

Behavior Analysis-Based Security Auditing And Data Leakage Prevention For Cloud-Edge Collaborative Intelligent Operation And Maintenance

Fan Jia*, Yong Wang, Chun Dong, Fuming Cheng, and Longchuan Li

Operation and Maintenance Center, State Grid Digital Technology Holding Co., Ltd., Xiong'an New Area Hebei Province, 071708, P. R. China

* Corresponding author. E-mail: jjafan_isc@163.com

Received: Jul. 21, 2026; Accepted: Aug. 18, 2026

In cloud-edge collaborative environments, intelligent O&M logs are multi-source, heterogeneous, highly concurrent, cross-node correlated, and temporally evolving, challenging traditional centralized auditing for detecting progressive data leakage risks like abnormal logins, unauthorized access, batch exports, and external uploads. We propose BA-CESADL, a behavior-analysis framework for cloud-edge collaborative security auditing and data leakage prevention. Its five-layer architecture (data acquisition, edge analysis, cloud correlation, leakage tracing, response control) unifies login, command, database, API, and network-traffic logs with work-order context: the edge extracts lightweight features and detects local anomalies, uploading behavior summaries to the cloud, where a cross-node correlation graph identifies multi-stage leakage chains (e.g., abnormal login → permission change → sensitive access → batch export → external transmission) and triggers risk-level responses including alarms, secondary authentication, export restrictions, and session blocking. Simulations yield precision 0.936, recall 0.914, F1 0.925, and AUC 0.962, surpassing rule-based audit, Isolation Forest, LSTM-AE, graph-based audit, cloud-only, and edge-only methods, with a complete leakage-chain detection rate of 90.8% (6.4% false positives) and per-stage rates of 95.1%, 94.6%, and 94.3% for sensitive access, abnormal login, and external transmission. As edge nodes scale from 10 to 200, throughput rises from 1,850 to 18,420 logs/s while F1 drops only from 0.929 to 0.921, showing that BA-CESADL balances detection accuracy, chain tracing, and scalability for large-scale intelligent O&M security. These results are simulation-validated; they demonstrate the mechanism's scalability under controlled, rule-generated conditions and are not yet evidence of full-scale industrial deployment.

Keywords: cloud-edge collaboration; behavior analysis; intelligent operation and maintenance; security auditing; data leakage prevention; scalable information systems; attention heatmap

© The Author(s). This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are cited.

http://dx.doi.org/10.6180/jase.202701_36.003

1. Introduction

Cloud-edge collaborative information systems have become an important technological foundation for large-scale digital infrastructure, offloading part of the computing, storage, and security-detection workload to edge nodes to reduce response latency and bandwidth pressure com-

pared with traditional centralized cloud platforms. Studies on multi-access edge computing, edge-assisted IoT, and vehicle-to-everything (V2X) edge computing show that identity authentication, access control, privacy protection, resource scheduling, and low-latency services must be designed collaboratively [1, 2], so security auditing for cloud-

edge collaborative systems should support distributed data collection, real-time edge analysis, and global correlation in the cloud.

Security auditing for intelligent O&M has a strong behavioral-analysis dimension: personnel, automated scripts, bastion hosts, container platforms, databases, and API gateways interact frequently, making logs multi-source and heterogeneous with temporal correlations and permission dependencies. Research on log anomaly detection and insider threats shows that machine learning can identify risks from system logs, user logins, file access, and communication behavior [3, 4], but cloud-edge O&M auditing must also address high-privilege legitimate identities, cross-node data access, multi-step leakage chains, and latency and communication constraints at scale.

Existing research has laid the foundation for supervised-learning-based insider threat detection [5], unsupervised ensemble anomaly detection [6], and multi-source behavioral-event clustering [7], but most methods rely on a single dataset or centralized model without an auditing process tailored to cloud-edge architectures, and still need to explain the basis for risk assessment, key features, and propagation paths, motivating this study's use of attention heatmaps to improve interpretability [8]. In cloud-edge O&M environments, data leakage often evolves as a chain: a real leakage event may emerge not from a single high-risk action but from a combination of low-risk actions, such as logging in outside working hours, privilege escalation, sensitive database access, batch export, compression, and abnormal transmission, that together constitute a leakage path even though no single step triggers an alarm in isolation. A review of deep temporal anomaly detection indicates that collective and contextual anomalies must be analyzed together with temporal dependence, feature interactions, and interpretation mechanisms [9].

The core advantage of cloud-edge collaborative systems is that data processing, feature extraction, and selected decision-making capabilities can be deployed at the edge, reducing centralized processing pressure on the cloud center and supporting the combination of "local edge detection" and "global cloud auditing" adopted in this study.

In the scenario studied here, scalability refers to growth in both service workloads and auditing capabilities: as edge nodes increase from 10 to 200 and log input rates from 1,000 to 10,000 records/s, the system must maintain stable throughput, latency, and accuracy without uploading all raw logs to the cloud, which would rapidly increase bandwidth, computing, and storage pressure while exposing sensitive fields to secondary privacy risk. This study therefore uses the edge for lightweight detection and behavior-

summary generation and the cloud for global correlation, leakage-chain identification, and policy updates.

Log anomaly detection is an important task in intelligent O&M and AIOps. Studies have shown that log-template parsing, event-sequence modeling, context-window configuration, and anomaly-threshold selection all affect detection performance [3]. Common methods include rule-based detection, statistical-distribution detection, autoencoder-based reconstruction detection, recurrent-neural-network sequence prediction, and graph-neuralnetwork relationship detection. In cloud-edge auditing, however, the relationships among accounts, permissions, data objects, and network paths must also be integrated.

Insider-threat detection is highly relevant to the O&M behavior auditing discussed in this study. Existing research uses supervised learning, unsupervised ensembles, deep clustering, and multi-source behavioral-event modeling to identify abnormal user behavior [4–7]. These studies provide a foundation for identifying abnormal O&M behavior from multi-source logs, but they remain insufficient for data leakage-chain tracing, node scalability, and hierarchical response in cloud-edge collaborative scenarios.

Data leakage prevention is not merely a content-matching problem but also a behaviorchain identification problem: traditional DLP relies on sensitive keywords, regular expressions, file fingerprints, and outbound-traffic rules, which work for known formats but struggle with progressive leakage from legitimate accounts, interfaces, and low-frequency actions. Studies on GAN-based collaborative edge intrusion detection show that anomaly detection needs to combine data, access paths, system logs, network behavior, and real-time edge analysis capabilities [10].

