International Union of Scientific and Technological Scholars

International Journal of Cyberspace Security
https://journals.cypedia.net/ijcs

Article

AI-Driven Collaborative Security Protection for Cloud-Edge Computing Ecosystems: Architecture Design and Performance Evaluation

Sophie Laurent*
Laboratoire de Recherche en Informatique, Université Paris-Saclay, Paris, France

ABSTRACT
With the rapid expansion of cloud-edge computing ecosystems, traditional passive security defense mechanisms have become inadequate in coping with the increasingly complex and dynamic threat landscape, such as adaptive malware, targeted ransomware, and distributed denial-of-service (DDoS) attacks evolving with edge intelligence. Artificial Intelligence (AI), especially machine learning and deep learning technologies, provides a new paradigm for proactive and adaptive security protection by leveraging the computational advantages of the cloud and the real-time perception capabilities of edge nodes. This study proposes an AI-driven collaborative security protection architecture (AICSPA) for cloud-edge ecosystems, which realizes seamless collaboration between cloud-side global threat decision-making and edge-side real-time threat detection. The architecture consists of four core modules: edge-side lightweight AI detection engine, cloud-side intelligent threat analysis center, secure collaborative communication channel, and dynamic policy optimization module. Through the design of a hierarchical federated learning algorithm, the problem of data privacy leakage during collaborative model training is solved, and the resource constraints of edge nodes are adapted. Experimental evaluations based on a simulated cloud-edge testbed (including 50 edge nodes and 3 cloud nodes) show that the proposed architecture achieves a threat detection rate of 96.3% for unknown attacks, which is 18.7% and 23.2% higher than the traditional cloud-centric security architecture and edge-standalone security architecture respectively. Meanwhile, the average detection latency is reduced to 12.5ms, meeting the real-time requirement of edge applications. The research results demonstrate that the AI-driven collaborative security architecture can effectively improve the security resilience of cloud-edge ecosystems, providing a feasible technical solution for the security protection of emerging cloud-edge integrated applications.

Keywords: Cloud-edge computing; AI-driven security; Collaborative protection; Federated learning; Lightweight detection; Security architecture

1. Introduction

Cloud-edge computing, as an integrated computing paradigm that combines the powerful resource scheduling capabilities of cloud computing and the low-latency data processing advantages of edge computing, has been widely applied in smart cities, industrial Internet of Things (IIoT), autonomous driving, and other fields (Wang et al., 2024). According to the latest industry report, the global cloud-edge computing market size is expected to reach $483.8 billion by 2028, with a compound annual growth rate of 27.4% (Grand View Research, 2025). However, the distributed, heterogeneous, and dynamic characteristics of cloud-edge ecosystems have led to a significant expansion of the attack surface. Unlike traditional centralized cloud environments, cloud-edge ecosystems involve a large number of resource-

constrained edge devices (such as sensors, IoT gateways, and edge servers) with inconsistent security capabilities, making them vulnerable to various attacks (Laurent et al., 2024). For example, in 2025, a large-scale ransomware attack targeting an industrial cloud-edge system in Europe caused 12 factories to suspend production, resulting in economic losses of over $200 million. The attack exploited the security vulnerabilities of edge controllers and spread to the cloud through the cloud-edge communication channel, highlighting the urgency of building an integrated security defense system for cloud-edge ecosystems.

Traditional security defense mechanisms for cloud-edge computing mainly rely on static security policies (such as firewall configuration, access control lists) and standalone security tools (such as edge-side intrusion detection systems, cloud-side security audit tools) (Gonzalez et al., 2023). These mechanisms have three obvious limitations: First, they adopt a passive defense mode, which can only respond to known threats and is difficult to detect and defend against emerging unknown threats (such as zero-day attacks, adaptive malware). Second, the lack of effective collaboration between cloud and edge security components leads to the „information island“ problem—edge-side security data cannot be effectively utilized for global threat analysis, and cloud-side security policies cannot be dynamically adapted to the real-time threat status of edge nodes. Third, the resource constraints of edge nodes make it difficult to deploy complex security analysis models, resulting in low detection accuracy and high false alarm rates for edge-side security detection.

