on Joint Cloud Computing (JCC), IEEE, 2025

[11] M. S. Hossain, M. A. Hossain, M. S. Islam: I-MPaFS: Enhancing EDoS Attack Detection in Cloud Through a Data-driven Approach, Journal of Cloud Computing, vol. 13, no. 1, p. 117, 2024

[12] P. Renukadevi, S. Amaran, A. Vikram, T. P. Rao, M. K. Ishak: Enhancing Cybersecurity Through Fusion of Optimization With Deep Wavelet Neural Networks on Denial of Wallet Attack Detection in Serverless Computing, IEEE Access, vol. 13, pp. 44927–44940, 2025

[13] P. Trinh Dinh, M. Park: Dynamic Economic-Denial-of-Sustainability (EDoS) Detection in SDN-based Cloud, in Proceedings of 2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC), IEEE, 2020, pp. 64–69

[14] F. T. Liu, K. M. Ting, Z.-H. Zhou: Isolation Forest, in Proceedings of IEEE International Conference on Data Mining (ICDM), 2008

[15] D. Taibi, N. El Ioini, C. Pahl, J. Niederkofler: Patterns for Serverless Functions (FaaS): A Multivocal Literature Review, in Proceedings of CLOSER (SCITEPRESS), 2020

[16] H. Shafiei, A. Khonsari, P. Mousavi: Serverless Computing: A Survey of Opportunities, Challenges and Applications, engrXiv, 2020

[17] A. Ebrahimi, M. Ghobaei-Arani, H. Saboohi: Cold Start Latency Mitigation Mechanisms in Serverless Computing: Taxonomy, Review, and Future Directions, Journal of Systems Architecture (Elsevier), 2024

[18] M. Ghorbian, M. Ghobaei-Arani: A Survey on the Cold Start Latency Approaches in Serverless Computing: An Optimization-based Perspective, Computing (Springer), 2024

[19] S. VivinSandar, S. Shenai: Economic Denial of Sustainability in Cloud Services Using HTTP and XML Based DDoS Attacks, IJCA, 2012

[20] M. H. Sqalli, F. Al-Haidari, K. Salah: EDoS-Shield—A Two-Steps Mitigation Technique Against EDoS Attacks in Cloud Computing, in Proceedings of IEEE UCC, 2011

[21] M. Masood, Z. Anwar, S. A. Raza, M. A. Hur: EDoS Armor: A Cost Effective EDoS Attack Mitigation Framework for E-commerce Applications in Cloud, in Proceedings of IEEE INMIC, 2013

[22] M. Naresh Kumar et al.: Mitigating EDoS in Cloud Computing Using In-cloud Scrubber Service, in Proceedings of IEEE CICN, 2012

[23] A. Shawahna, M. Abu-Amara, A. Mahmoud, Y. E. Osais: EDoS-ADS: An Enhanced Mitigation Technique Against EDoS Attacks, IEEE Transactions on Cloud Computing, 2018

[24] P. Singh, S. Manickam, S. U. Rehman: A Survey of Mitigation Techniques Against EDoS Attack on Cloud Computing Architecture, in Proceedings of IEEE ICRITO, 2014

[25] Z. A. Baig, F. Binbeshr: Controlled Virtual Resource Access to Mitigate EDoS Attacks Against Cloud Infrastructures, in Proceedings of IEEE CloudCom-Asia, 2013

[26] J. Xiong, M. Wei, Z. Lu, Y. Liu: Warmonger Attack: A Novel Attack Vector in Serverless Computing, IEEE/ACM Transactions on Networking, vol. 32, no. 6, pp. 4758–4773, 2024

[27] A. Bremler-Barr, M. Czeizler: Tandem Attack: DDoS Attack on Microservices Auto-scaling Mechanisms, in Proceedings of IEEE INFOCOM Workshops, 2023

[28] C. Joshi et al.: High-Fidelity Simulated Dataset for Enhanced Detection of DoW Attacks in Serverless Architecture, in Proceedings of IEEE ICBDS, 2024

[29] P. Trinh Dinh, M. Park: EDoS Detection Using GANs in SDN-based Cloud, in Proceedings of IEEE ICCE, 2021

[30] A. Mohammadi Ghaleh, S. Sedighian Kashi: Machine Learning Based Detection of Denial of Wallet Attacks in Serverless Computing, SN Computer Science, 2025

[31] A. K. Alkhalifa, M. Aljebreen, R. Alanazi, N. Ahmad, S. Alahmari, O. Alrusaini, A. Alqazzaz, H. Alkhiri: Mitigating Malicious Denial of Wallet Attack Using Attribute Reduction With Deep Learning Approach for Serverless Computing on Next Generation Applications, Scientific Reports, vol. 15, p. 18024, 2025

[32] T. Kasuba, S. Saravanan, V. V. S. S. S. Balaram: Intelligent Threat Detection Framework for Serverless Cloud Computing Using Supervised ML Algorithms, OPSEARCH, 2025

[33] R. Sommer, V. Paxson: Outside the Closed World: On Using Machine Learning for Network Intrusion Detection, in Proceedings of IEEE Symposium on Security and Privacy (S&P), 2010

[34] V. Chandola, A. Banerjee, V. Kumar: Anomaly Detection: A Survey, ACM Computing Surveys, vol. 41, no. 3, 2009

[35] H. Myrbakken, R. Colomo-Palacios: DevSecOps: A Multivocal Literature Review, in Proceedings of Software Quality Days (SWQD), 2017

[36] R. N. Rajapakse, M. Zahedi, M. A. Babar, H. Shen: Challenges and Solutions When Adopting DevSecOps: A Systematic Review, Information and Software Technology, vol. 141, 2022