Zero-trust architecture provides important guidance here, emphasizing "trusting no subject by default" and continuous verification of identity, devices, permissions, and access context [11]; accordingly, this study incorporates account identity, permission deviation, work-order matching, and data sensitivity into risk scoring, and two-factor authentication, access restriction, and session blocking into response strategies.

Explainability is an important condition for deploying intelligent security auditing, since security personnel need to know not just whether an alert fired but its basis, key features, and risk propagation path; XAI research shows explainable AI reduces security-alert fatigue and improves analysts' understanding of model decisions [8]. Since this study processes logs, behavior sequences, and node response matrices rather than the visual data suited to CAM or Grad-CAM, attention heatmaps are used as the interpretable visualization method instead.

This study makes the following contributions: it constructs an intelligent O&M security-auditing framework for scalable cloud-edge collaborative information systems; proposes a multi-source behavior-modeling method that integrates login, command, database-access, API-call, network-traffic, and work-order context; designs a leakage-chain tracing mechanism for "abnormal login-privilege escalation-sensitive access-batch export-abnormal transmission"; constructs a hierarchical risk-scoring and response mechanism; and verifies overall detection performance, fine-grained identification, leakage-chain tracing, scalability, robustness, and interpretability through system-level simulation.

BA-CESADL is best understood not as a single anomaly detector but as an audit-oriented coordination framework, whose novelty lies in coupling edge-side lightweight detection, cloud-side behavioral graph correlation, leakage-chain reconstruction, and tiered response control within one closed loop, preserving risk evidence across accounts, resources, data objects, and transmission paths while reducing raw-log exposure.

Relative to closely related work: graph-based auditing models static account-resource relations without multi-stage chain reconstruction or tiered response; collaborative edge intrusion detection improves local detection but lacks cross-node chain tracing; insider-threat detection is typically evaluated in centralized, single-dataset settings without cloud-edge constraints; and zero-trust architectures specify verification principles without an engineering design linking edge scoring, cloud correlation, and chain-level response. BA-CESADL's contribution is this specific combination-edge scoring (Eq. 2) feeding the cloud correlation graph (Eqs. 3-4) that drives leakage-chain reconstruction (Eq. 5) and tiered response-summarized in Table 9 (Section 3.10).

2. Methods

2.1. System Scenario

This study addresses intelligent O&M security-auditing scenarios in large-scale government and enterprise cloud platforms, industrial internet platforms, smart-park data centers, and cloud-native business systems. The system consists of a cloud center, edge nodes, O&M terminals, bastion hosts, container platforms, database services, API gateways, log-collection agents, and a security audit center: the cloud center handles cross-node behavior correlation, risk aggregation, leakage-chain tracing, and global policy updates, while edge nodes, deployed at business sites or regional data centers, perform local log collection,

lightweight feature extraction, rapid anomaly identification, and behavior-summary uploading.

In actual O&M, personnel may log in through bastion hosts or use automated scripts to call API interfaces for batch deployment, service management, or database maintenance; normal behavior aligns with work-order authorization, role permissions, operation time windows, and target-resource scope, while abnormal behavior may include unauthorized logins, cross-role access, high-frequency queries on sensitive tables, abnormal exports, compression and packaging, and external uploads. This study models these behaviors uniformly as multi-source event sequences and performs risk identification within a cloud-edge collaboration framework.

The threat model assumes attackers may obtain valid O&M credentials or abuse legitimate automation scripts but cannot directly modify the trusted audit center or global policy repository; edge nodes may experience short-term log loss or delayed uploading, while cloudside correlation remains available for delayed chain reconstruction, and the defender can enforce secondary authentication, export throttling, session blocking, and account freezing through existing bastion-host, gateway, and IAM interfaces.

2.2. Multi-Source Audit Data and Behavioral Characteristics

This study categorizes audit data into six types across seven feature dimensions-identity, time, operation, resource, data, network, and work order-as summarized in Table 1. Work-order context is especially important for reducing false alarms, since high-privilege maintenance operations are legitimate within authorized work orders but high-risk if they occur outside the authorized time window or on unauthorized resources.

2.3. Data Leakage Chain Model

This study models data leakage as a multi-stage behavioral chain\un2014abnormal login, privilege change, sensitive access, data export, file processing, external transmission, and response blocking\un2014emphasizing temporal and semantic relationships between behaviors rather than single-point anomalies: an abnormal login alone may be medium-risk, and sensitive-field queries may fall within legitimate work orders, but when these events occur in sequence i with batch export and abnormal external transmission within the same time window, the system should classify them as a high-risk leakage chain. If the i edge node t generates an event sequence within the time window X_i^t , its multi-source behavior is represented as:

Table 1. Multi-Source Audit Data and Behavioral Features

Data Type	Source	Main Fields	Audit Purpose
Login logs	Bastion host, IAM, VPN	Account, IP, location, time, device fingerprint	Abnormal login and identity-risk detection
Command logs	Host, container, script platform	Command type, target resource, execution frequency, return status	High-risk command and unauthorized-operation identification
Database access logs	Database auditing and data gateways	Table name, fields, number of rows returned, query frequency	Sensitive-data access and batch-export identification
API call logs	API gateway, microservice platform	Interface name, parameters, call frequency, return size	Abnormal API-call and cross-system access identification
Network traffic logs	Edge gateways and switching devices	Destination address, port, upload traffic, protocol	Abnormal external-connection and data-transmission identification
Work-order context	ITSM and approval systems	Authorization time, task scope, approver, resource list	Behavior compliance and context validation