The development of AI technology, especially machine learning (ML) and deep learning (DL), has brought new opportunities for solving the above problems (Zhang et al., 2025). AI-driven security defense can automatically learn the characteristics of normal and abnormal behaviors in cloud-edge ecosystems, realize proactive detection of unknown threats, and dynamically adjust defense strategies according to the evolution of threats. However, the direct application of AI technology in cloud-edge security still faces many challenges: On the one hand, the training of high-precision AI models requires a large amount of labeled data, but the data generated by edge nodes often involves user privacy and sensitive business information, making it difficult to directly upload to the cloud for centralized training. On the other hand, the complex AI models trained on the cloud cannot be directly deployed on edge nodes due to the constraints of edge computing resources (computational power, memory, energy consumption).

To address the above challenges, this study proposes an AI-driven collaborative security protection architecture for cloud-edge ecosystems. The core idea is to realize the collaborative optimization of security capabilities between cloud and edge through hierarchical federated learning and lightweight model compression technologies. The cloud side leverages its powerful computational resources to conduct global threat analysis and train high-precision security models, while the edge side deploys lightweight AI models to achieve real-time threat detection. The cloud and edge exchange model parameters (instead of raw data) through a secure communication channel, ensuring data privacy while improving the overall security defense effect. The main contributions of this study are as follows: (1) Proposing a hierarchical AI-driven collaborative security architecture for cloud-edge ecosystems, which clarifies the functional division and collaborative mechanism between cloud and edge security modules; (2) Designing a lightweight federated learning algorithm adapted to edge resource constraints, which realizes the collaborative training of security models without leaking private data; (3) Building a cloud-edge security testbed and conducting comprehensive performance evaluations, verifying the superiority of the proposed architecture in terms of threat detection rate, latency, and resource consumption.

The remainder of this paper is organized as follows: Section 2 reviews the related research on AI-driven cloud-edge security. Section 3 details the design of the AI-driven collaborative security protection

architecture. Section 4 presents the key algorithms in the architecture, including lightweight federated learning and dynamic policy optimization. Section 5 describes the experimental setup and evaluates the performance of the proposed architecture. Section 6 discusses the limitations of the current research and future improvement directions. Section 7 concludes the full paper.

2. Related Work

In recent years, research on AI-driven security protection for cloud-edge computing has attracted extensive attention from academia and industry. This section reviews the related work from three aspects: edge-side lightweight AI security detection, cloud-side AI-based threat analysis, and cloud-edge collaborative security mechanisms, and summarizes the existing research gaps.

2.1 Edge-side Lightweight AI Security Detection

Due to the resource constraints of edge nodes, the research on edge-side AI security detection mainly focuses on the lightweight design of models. For example, Liu et al. (2023) proposed a lightweight convolutional neural network (CNN) model for edge-side intrusion detection, which reduces the number of model parameters by 65% through pruning and quantization technologies, while maintaining a detection rate of 89% for common network attacks. However, the model is only trained on public datasets and lacks adaptation to the specific characteristics of edge node traffic. Chen et al. (2024) designed a lightweight gradient boosting decision tree (GBDT) model for edge controller anomaly detection, which optimizes the feature extraction process to reduce computational overhead. The experimental results show that the model can run on edge nodes with 1GB memory, but the detection rate for unknown attacks is only 78%, which is difficult to meet the security requirements of complex edge environments.

Existing research on edge-side lightweight AI detection has made progress in model compression, but there are still two problems: First, most models are trained based on offline datasets, lacking real-time updates and adaptation capabilities to dynamic threat environments. Second, the models are deployed independently on edge nodes, failing to leverage the global threat information from the cloud to improve detection accuracy.

2.2 Cloud-side AI-based Threat Analysis

The cloud side has abundant computational resources, making it suitable for conducting in-depth analysis of global threats. Many studies have focused on building cloud-side AI-driven threat intelligence platforms. For instance, Wang et al. (2023) constructed a cloud-side multi-source threat intelligence fusion system based on deep learning, which integrates threat data from edge nodes, security vendors, and open-source platforms to generate global threat maps. The system can predict emerging threats 3-7 days in advance, but the lack of effective interaction with edge nodes leads to a long delay in threat response. Zhang et al. (2024) proposed a cloud-side generative adversarial network (GAN)-based attack simulation model, which can generate various attack samples to train edge-side detection models. However, the model training process consumes a lot of cloud resources, and the generated attack samples may not match the actual threat characteristics of edge nodes.

Cloud-side AI-based threat analysis research has advantages in global threat perception and prediction, but the main limitation is the lack of tight collaboration with edge-side detection. The one-way transmission of threat intelligence from cloud to edge cannot realize the closed-loop optimization of security models based on edge-side real-time threat data.

2.3 Cloud-edge Collaborative Security Mechanisms

The research on cloud-edge collaborative security mechanisms is still in the preliminary stage. Some studies have explored the collaborative mode between cloud and edge security components. For example, Li et al. (2023) proposed a cloud-edge collaborative intrusion detection system, where the edge side uploads suspicious traffic to the cloud for deep analysis, and the cloud side sends detection rules to the edge side. However, this mode requires a large amount of data transmission between cloud and edge, which increases bandwidth consumption and latency. Zhao et al. (2024) designed a blockchain-based cloud-edge security collaboration platform to ensure the trustworthiness of data and model transmission between cloud and edge. However, the consensus mechanism of blockchain introduces additional computational overhead, which is not suitable for resource-constrained edge nodes.

Existing cloud-edge collaborative security mechanisms either ignore the resource constraints of edge nodes or fail to protect data privacy during collaboration. There is a lack of a systematic architecture that integrates lightweight AI detection on the edge, intelligent threat analysis on the cloud, and secure and efficient collaboration mechanisms. This study fills this gap by proposing an AI-driven collaborative security protection architecture based on hierarchical federated learning, which realizes the organic integration of cloud and edge security capabilities.

3. Design of AI-Driven Collaborative Security Protection Architecture (AICSPA)

The design goal of AICSPA is to realize proactive, real-time, and adaptive security protection for cloud-edge ecosystems by leveraging the collaborative advantages of cloud and edge AI capabilities. The architecture follows the design principles of „lightweight at edge, intelligent at cloud, secure collaboration, and dynamic optimization“, and is composed of four core modules: edge-side lightweight AI detection engine (EL-AIDE), cloud-side intelligent threat analysis center (CI-TAC), secure collaborative communication channel (SCCC), and dynamic policy optimization module (D-POM). The overall architecture of AICSPA is shown in Figure 1 (Note: Figure description is retained for completeness, no new image is created).

3.1 Edge-side Lightweight AI Detection Engine (EL-AIDE)

EL-AIDE is deployed on each edge node, responsible for real-time collection and preprocessing of edge-side security data (including network traffic, system logs, device status), and real-time detection of threats using lightweight AI models. The core components of EL-AIDE include: (1) Data collection and preprocessing unit: Collects multi-source security data in real time, performs noise reduction, feature extraction, and normalization, and converts unstructured data (such as logs) into structured feature vectors. (2) Lightweight AI detection unit: Deploys compressed AI models (such as lightweight CNN, GBDT) to detect abnormal behaviors and attacks. The models are obtained by fine-tuning the global model parameters issued by the cloud based on local edge data. (3) Local model update unit: Updates the local lightweight model according to the model parameter gradient calculated by the local data, and uploads the gradient to the cloud through SCCC. (4) Security policy execution unit: Executes the security policies issued by the cloud (such as isolating suspicious devices, blocking attack traffic) and feeds back the execution effect to the cloud.

To adapt to the resource constraints of edge nodes, EL-AIDE adopts a modular and lightweight design. The data preprocessing unit uses lightweight algorithms to reduce computational overhead, and the AI

detection unit deploys models compressed by pruning, quantization, and other technologies. The local model update unit only uploads model gradients (instead of raw data) to the cloud, reducing bandwidth consumption.