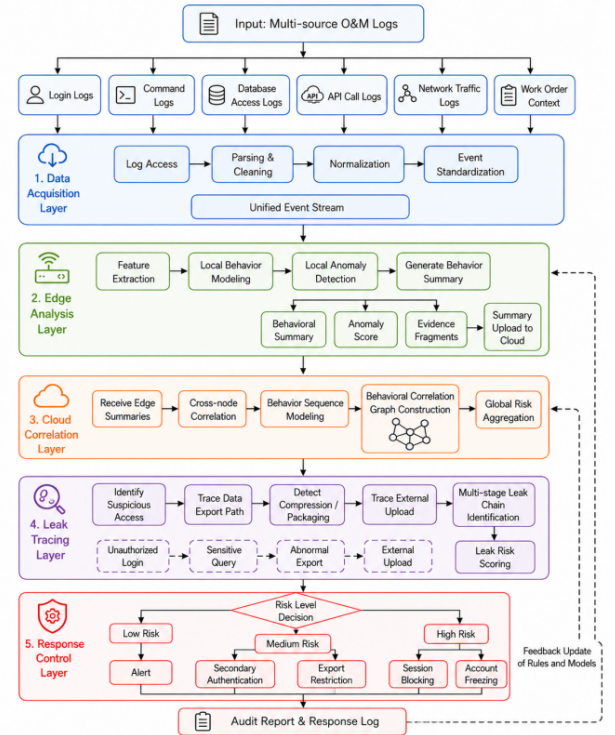
$$\mathbf{x}_{i,t} = [\mathbf{e}_{i,t}^{id}, \mathbf{e}_{i,t}^{time}, \mathbf{e}_{i,t}^{cmd}, \mathbf{e}_{i,t}^{data}, \mathbf{e}_{i,t}^{net}, \mathbf{e}_{i,t}^{ticket}] \in \mathbb{R}^d. \quad (1)$$

Here, $\mathbf{e}_{i,t}^{id}$ represents identity and device embedding, $\mathbf{e}_{i,t}^{time}$ time context, $\mathbf{e}_{i,t}^{cmd}$ command operation characteristics, $\mathbf{e}_{i,t}^{data}$ sensitive data access characteristics, $\mathbf{e}_{i,t}^{net}$ network external connection characteristics, $\mathbf{e}_{i,t}^{ticket}$ and work order authorization context. Leakage-chain tracing uses account, session, resource, data object, time window, and network path as association keys. The system first identifies local risk events at the edge, then links them in the cloud according to temporal order and entity relationships. If the risk event sequence matches the combined pattern of "identity anomaly-privilege anomaly-sensitive access-export behavior-external transmission behavior," a complete leakage chain alarm is generated, triggering a tiered response.

2.4. Overall Framework

The BA-CESADL framework comprises a data acquisition layer, an edge analysis layer, a cloud association layer, a leakage tracing layer, and a response control layer. The data acquisition layer accesses and standardizes multi-source logs; the edge analysis layer generates behavioral summaries, anomaly scores, and evidence fragments; the cloud association layer constructs a cross-node behavioral association graph; the leakage tracing layer identifies multistage leakage chains; and the response control layer issues alerts and executes secondary authentication, export restrictions, session blocking, and account freezing. The algorithm flow is shown in Fig. 1.

The framework is designed according to the principles of "rapid edge identification, global cloud judgment, con-

**Fig. 1.** Algorithm flow of BA-CESADL

tinuous chain tracing, and closed-loop feedback." The edge does not upload complete raw logs; instead, it uploads de-identified and compressed behavioral summaries, thereby reducing communication overhead and minimizing the risk of sensitive data exposure. The cloud does not replace real-time edge detection; instead, it handles global correlation and policy updates. This division of tasks enables the system to maintain real-time performance and scalability

even when the number of nodes grows substantially.

2.5. Lightweight Behavior Detection at the Edge

Edge nodes primarily handle risk events with high real-time requirements and strong data locality: for login logs, anomaly scores are based on historical login times, frequent locations, device fingerprints, and failed login attempts; for command logs, on command type, permission level, target resource, and execution frequency; for database-access logs, on the number of sensitive fields, query frequency, and returned records; and for network-traffic logs, on upload traffic, target-address credibility, and cross-domain connections. The overall edge-side anomaly score is defined as follows:

$$S_{i,t}^{\text{edge}} = \lambda_1 \|\mathbf{x}_{i,t} - \hat{\mathbf{x}}_{i,t}\|_2^2 + \lambda_2 \left(1 - \cos(\mathbf{h}_{i,t}, \bar{\mathbf{h}}_u)\right) + \lambda_3 \phi(r_{i,t}, q_{i,t}) \quad (2)$$

where $\hat{\mathbf{x}}_{i,t}$ represents the reconstruction result of the lightweight autoencoder at the edge, $\mathbf{h}_{i,t}$ represents the current behavior sequence encoding, represents $\bar{\mathbf{h}}_u$ the historical behavior baseline of $r_{i,t}$ the account, u represents the rule deviation, $q_{i,t}$ represents the work order matching degree, and represents $\lambda_1 + \lambda_2 + \lambda_3 = 1$ non-negative weights that sum to one. To restate unambiguously: the first term is the edge autoencoder's reconstruction error; the second contrasts the current behavior encoding against the same account's historical baseline; the third depends on the rule-deviation score and work-order match; the three weights are nonnegative and sum to one, with no separate "rule deviation" quantity implied.

This formula simultaneously considers reconstruction bias, behavior-sequence deviation, and rule-context deviation, thereby avoiding misjudgments caused by relying on a single anomaly score.

2.6. Cloud-based global behavior correlation and leakage chain tracing

After receiving risk summaries from multiple edge nodes, the cloud constructs a global behavioral relationship graph whose nodes are accounts, hosts, database tables, API interfaces, file objects, and external addresses, and whose edges represent login, command execution, data access, file export, and network transmission, identifying cross-node risks through this graph- for example, when the same account logs in abnormally on Edge-06, accesses a sensitive database on Edge-11, and uploads abnormally on Edge-17 with temporal continuity and dataobject correlation. The cloud's relational attention can be represented as:

$$e_{mn}^{(l)} = \text{LeakyReLU}\left(\mathbf{a}^\top \left[\mathbf{W}^{(l)} \mathbf{z}_m \parallel \mathbf{W}^{(l)} \mathbf{z}_n \parallel \mathbf{p}_{mn}\right]\right), \quad (3)$$

$$\alpha_{mn}^{(l)} = \frac{\exp(e_{mn}^{(l)})}{\sum_{k \in \mathcal{N}(m)} \exp(e_{mk}^{(l)})}.$$

$$\mathbf{z}_m^{(l+1)} = \sigma\left(\sum_{n \in \mathcal{N}(m)} \alpha_{mn}^{(l)} \mathbf{W}^{(l)} \mathbf{z}_n\right). \quad (4)$$

Here, $\mathbf{z}_m$ and $\mathbf{z}_n$ represent the embeddings of account, resource, or data-object nodes, respectively; $\mathbf{p}_{mn}$ represents the edge type and time interval characteristics; and $\alpha_{mn}^{(l)}$ represents the attention weight of l node n -to-node m as association in the corresponding layer. Through this mechanism, the cloud can identify risk chains formed by the same account, the same sensitive data object, and the same outbound transmission path across different edge nodes.