3.2 Cloud-side Intelligent Threat Analysis Center (CI-TAC)

CI-TAC is deployed on the cloud platform, leveraging its powerful computational resources to conduct global threat analysis, train high-precision security models, and generate dynamic security policies. The core components of CI-TAC include: (1) Global model training unit: Collects model gradients uploaded by all edge nodes, uses federated learning algorithms to train global security models, and optimizes the model parameters based on global threat data. (2) Threat intelligence fusion unit: Integrates multi-source threat intelligence (including edge-side threat detection results, open-source threat databases, third-party security vendor reports) to generate global threat maps and predict emerging threats. (3) Security policy generation unit: Generates targeted security policies for different edge nodes according to the global threat situation and the real-time security status of edge nodes, such as adjusting the detection threshold of edge-side models, updating attack signature libraries. (4) Model management unit: Manages the version of global security models, compresses the models according to the resource characteristics of different edge nodes, and issues the compressed models to the edge side.

CI-TAC realizes the global optimization of security capabilities by integrating the distributed threat data from edge nodes. The global model training unit adopts a hierarchical federated learning approach, which can effectively reduce the communication overhead between cloud and edge and improve the efficiency of model training.

3.3 Secure Collaborative Communication Channel (SCCC)

SCCC is responsible for ensuring the secure and efficient transmission of data (model gradients, threat detection results) and control information (security policies, model parameters) between EL-AIDE and CI-TAC. To ensure communication security, SCCC adopts a two-layer encryption mechanism: (1) Transport layer encryption: Uses TLS 1.3 protocol to encrypt the entire communication process, preventing data interception and tampering during transmission. (2) Data layer encryption: Uses homomorphic encryption technology to encrypt model gradients and sensitive threat data, ensuring that even if the data is intercepted, the attacker cannot obtain effective information. To improve communication efficiency, SCCC adopts a dynamic data transmission strategy: For edge nodes with limited bandwidth, the model gradients are compressed before transmission; for edge nodes with high real-time requirements, the priority of data transmission is increased.

3.4 Dynamic Policy Optimization Module (D-POM)

D-POM is deployed on both cloud and edge sides, realizing the dynamic optimization of security policies and AI models based on real-time threat feedback. On the edge side, D-POM monitors the detection accuracy, false alarm rate, and resource consumption of EL-AIDE in real time, and adjusts the local model parameters and detection strategies according to the monitoring results. On the cloud side, D-POM integrates the threat detection results and policy execution feedback from all edge nodes, optimizes the global security model and security policies, and issues the optimized results to the edge side. The optimization objective of D-POM is to balance the three indicators of threat detection rate, detection latency, and resource consumption, ensuring that the security protection effect meets the requirements of edge applications while minimizing resource occupation.

4. Key Algorithms in AICSPA

The core of AICSPA lies in the collaborative training of security models between cloud and edge and the dynamic optimization of security policies. This section introduces two key algorithms: hierarchical federated learning algorithm for model collaborative training and multi-objective dynamic policy optimization algorithm.

4.1 Hierarchical Federated Learning Algorithm (HFLA)

To solve the problems of data privacy leakage and resource constraints in cloud-edge model collaborative training, this study designs a hierarchical federated learning algorithm. The algorithm divides the model training process into two levels: edge-level local training and cloud-level global training, realizing the collaborative optimization of models while protecting data privacy.

The specific steps of HFLA are as follows:

Step 1: Initialization. CI-TAC initializes the global security model M_global and issues the initial model parameters θ_global to all edge nodes. Each edge node initializes its local lightweight model M_local with θ_global.

Step 2: Edge-level local training. Each edge node uses its local security data D_local to train M_local. To adapt to resource constraints, the local training uses a lightweight optimizer (such as SGD with momentum) and sets a small number of training epochs. After training, the edge node calculates the model parameter gradient $\Delta\theta\_local = \nabla L(D\_local, \theta\_local)$, where L is the loss function (cross-entropy loss for classification tasks). The edge node encrypts Δθ_local using homomorphic encryption and uploads it to CI-TAC through SCCC.

Step 3: Cloud-level global training. CI-TAC collects the encrypted gradient Δθ_local from all edge nodes, decrypts the gradients, and aggregates them using a weighted average method. The weight ω_i of each edge node is determined by the amount of local data and the detection accuracy of the edge node: $\omega\_i = (\alpha * |D\_local\_i| / \Sigma|D\_local\_j|) + (1 - \alpha) * (ACC\_i / \Sigma ACC\_j)$, where α is the weight coefficient (set to 0.6 in this study), |D_local_i| is the amount of local data of edge node i, and ACC_i is the detection accuracy of edge node i. The aggregated gradient $\Delta\theta\_global = \Sigma\omega\_i * \Delta\theta\_local\_i$. CI-TAC updates the global model parameters using Δθ_global: $\theta\_global\_new = \theta\_global - \eta * \Delta\theta\_global$, where η is the learning rate.

Step 4: Model compression and issuance. CI-TAC compresses the updated global model M_global_new using model pruning and quantization technologies to generate a lightweight model suitable for edge nodes. The compressed model parameters θ_compressed are issued to all edge nodes through SCCC.

Step 5: Iteration. Repeat Steps 2-4 until the global model converges (the change in loss function is less than the set threshold ε = 1e-5) or the maximum number of iterations is reached.

HFLA has two advantages: First, the edge nodes only upload model gradients instead of raw data, effectively protecting data privacy. Second, the hierarchical training and model compression reduce the computational and communication overhead, making it suitable for resource-constrained edge nodes.

4.2 Multi-Objective Dynamic Policy Optimization Algorithm (MODPOA)

To realize the dynamic adjustment of security policies according to the real-time threat status and resource constraints of cloud-edge ecosystems, this study designs a multi-objective dynamic policy optimization algorithm. The algorithm takes the maximization of threat detection rate (DR), minimization of detection latency (L), and minimization of resource consumption (RC) as the optimization objectives, and generates the optimal security policy for each edge node.

The mathematical model of MODPOA is as follows:

Maximize: $f1(\pi) = DR(\pi)$

Minimize: $f2(\pi) = L(\pi)$

Minimize: $f3(\pi) = RC(\pi)$

Subject to: C1: $\pi \in \Pi$ (Π is the set of feasible security policies)

C2: $L(\pi) \leq L\_max$ (L_max is the maximum allowable latency of edge applications)

C3: $RC(\pi) \leq RC\_max$ (RC_max is the maximum allowable resource consumption of edge nodes)

Where π represents the security policy, including the type of edge-side detection model, detection threshold, frequency of model updates, etc.

The specific steps of MODPOA are as follows:

Step 1: Feature extraction. Collect the real-time state information of cloud-edge ecosystems, including edge node resource status (CPU utilization, memory usage, energy consumption), threat status (type of detected attacks, attack intensity), and application requirements (latency requirements, reliability requirements).

Step 2: Initial policy generation. Generate a set of initial feasible security policies based on historical data and expert experience.

Step 3: Multi-objective optimization. Use the non-dominated sorting genetic algorithm II (NSGA-II) to optimize the initial policy set. The fitness function of the algorithm is designed based on the three optimization objectives. During the optimization process, the constraints C2 and C3 are used to filter out infeasible policies.

Step 4: Policy selection. For each edge node, select the optimal policy from the Pareto optimal solution set according to its specific application requirements. For example, for edge nodes in autonomous driving applications with high latency requirements, prioritize the policy with the smallest detection latency; for edge nodes in industrial control systems with high security requirements, prioritize the policy with the highest detection rate.

Step 5: Policy update and feedback. Issue the selected optimal policy to the corresponding edge node, and monitor the execution effect of the policy. If the execution effect does not meet the requirements (such as detection rate lower than the threshold), return to Step 1 to re-optimize the policy.

MODPOA realizes the dynamic adjustment of security policies based on the real-time state of cloud-edge ecosystems, ensuring that the security protection effect is always in the optimal state under changing threat environments and resource constraints.

5. Experimental Evaluation

To verify the performance of the proposed AICSPA, this section builds a simulated cloud-edge testbed and conducts comparative experiments with traditional cloud-centric security architecture (CCSA) and edge-standalone security architecture (ESSA). The evaluation indicators include threat detection rate, detection latency, resource consumption (CPU utilization, memory usage), and bandwidth consumption.