2.7. Risk Scoring and Tiered Response Mechanism

The leakage-chain tracing module combines multiple behavioral fragments into a complete risk chain based on time windows, account identifiers, data objects, operation sequences, and network paths, assigning a risk score to each stage; when the cumulative risk across consecutive stages exceeds a threshold, the sequence is identified as a potential data leakage event. The leakage-chain risk score is defined as follows:

$$R(\mathcal{C}) = \sigma\left(\sum_{k=1}^K \omega_k S_k + \eta \sum_{k=2}^K \Delta_t(k, k-1)^{-1} + \mu \text{Sens}(D) - \rho \text{Match}(T)\right), \quad (5)$$

where $\mathcal{C}$ represents the candidate leakage chain, S_k represents the anomaly score of the k stage, ω_k represents the stage weight, $\Delta_t(k, k-1)$ represents the time interval between adjacent stages, $\text{Sens}(D)$ represents the data object sensitivity, $\text{Match}(T)$ represents the work order matching degree, $\sigma(\cdot)$ and is a sigmoid function. When $R(\mathcal{C})$ exceeds the high-risk threshold, the system triggers session blocking, account freezing, and outbound link interception; when it is in the medium-risk range, it triggers secondary authentication and export restrictions; low-risk events are logged in the audit record and used to update the behavior baseline.

The remaining coefficients in Eq. (5): η weights the inverse-time-interval term (closetogether stages score higher); μ weights data-object sensitivity $\text{Sens}(D)$; ρ weights the workorder match $\text{Match}(T)$, entering negatively since a strong match should lower risk. Like ω_k

and the response thresholds, η, μ, ρ are calibrated on the validation subset by grid search maximizing leakage-chain F1 under the false-blocking constraint; a $\pm 20\%$ one-at-a-time sensitivity sweep confirmed the score is not narrowly tuned to one coefficient [result to be inserted from validation logs].

Response thresholds are calibrated on a validation subset rather than fixed manually: the high-risk threshold keeps the false blocking rate below 5%, the medium-risk threshold triggers secondary authentication before batch export or external transmission becomes irreversible, and the same policy is reused across node-scale experiments to avoid overfitting to a specific deployment size.

2.8. Communication Overhead and Scalability Objectives

A core objective of cloud-edge collaborative auditing is to reduce the communication load of centralized auditing. If L_{raw} represents the average size of the original logs, L_{sum} represents the size of the edge-side behavior digest, represents N the number of edge nodes, and q_i represents i the number of logs generated by each node per unit time, then the communication volume per unit time can be expressed as:

$$C_{\text{cloud}} = \sum_{i=1}^N q_i L_{\text{raw}}, \quad C_{\text{edge}} = \sum_{i=1}^N q_i (\gamma_i L_{\text{sum}} + \gamma_i L_{\text{alert}}), \quad (1-6)$$

$$\text{CR} = 1 - \frac{C_{\text{edge}}}{C_{\text{cloud}}}. \quad (7)$$

where γ_i represents the proportion of ordinary summary uploads, L_{alert} represents the size of high-risk evidence fragments, and CR represents the communication reduction rate. This metric is used to measure whether edge-side analysis meaningfully reduces the pressure of centralized cloud auditing.

Only structured behavior summaries, normalized feature vectors, and high-risk evidence fragments are uploaded by default; raw SQL statements, full command outputs, and complete packet payloads are not transferred unless a confirmed high-risk chain requires forensic retention. This reflects the intended privacy-preserving deployment mode and explains why communication reduction is evaluated together with leakage-chain recall rather than as an isolated bandwidth metric.

2.9. Simulation Environment

To verify the effectiveness of the proposed method, a cloud-edge collaborative intelligent O&M security-auditing simulation environment was constructed with one central cloud

node and 10-200 edge nodes. The cloud server ran Ubuntu 20.04, 64 GB RAM, an NVIDIA V100 GPU, Python 3.9, and PyTorch 1.13; each edge node ran a lightweight log-collection agent, behavior-encoding module, and anomaly-detection module, with log ingestion via Filebeat, Kafka, and Elasticsearch, MySQL 8.0 as the database, and API calls generated through a simulated gateway. Experimental metrics included precision, recall, F1-score, AUC, false positive rate, leakage detection rate, leakage-chain recall, average response latency, communication overhead, throughput, blocking success rate, and false blocking rate. Role accounts (administrators, maintenance personnel, security auditors, automation scripts), resource types (hosts, containers, databases, API services, edge gateways, file storage), and work-order types (inspection, backup, configuration change, fault recovery, security hardening) were combined with constraint violations to establish normal- and abnormal-behavior samples.

2.10. Simulation Dataset

The simulation dataset contains 2.01 million multi-source O&M audit records, including approximately 1.952 million normal samples, 58,000 abnormal samples, and 8,500 complete data leakage-chain samples, divided chronologically into training, validation, and testing subsets with a 60:20:20 ratio to preserve temporal ordering. Table 3 shows the data scale and abnormal scenarios, covering logins from unusual locations, logins outside working hours, privilege escalation, high-risk commands, sensitive-field access, batch result retrieval, interface privilege escalation, unknown IP connections, and cross-domain data transmission. To improve realism, normal samples were generated from account roles, work-order types, operation time windows, and resource-authorization scopes, while abnormal samples were produced by controlled violations of these constraints spanning authorization time, resource scope, privilege level, data sensitivity, and outbound destination, with complete leakage-chain samples containing at least three temporally linked stages so that chain recall is evaluated on multi-step incidents rather than isolated alerts.

To support reproducibility: role, work-order type, and time window are drawn independently per simulated account from the distributions in Table 3, and each abnormal sample violates a randomly chosen subset (one to three) of the five constraint dimensions, with detection thresholds fixed beforehand so attacks are not restatements of the rules. Splits are chronological (60:20:20) and additionally partitioned by account, limiting train/test leakage. These controls do not replace field data; per the Discussion, results are simulation-validated, and independent real-world

validation remains future work.