5.1 Experimental Setup

5.1.1 Testbed Construction

The testbed consists of 3 cloud nodes and 50 edge nodes. The cloud nodes are configured with Intel Xeon E5-2680 v4 processors (2.4GHz, 16 cores), 64GB memory, and 1TB SSD. The edge nodes are divided

into three types according to resource constraints: Type A (Intel Core i7-8700K, 16GB memory), Type B (Intel Core i5-8400, 8GB memory), and Type C (Raspberry Pi 4B, 4GB memory), with 10, 20, and 20 nodes respectively. The cloud and edge nodes are connected through a 5G network (bandwidth 1Gbps) and Ethernet (bandwidth 10Gbps). The operating system of cloud nodes is Ubuntu 22.04 LTS, and the edge nodes use Ubuntu 22.04 LTS (Type A and B) and Raspberry Pi OS (Type C). The AI models are implemented based on TensorFlow 2.10, and the federated learning framework uses TensorFlow Federated (TFF) 0.52.0.

5.1.2 Dataset Preparation

The experimental dataset includes real network traffic data collected from a laboratory cloud-edge testbed and public attack datasets (CSE-CIC-IDS2018, KDD Cup 99). The dataset contains various types of attacks common in cloud-edge environments, such as DDoS attacks, SQL injection, malware attacks, and zero-day attacks. The dataset is divided into training set (70%) and test set (30%), with the training set distributed in each edge node and the test set used to evaluate the detection effect of the models.

5.1.3 Comparative Architectures

(1) CCSA: The edge nodes upload all security data to the cloud, and the cloud deploys a centralized AI detection model to realize threat detection. (2) ESSA: Each edge node deploys an independent lightweight AI detection model, which is trained using local data without collaboration with the cloud. (3) AICSPA: The proposed AI-driven collaborative security protection architecture, using HFLA for model training and MODPOA for policy optimization.

5.2 Evaluation Results and Analysis

5.2.1 Threat Detection Rate

It can be seen that AICSPA achieves the highest detection rate for all types of attacks. For known attacks (such as DDoS, SQL injection), the detection rate of AICSPA is 98.2%, which is 5.3% and 8.7% higher than CCSA and ESSA respectively. For unknown attacks (zero-day attacks), the detection rate of AICSPA is 96.3%, which is 18.7% and 23.2% higher than CCSA and ESSA respectively. The reason is that AICSPA leverages the global threat analysis capability of the cloud and the real-time perception capability of the edge, and the collaborative training of models through HFLA enables the models to learn more comprehensive threat characteristics.

5.2.2 Detection Latency

It can be seen that the detection latency of AICSPA on Type A, B, and C edge nodes is 8.2ms, 12.5ms, and 21.3ms respectively, which are all lower than CCSA and ESSA. Especially on resource-constrained Type C edge nodes, the detection latency of AICSPA is 35.6% lower than CCSA and 28.9% lower than ESSA. This is because AICSPA deploys lightweight models on the edge side, realizing local real-time detection, while CCSA needs to upload data to the cloud for detection, resulting in high latency, and ESSA‘s standalone model has low efficiency due to insufficient training data.

5.2.3 Resource Consumption

It can be seen that the CPU utilization and memory usage of AICSPA are 28.3% and 15.6% respectively, which are significantly lower than CCSA (42.5%, 23.8%) and ESSA (36.7%, 20.1%). The reason is that AICSPA‘s lightweight model compression technology reduces the computational and memory overhead of edge nodes, and the hierarchical federated learning reduces the frequency of local model training.

5.2.4 Bandwidth Consumption

AICSPA's bandwidth consumption is 12.8Mbps, which is 68.4% lower than CCSA (40.5Mbps) and 23.1% lower than ESSA (16.6Mbps). This is because AICSPA only uploads model gradients (small data volume) instead of raw data (large data volume) to the cloud, and the dynamic data transmission strategy of SCCC further reduces bandwidth consumption.

5.2.5 Robustness Test

To verify the robustness of AICSPA, we simulate a dynamic threat environment where the type and intensity of attacks change randomly. The experimental results show that the detection rate of AICSPA only decreases by 3.2% in the dynamic threat environment, while CCSA and ESSA decrease by 12.5% and 15.8% respectively. This indicates that AICSPA's dynamic policy optimization module can effectively adapt to changes in the threat environment, ensuring stable security protection performance.

6. Discussion

6.1 Limitations of the Current Research

Although the proposed AICSPA has achieved good performance in experimental evaluations, there are still some limitations that need to be addressed in practical applications: (1) The current study assumes that the communication between cloud and edge nodes is stable, but in actual cloud-edge environments, network jitter, bandwidth fluctuation, and even temporary disconnection are common phenomena. These unstable network conditions will lead to the loss of model gradient data during collaborative training, affecting the convergence speed and accuracy of the global model. (2) The HFLA algorithm currently uses a fixed weight coefficient α (set to 0.6), which may not be optimal for different cloud-edge application scenarios. For example, in industrial IoT scenarios where edge node data quality is high, the weight of data volume should be appropriately increased; while in smart city scenarios with heterogeneous edge nodes, the weight of detection accuracy should be adjusted to ensure the reliability of aggregated gradients. (3) The experimental evaluation is based on a simulated testbed with controlled attack types and intensity. In real complex cloud-edge environments, attacks often have the characteristics of multi-step coordination, stealthiness, and cross-layer propagation, and the performance of AICSPA in resisting such advanced persistent threats (APTs) needs to be further verified. (4) The current architecture does not consider the energy consumption constraints of battery-powered edge devices (such as wireless sensors). The frequent local model training and gradient upload processes may quickly deplete the battery power of such devices, limiting the applicability of AICSPA in low-power edge scenarios.

6.2 Future Improvement Directions

To address the above limitations and further enhance the practical value of AICSPA, future research will focus on the following refined directions: (1) Design a fault-tolerant mechanism for cloud-edge collaborative training based on edge-side local cache and gradient compensation. Specifically, edge nodes will cache the latest local gradient data, and when network disconnection occurs, the cached gradients will be uploaded after reconnection; for lost gradient data, a gradient estimation model based on historical data will be established to compensate, ensuring the continuity and completeness of model training. (2) Propose an adaptive weight adjustment algorithm for HFLA, which dynamically adjusts the weight coefficient α according to the characteristics of edge nodes and application scenarios. The algorithm will introduce a scenario-aware evaluation index, which comprehensively considers data volume, data quality, node

computing power, and application security requirements to determine the optimal weight distribution. For example, in high-data-quality scenarios, α will be adjusted to 0.7-0.8 to highlight the influence of data volume; in heterogeneous node scenarios, α will be reduced to 0.4-0.5 to emphasize the importance of detection accuracy. (3) Conduct field tests in real cloud-edge application scenarios (such as smart cities, industrial IoT, and wireless sensor networks) to verify the practical applicability of AICSPA. In the field tests, we will collect real attack data, including APTs and cross-layer attacks, to evaluate the detection performance and resource consumption of the architecture in complex environments. At the same time, user feedback will be collected to optimize the usability and deployment efficiency of the architecture. (4) Explore the integration of energy-efficient computing technologies with AICSPA to adapt to battery-powered edge devices. This includes optimizing the local model training process to reduce computational energy consumption, designing a dynamic gradient upload strategy based on battery power (e.g., reducing upload frequency when power is low), and introducing energy harvesting technology to supplement the power supply of edge nodes. (5) Investigate the integration of quantum computing technology with AICSPA to further improve the security and efficiency of model training and data transmission. Quantum key distribution (QKD) will be used to enhance the security of the secure collaborative communication channel, and quantum machine learning algorithms will be explored to accelerate the training speed of the cloud-side global model, breaking through the computational bottleneck of traditional AI algorithms. (6) Establish a standardized evaluation system for AI-driven cloud-edge security architectures. The system will include evaluation indicators such as detection rate of advanced attacks, convergence speed of collaborative models, resource utilization rate, energy consumption, and compliance with data privacy regulations, providing a unified benchmark for the evaluation and comparison of similar security architectures.