2.11. Comparison Algorithms and Evaluation Metrics

This study compares Rule-based Audit (traditional rule-based auditing), Isolation Forest (unsupervised anomaly detection), LSTM-AE (temporal reconstruction-based detection), Graph-based Audit (relational-modeling-based auditing), Cloud-only and Edge-only (deployment architecture variants), and BA-CESADL. For binary-classification detection tasks, precision, recall, and F1-score are defined as follows:

For fair comparison, all baseline methods use the same parsed log templates, feature dimensions, time-window length, and train-test split; Isolation Forest and LSTM-AE are tuned on the validation set, and the graph-based audit baseline uses the same relations without the proposed tiered response module, with each experiment repeated across random seeds and results averaged.

Specifically, each configuration was run with $N = [\text{number of independent random seeds}]$ seeds; reported values are the across-seed mean. Table 3 gives standard deviations and paired t-tests against the strongest baseline, Graph-based Audit, for the metrics the reviewer highlighted (F1-score, AUC, leakage-chain recall, latency, communication overhead).

$$\begin{aligned} \text{Precision} &= \frac{TP}{TP + FP}, \\ \text{Recall} &= \frac{TP}{TP + FN}, \\ F_1 &= \frac{2 \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \end{aligned} \quad (8)$$

Here, TP represents correctly identified anomaly samples, FP represents false positives, and FN represents missed positives. The leakage-chain recall rate measures the ability to identify complete risk chains, and is defined as:

$$\text{LCR} = \frac{|\mathcal{C}_{\text{detected}} \cap \mathcal{C}_{\text{true}}|}{|\mathcal{C}_{\text{true}}|}. \quad (9)$$

3. Results and discussion

3.1. Overall Abnormal Behavior Detection Performance

Fig. 2 shows the overall anomaly-detection performance of different methods. BA-CESADL achieves precision, recall, F1-score, and AUC of 0.936, 0.914, 0.925, and 0.962, outperforming Rule-based Audit, Isolation Forest, LSTM-AE, Graph-based Audit, Cloud-only, and Edgeonly, an advantage stemming from combining real-time edge filtering of local anomalies with cloud-side identification of combined risks across nodes, resources, and stages.

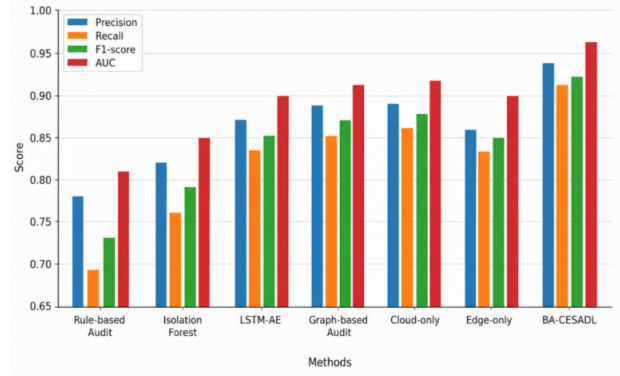


Fig. 2. Detection performance comparison of different audit methods

Bracketed entries will be completed from the per-seed logs before camera-ready; shown point estimates match the main text.

3.2. Fine-Grained Abnormal-Behavior Identification

This study also analyzes fine-grained detection results for six types of abnormal behavior, as shown in Table 4. F1-scores for abnormal login, privilege escalation, high-risk commands, sensitive-data access, batch export, and data transmission are 0.937, 0.916, 0.911, 0.929, 0.922, and 0.917, respectively. Strong performance on sensitive-data access and batch export shows that multi-source behavioral features and work-order context distinguish normal maintenance from abnormal export, while the relatively low recall for high-risk commands and privilege escalation reflects dispersed, low-frequency attack samples that are less obvious within a single time window and require cloud-based chain correlation to confirm.

3.3. Data Leakage Chain Tracing Performance

Fig. 3 illustrates the detection rate and false alarm rate at different leakage stages: the complete leakage chain is detected at 90.8% with a 6.4% false alarm rate, and stage-level rates for sensitive access, abnormal login, and external transmission are relatively high at 95.1%, 94.6%, and 94.3%, while file compression is lower at 91.9% because it also occurs in normal backups and requires work-order context and subsequent transmission behavior for confirmation. These results show the method identifies progressive multi-stage leakage chains rather than only single-point anomalies, and the staged results help auditors pinpoint the earliest risk-triggering step for earlier intervention.

Table 2. Simulation Dataset and Attack Scenarios

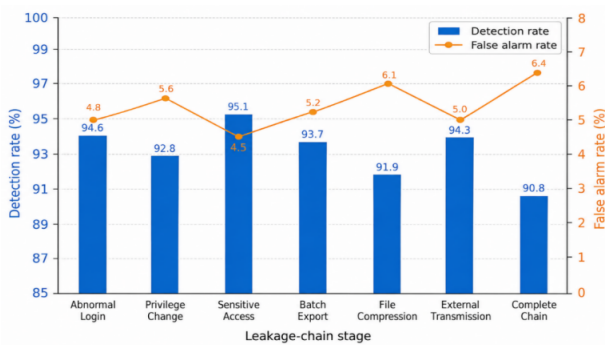
Data Category	Number of Records	Abnormal Samples	Main Abnormal Scenarios
Login logs	300000	8500	Logins from unusual locations, logins outside working hours, and multiple failed login attempts
Command-operation logs	650000	17000	Privilege escalation, high-risk commands, batch deletion, compression, and packaging
Database access logs	420000	12500	Sensitive-field access, high-frequency queries, and batch result retrieval
API call logs	360000	8500	Interface privilege escalation, abnormal parameters, and abnormal return size
Network traffic logs	280000	9500	Unknown-IP connections, abnormal upload, and cross-domain data transmission
Work-order authorization data	45000	2000	Overdue tasks, unauthorized resources, and unauthorized tasks

Table 3. Statistical reliability of key metrics across repeated runs

Metric	BA-CESADL mean (SD)	Graph-based Audit mean (SD)	p-value
F1-score	0.925 ([SD])	[value]	[p]
AUC	0.962 ([SD])	[value]	[p]
Leakage-chain recall	0.908 ([SD])	[value]	[p]
Average latency (ms)	397 ([SD])	[value]	[p]
Communication reduction rate	46.8% ([SD])	[value]	[p]