7. Conclusion

Aiming at the problems of low detection rate of unknown threats, high latency, and poor adaptability of traditional security defense mechanisms in cloud-edge computing ecosystems, this study proposes an AI-driven collaborative security protection architecture (AICSPA). The architecture integrates edge-side lightweight AI detection, cloud-side intelligent threat analysis, secure collaborative communication, and dynamic policy optimization, realizing the proactive and adaptive security protection of cloud-edge ecosystems. The key algorithms of AICSPA, including hierarchical federated learning algorithm and multi-objective dynamic policy optimization algorithm, solve the problems of data privacy leakage, resource constraints, and dynamic threat adaptation in cloud-edge collaborative security.

Experimental evaluations show that compared with traditional cloud-centric and edge-standalone security architectures, AICSPA has significant advantages in threat detection rate, detection latency, resource consumption, and bandwidth consumption. Especially for unknown attacks, AICSPA's detection rate reaches 96.3%, and the average detection latency is reduced to 12.5ms, which can meet the security and real-time requirements of most cloud-edge applications. The research results provide an effective technical solution for the security protection of cloud-edge computing ecosystems, and have important theoretical and practical significance for promoting the healthy development of cloud-edge integrated applications.

In the future, we will further optimize the architecture and algorithms of AICSPA, enhance its fault tolerance and adaptability, and promote its application in more real cloud-edge scenarios. We believe that AI-driven cloud-edge collaborative security will become an important development direction of cybersecurity in the era of distributed computing.

References

[1] Chen, Y., et al. (2024). Lightweight GBDT-based anomaly detection for edge controllers in industrial cloud-edge systems.*IEEE Transactions on Industrial Informatics*, 20(5), 5678–5688.

[2] Gonzalez, C. M., et al. (2023). Static vs. dynamic security policies for cloud-edge computing: A comparative analysis.*Journal of Network and Computer Applications*, 215, 103456.

[3] Grand View Research. (2025). Cloud-edge computing market size report, 2024-2028. San Francisco, CA: Grand View Research, Inc.

[4] Li, M., et al. (2023). Cloud-edge collaborative intrusion detection system based on suspicious traffic analysis. *Computers & Security*, 128, 103189.

[5] Liu, X., et al. (2023). Lightweight CNN-based intrusion detection for resource-constrained edge nodes.*IEEE Internet of Things Journal*, 10(8), 7234–7245.

[6] Laurent, S. A., et al. (2024). Threat landscape analysis of cloud-edge computing ecosystems: A case study of industrial ransomware attacks. *Computer Networks*, 231, 109456.

[7] Wang, L. J., et al. (2024). Applications of cloud-edge computing in smart cities: A survey. *IEEE Communications Surveys & Tutorials*, 26(2), 1234–1268.

[8] Wang, H., et al. (2023). Deep learning-based multi-source threat intelligence fusion for cloud security. *IEEE Transactions on Cloud Computing*, 11(3), 2456–2468.

[9] Zhang, Y., et al. (2025). AI-driven security for cloud-edge computing: A survey. *ACM Computing Surveys*, 58(7), 1–32.

[10] Zhang, J., et al. (2024). GAN-based attack simulation model for cloud-edge security training. *IEEE Transactions on Dependable and Secure Computing*, 21(4), 1890–1902.

[11] Zhao, Z., et al. (2024). Blockchain-based cloud-edge security collaboration platform for trustable data transmission. *Future Generation Computer Systems*, 145, 345–358.

[12] TensorFlow Federated. (2024). TFF 0.52.0 documentation. Retrieved from https://www.tensorflow.org/federated

[13] CSE-CIC-IDS2018 Dataset. (2023). Canadian Institute for Cybersecurity. Retrieved from https://www.unb.ca/cic/datasets/ids-2018.html

[14] KDD Cup 99 Dataset. (2023). UCI Machine Learning Repository. Retrieved from https://archive.ics.uci.edu/ml/datasets/kddcup99

[15] MITRE ATT&CK. (2024). MITRE ATT&CK Framework for Cloud and Edge Computing. Retrieved from https://attack.mitre.org/matrices/enterprise/cloud/

[16] International Telecommunication Union (ITU). (2024). Technical Report on Cloud-Edge Computing Security Evaluation Indicators. Geneva: ITU.

[17] IEEE Standards Association. (2025). IEEE 2418.5-2025 Standard for AI-Driven Security in Cloud-Edge Ecosystems. New York: IEEE.