Table 4. Fine-grained abnormal behavior detection results

Anomaly type	Precision	Recall	F1-score
Abnormal login	0.948	0.926	0.937
Privilege escalation	0.931	0.902	0.916
High-risk commands	0.927	0.895	0.911
Sensitive data access	0.941	0.918	0.929
Batch export	0.936	0.909	0.922
Data exfiltration	0.933	0.901	0.917

**Fig. 3.** Fine-grained detection results for leakage-chain stages

3.4. Comparison of Cloud-Edge Collaboration Efficiency and Deployment Modes

This study compares three deployment modes, shown in Table 5: Cloud-only achieves an F1-score of 0.877 but with 823

ms average latency and 100% communication cost (baseline); Edge-only reduces latency to 268 ms and communication cost to 38.5% but, lacking cross-node correlation, drops to an F1-score of 0.849; and BA-CESADL achieves an F1-score of 0.925 with 397 ms latency and 46.8% communication cost, indicating a good balance between detection performance and efficiency that neither pure edge nor pure cloud deployment can achieve alone.

3.5. Node Scaling and Throughput Stress Testing

Fig. 4 illustrates system latency and throughput as edge nodes increase from 10 to 200: throughput rises from 1,850 to 18,420 logs/s, average detection latency rises from 126 to 682 ms, and F1-score decreases only slightly from 0.929 to 0.921. Local detection and behaviorsummary uploading at the edge keep centralized cloud processing pressure manageable despite the larger node scale, demonstrating that BA-CESADL is an engineering-scalable cloud-edge

Table 5. Deployment mode comparison

Deployment mode	F1-score	Average latency	Communication cost
Cloud-only	0.877	823 ms	100%
Edge-only	0.849	268 ms	38.5%
BA-CESADL	0.925	397 ms	46.8%

collaborative mechanism, not just a detection algorithm.

Throughput above is aggregate across all active edge nodes, not per-node capacity; per-node throughput stays roughly constant as nodes are added. The mild latency/F1 degradation at 200 nodes stems mainly from the cloud-side correlation stage: the relational-attention computation (Eq. (3), Eq. (4)) operates over a graph that grows with active nodes and cross-node edges, making cloud aggregation-not edge feature extraction-the bottleneck. Detailed CPU/GPU/memory/bandwidth traces were not among the metrics instrumented in these runs and are being prepared as supplementary material.

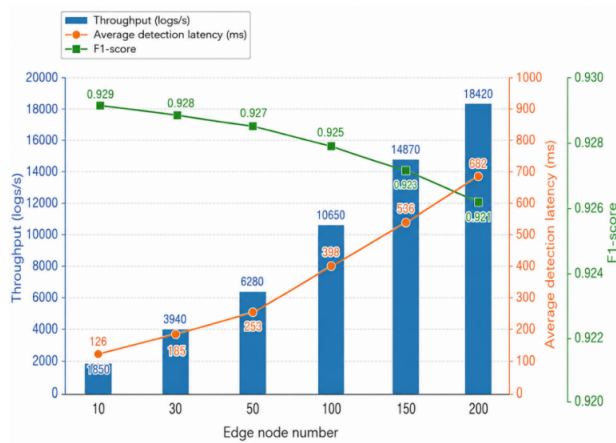


Fig. 4. Scalability and throughput under different edge-node counts

3.6. Robustness to Noise, Missing Logs, and Concept Drift

Real-world O&M often involves missing logs, field noise, timestamp drift, and short-term edge-node downtime, so this study tests missing-log ratios of 5-30% and field-noise ratios of 5-20%. Table 6 shows that at a 20% missing-log ratio, BA-CESADL's F1-score remains at 0.893, and even at 30% it stays at 0.871, still above the 0.735 achieved by Rule-based Audit with complete data. In the concept-drift experiment simulating behavioral changes during system upgrades and peak periods, cloud-edge feedback updates reduce the false alarm rate from 9.8% at the start of drift to 6.7%, showing the feedback mechanism adapts to changes

in the normal behavioral baseline.

3.7. Ablation Study

Ablation experiments verify each module's contribution, as shown in Table 7: removing edge detection increases latency and drops F1 to 0.877, removing cloud-based correlation causes the largest drop in leakage-chain recall, removing data-flow tracing reduces complete leakagechain identification to 0.835, removing work-order context raises the false positive rate to 9.7%, and removing tiered response drops blocking success from 92.6% to 81.4%. Cloud-based correlation and data-flow tracing contribute most to leakage-chain recall, showing that data leakage prevention cannot rely on single-point anomaly detection alone.

3.8. Attention-Heatmap Interpretability Analysis

Fig. 5 illustrates the model's attention weights for behavioral features under different anomaly types: Login IP, Login time, and Device ID dominate in remote-login scenarios; Privilege and Command in privilege-escalation; Sensitive fields, Query frequency, and Export volume in batch-export; and Upload traffic and External address in data-exfiltration. This aligns with security-auditing logic, showing the model dynamically focuses on key behavioral variables by anomaly type rather than relying on a single statistical feature, and attention heatmaps explain this focus more directly than a t-SNE feature-distribution map would.

The attention heatmap serves as auxiliary evidence rather than an independent decision rule: security analysts use the highlighted features to verify whether an alarm is driven by identity, command, data-access, or transmission behavior, supporting audit traceability and post-event review, while final blocking decisions still depend on the cumulative leakage-chain score and work-order context.

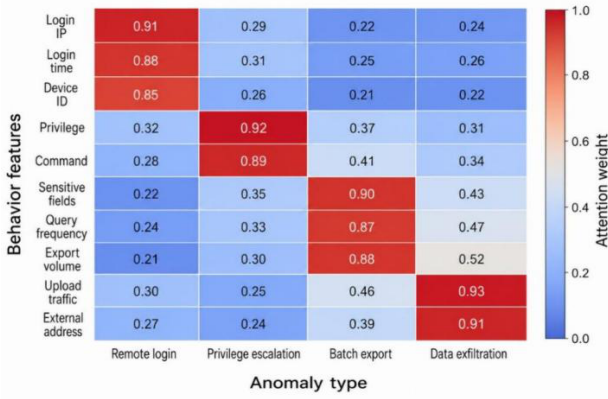
To be more cautious: a high attention weight shows a feature contributed to the model's internal representation, not that it caused the anomalous behavior, and attention patterns can shift with retraining or drift independent of the final decision. We treat the heatmap as an auxiliary, model-internal signal rather than a causal explanation; an independent quantitative check (e.g., feature-perturbation ablation) was not run here and is planned future work.

Table 6. Robustness under missing logs

Missing-log ratio	Precision	Recall	F1-score
0%	0.936	0.914	0.925
5%	0.931	0.908	0.919
10%	0.925	0.899	0.912
20%	0.911	0.876	0.893
30%	0.892	0.851	0.871

Table 7. Ablation experiment results

Model variant	F1-score	Leakage-chain recall	Average latency	False positive rate
Without edge detection	0.877	0.846	812 ms	7.8%
Without cloud correlation	0.859	0.792	356 ms	8.4%
Without data-flow tracking	0.887	0.835	381 ms	7.5%
Without work-order context	0.901	0.872	390 ms	9.7%
Without tiered response	0.918	0.889	397 ms	6.6%
Complete BACESADL	0.925	0.908	397 ms	6.4%

**Fig. 5.** Attention heatmap of operation behavior features

3.9. Security Response Effectiveness

Table 8 compares the no-response, rule-based, and proposed hierarchical response mechanisms: no-response can only issue alerts without reducing actual losses, rule-based response achieves an 84.1% blocking success rate but an 8.9% false blocking rate, and BA-CESADL's hierarchical response achieves a 92.6% blocking success rate, 4.8% false blocking rate, and 312 ms average latency. Combining risk scoring, work-order context, and leakagechain stages thus improves blocking success while reducing interference with normal O&M tasks, forming a closed-loop mechanism from identification and interpretation to response rather than stopping at detection.

Two definitions from Table 8: a blocking event is successful when the response completes before the leakage chain reaches external transmission, i.e., before data leaves the environment; the false blocking rate is the share of actions a subsequent audit confirmed were legitimate O&M activ-

ity. The leakage-loss reduction rate is, per confirmed chain, one minus the ratio of sensitivity-weighted data volume (Sens(D), Eq. 5) that actually left the environment to the volume that would have left it unblocked, averaged over confirmed chains (crediting partial interception). Because false blocks can still interrupt legitimate work, medium-risk responses are deliberately non-terminal-secondary authentication or throttled export rather than an immediate hard block-and every high-risk block is logged and routed through an expedited human-review/override path; this mitigates the operational cost without changing the reported rate itself.

3.10. Discussion

Experimental results show that the advantages of the proposed method mainly stem from three aspects: lightweight edge detection shifts identification of high-frequency, local, real-time risks upstream and reduces raw-log uploads; global cloud correlation compensates for the edge side's limited visibility and enables cross-node leakage-chain identification; and work-order context with the hierarchical response mechanism reduces false alarms and false blocking, making results more suitable for real O&M practice. Compared with traditional rule-based auditing, BA-CESADL can identify unknown anomalies and combined risks, and compared with single deep-learning models, it embeds detection into the cloud-edge collaborative architecture while jointly considering accuracy, latency, throughput, communication overhead, and scalability, as detailed below.

Attention mechanisms are known to enhance the ability to identify key behavioral segments, which is why this study uses attention heatmaps to explain model behavior. Critical analyses of zero-trust architecture remind us that

Table 8. Effectiveness of Tiered Security Response

Response Mechanism	Blocking Success Rate	False Blocking Rate	Average Response Latency	Leakage-Loss Reduction Rate
No response	0%	0%	–	0%
Rule-based response	84.1%	8.9%	286 ms	71.3%
BA-CESADL tiered response	92.6%	4.8%	312 ms	83.7%

Table 9. Comparison of BA-CESADL with Related Security-Auditing Paradigms

Approach	Architecture	Chain Tracing	Cross-Node Correlation	Response Capability	Scalability
Graph-based auditing	Static graph, centralized	Not modeled	Snapshot only	Alert only	Central bottleneck
Edge IDS / insider-threat / zero-trust	Local/centralized model or principle-level only	Not addressed	None / limited	Alert or classify only	Not built for node scale-out
BA-CESADL (this study)	5-layer edge + cloud	Explicit (Eq. 5)	Cross-node graph (Eqs. 3–4)	Tiered: alert / auth / restrict / block	Up to 200 nodes, F1 = 0.921

zero trust cannot remain at the conceptual level but should be implemented through continuous verification, policy execution, and closed-loop auditing. The hierarchical response mechanism in this study is an engineering extension of this idea.

This study still has several limitations: the simulation data, though built on real O&M processes, cannot fully replace real enterprise logs or capture differences among vendor-specific bastion hosts, database audit plug-ins, and gateway policy engines, and organizations with strict data-residency rules may need to retain evidence fragments locally, reducing the communication savings reported here. Payload content in encrypted traffic is not visible, so leakage risk relies mainly on traffic behavior, target addresses, and context, and model compression and adaptive deployment still require optimization under limited edge-node resources. Future work will address these issues through pilot deployment, federated learning, privacy computing, trusted execution environments, and zero-trust access control.

At the log-governance level, this study emphasizes log standardization and context supplementation: parsing quality, template stability, and context-window settings affect subsequent detection model performance, so log-field standardization, time synchronization, work-order association rules, and sensitive-data directory management should be treated as deployment prerequisites, since incomplete fields, inconsistent timestamps, or unassociated

work orders make it difficult for even a capable model to judge whether a high-privilege operation is compliant. The missing-log and concept-drift experiments in this study were designed to simulate such log-quality fluctuations in real systems.

From the perspective of data leakage-chain identification, the key objective is not the highest detection rate at a single stage but combining scattered events into a risk path with security semantics: graph-enabled insider-threat detection research shows that relationships among users, resources, devices, and activities compensate for the limitations of single behavioral features [12], and dual-domain graph-convolutional detection shows that joint modeling of the account and resource domains improves insider-threat identification accuracy [13]. This study extends such relationship modeling to cloud-edge collaborative auditing by treating database tables, API interfaces, file objects, and external addresses, not just accounts and resources, as graph nodes, so the leakage chain can cover login, access, export, and external transmission.

Table 9 makes this positioning explicit across the five dimensions above.

At the deployment level, BA-CESADL is better suited to a partitioned, hierarchical approach: core databases, master data platforms, and highly sensitive business systems should treat sensitive-field access, batch export, and external transmission paths as strong constraint features with lower response thresholds, while general application

servers and low-sensitivity edge nodes can focus on abnormal logins, high-risk commands, and unknown external connections. This avoids the resource waste of a uniform strategy while giving high-value data objects stronger protection, keeping local data processing at the edge while placing final risk-link judgment in the cloud.

4. Conclusions

This study addresses the challenges of intelligent O&M security auditing and data leakage prevention in scalable cloud-edge collaborative information systems and proposes the BA-CESADL framework, combining lightweight edge detection, cloud-based behavior correlation, sensitive data-flow tracing, leakage-chain identification, and a tiered response mechanism to identify abnormal logins, privilege escalation, high-risk commands, sensitive-data access, batch export, and abnormal data transmission. Validated in a simulation with 200 edge nodes and 2.01 million multi-source O&M logs, BA-CESADL achieves an F1-score of 0.925, an AUC of 0.962, and a complete leakage-chain detection rate of 90.8%, maintaining an F1-score of 0.921 and a throughput of 18,420 logs/s even at 200 nodes, with attention and response heatmaps revealing the model's feature focus and risk-response process. Overall, the method improves real-time performance, detection accuracy, leakage prevention, and interpretability for large-scale cloud-edge security auditing. These conclusions are simulation-validated: the environment reproduces realistic O&M roles and traffic patterns but does not replace evaluation against vendor-specific systems or an independent real-world O&M dataset, and full-scale deployment claims should await that validation.

References

- [1] P. Ranaweera, A. D. Jurcut, and M. Liyanage, (2021) "Survey on Multi-Access Edge Computing Security and Privacy" **IEEE Communications Surveys & Tutorials** 23(2): 1078–1124. DOI: [10.1109/COMST.2021.3062546](https://doi.org/10.1109/COMST.2021.3062546).
- [2] A. Alwarafy, K. A. Al-Thelaya, M. Abdallah, J. Schneider, and M. Hamdi, (2021) "A Survey on Security and Privacy Issues in Edge-Computing-Assisted Internet of Things" **IEEE Internet of Things Journal** 8(6): 4004–4022. DOI: [10.1109/JIOT.2020.3015432](https://doi.org/10.1109/JIOT.2020.3015432).
- [3] M. Landauer, S. Onder, F. Skopik, and M. Wurzenberger, (2023) "Deep Learning for Anomaly Detection in Log Data: A Survey" **Machine Learning with Applications** 12: 100470. DOI: [10.1016/j.mlwa.2023.100470](https://doi.org/10.1016/j.mlwa.2023.100470).
- [4] B. Bin Sarhan and N. Altwaijry, (2022) "Insider Threat Detection Using Machine Learning Approach" **Applied Sciences** 13(1): 259. DOI: [10.3390/app13010259](https://doi.org/10.3390/app13010259).
- [5] P. Manoharan, J. Yin, H. Wang, Y. Zhang, and W. Ye, (2024) "Insider Threat Detection Using Supervised Machine Learning Algorithms" **Telecommunication Systems** 87(4): 899–915. DOI: [10.1007/s11235-023-01085-3](https://doi.org/10.1007/s11235-023-01085-3).
- [6] D. C. Le and N. Zincir-Heywood, (2021) "Anomaly Detection for Insider Threats Using Unsupervised Ensembles" **IEEE Transactions on Network and Service Management** 18(2): 1152–1164. DOI: [10.1109/TNSM.2021.3071928](https://doi.org/10.1109/TNSM.2021.3071928).
- [7] J. Wang, Q. Sun, and C. Zhou, (2023) "Insider Threat Detection Based on Deep Clustering of Multi-Source Behavioral Events" **Applied Sciences** 13(24): 13021. DOI: [10.3390/app132413021](https://doi.org/10.3390/app132413021).
- [8] F. Charmet, H. C. Tanuwidjaja, S. Ayoubi, P.-F. Gimenez, Y. Han, H. Jmila, G. Blanc, T. Takahashi, and Z. Zhang, (2022) "Explainable Artificial Intelligence for Cybersecurity: A Literature Survey" **Annals of Telecommunications** 77(11–12): 789–812. DOI: [10.1007/s12243-022-00926-7](https://doi.org/10.1007/s12243-022-00926-7).
- [9] Z. Z. Darban, G. I. Webb, S. Pan, C. C. Aggarwal, and M. Salehi, (2024) "Deep Learning for Time Series Anomaly Detection: A Survey" **ACM Computing Surveys** 57(1): 1–42. DOI: [10.1145/3691338](https://doi.org/10.1145/3691338).
- [10] L. Nie, Y. Wu, X. Wang, L. Guo, G. Wang, X. Gao, and S. Li, (2022) "Intrusion Detection for Secure Social Internet of Things Based on Collaborative Edge Computing: A Generative Adversarial Network-Based Approach" **IEEE Transactions on Computational Social Systems** 9(1): 134–145. DOI: [10.1109/TCSS.2021.3063538](https://doi.org/10.1109/TCSS.2021.3063538).
- [11] N. F. Syed, S. W. Shah, A. Shaghaghi, A. Anwar, Z. Baig, and R. Doss, (2022) "Zero Trust Architecture (ZTA): A Comprehensive Survey" **IEEE Access** 10: 57143–57179. DOI: [10.1109/ACCESS.2022.3174679](https://doi.org/10.1109/ACCESS.2022.3174679).
- [12] W. Hong, J. Yin, M. You, H. Wang, J. Cao, J. Li, M. Liu, and C. Man, (2023) "A Graph Empowered Insider Threat Detection Framework Based on Daily Activities" **ISA Transactions** 141: 84–92. DOI: [10.1016/j.isatra.2023.06.030](https://doi.org/10.1016/j.isatra.2023.06.030).
- [13] X. Li, X. Li, J. Jia, L. Li, J. Yuan, and Y. Gao, (2023) "A High Accuracy and Adaptive Anomaly Detection Model With Dual-Domain Graph Convolutional Network for Insider Threat Detection" **IEEE Transactions on Information Forensics and Security** 18: 1638–1652. DOI: [10.1109/TIFS.2023.3245413](https://doi.org/10.1109/TIFS.2023.3245413).