VOL. 21, NO. 8, APRIL 2026 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

RISK-ADAPTIVE CP-ABE FOR CLOUD-IOT USING DYNAMIC POLICIES AND REAL-TIME SECURITY ENFORCEMENT

Tanguturi Tejasree[1] and Shaik Ghouhar Taj[2]
[1]Department of Computer Science Engineering, JNTUA College of Engineering, Pulivendula, Andhra Pradesh, India
[2]Department of CSE, JNTUA College of Engineering, Pulivendula, India
E-Mail: tejashreee15@gmail.com

ABSTRACT

Cloud-IoT deployments increasingly require fine-grained access control that can respond to rapidly changing operational contexts, including suspicious access behavior, abnormal time or location, and device trust. This paper proposes a risk-adaptive Ciphertext-Policy Attribute-Based Encryption (CP-ABE) framework in an edge-cloud architecture, where an edge gateway computes a risk score and automatically strengthens or relaxes the encryption policy in real time. In low-risk states, data are encrypted under a baseline policy based on role and department; high-risk contexts trigger stricter, short-lived constraints, including location-zone verification, multi-factor authentication (MFA), and a 15-minute access window. The prototype was evaluated using a topology of 10 IoT nodes, an edge gateway, and cloud storage under brute-force, stolen-credential, and replay attacks. The results show that all 15 attacks were mitigated, corresponding to 100% adaptive defense accuracy, while maintaining practical overhead: 12.1 ms average encryption latency, 21.6 ms average decryption latency, and 59.0625 KB of cloud storage usage.

Keywords: Cloud-IoT security, attribute-based encryption (ABE), ciphertext-policy ABE (CP-ABE), dynamic access control, risk-adaptive security, context-aware policy enforcement, edge computing, real-time policy adaptation.

Manuscript Received 3 March 2026; Revised 8 April 2026; Published 20 June 2026

1. INTRODUCTION

Cloud-IoT systems increasingly support safety-critical and privacy-sensitive applications such as smart healthcare, smart homes, and smart grids, where IoT endpoints continuously generate data that are typically stored and processed on cloud platforms. Although cloud offloading improves scalability, it also expands the attack surface: adversaries can exploit stolen credentials, compromised devices, unusual locations, and abnormal access patterns to obtain sensitive records. Modern deployments therefore require dynamic security policies that adapt to changing environments and threats rather than relying on static, one-time authorization decisions [8], [15]-[17]. Blockchain-based ABE schemes further demonstrate the demand for distributed and auditable access control in industrial IoT environments [18].

Ciphertext-Policy Attribute-Based Encryption (CP-ABE) is widely used for fine-grained data sharing in untrusted storage. A data owner encrypts information under an access policy containing attributes such as role, department, and clearance, and only users whose keys satisfy that policy can decrypt. However, Cloud-IoT deployment presents practical challenges because IoT devices are resource-constrained, policies evolve, users and devices must be revoked, and cloud or edge assistance may be required to maintain acceptable latency. Fog-assisted revocation, outsourced encryption, and precomputation have therefore been proposed to reduce cryptographic overhead on constrained devices [5], [12], [13].

Despite these improvements, most CP-ABE systems still assume that the access policy is selected statically or updated administratively, which is poorly aligned with Cloud-IoT threats whose risk can change from minute to minute. Recent research addresses important but partial aspects of this problem, including policy hiding and cloud auditing [10], dynamic attribute and hidden-policy updates [9], outsourced encryption for constrained devices [12], [13], revocation in fog-enabled IoT [5], [6], [20], and hidden updatable access policies [19]. The dominant focus remains efficiency, privacy, or lifecycle management rather than automatic real-time strengthening or relaxation of ciphertext policies according to contextual risk.

Meanwhile, Zero Trust (ZT) access control emphasizes continuous verification using device posture, location, time, and behavioral indicators [16], [17]. Risk-aware access-control and adaptive edge-security models use trust or risk scores to modify authorization decisions dynamically [8], [15]. A combination of ZT access control and ABE has also been investigated for secure Cloud-IoT environments [14]. However, these models often enforce decisions at the gateway or service layer and may not provide cryptographic persistence at the data layer. If ciphertext is not bound to the current risk context, copied data or delayed access can bypass the intended policy. This creates a clear research gap: continuous risk-aware authorization must be integrated with CP-ABE so that the encryption policy itself evolves with risk and remains enforceable in an untrusted cloud.

To address this gap, this paper proposes a Risk-Adaptive CP-ABE framework for Cloud–IoT in an edge-cloud architecture. The core idea is to replace fixed encryption policies with dynamic CP-ABE policies

generated automatically from a lightweight risk engine that computes a real-time risk score using contextual signals such as location, time, device trust, abnormal traffic, and repeated failed access attempts. Under low risk, the system applies a baseline CP-ABE policy (e.g., Role=Doctor AND Dept=Cardiology). Under high risk, it automatically strengthens the policy by adding constraints such as MFA verification and short-lived time windows (e.g., Time Window=15min), thereby reducing the blast radius of compromised credentials and suspicious sessions.

Accordingly, this study pursues three objectives: (1) to build a secure CP-ABE access-control layer for Cloud-IoT data, (2) to develop an intelligent risk and context engine that dynamically updates access policies, and (3) to evaluate the resulting security-performance trade-offs against static ABE baselines using latency, policy-update overhead, storage overhead, and attack-resilience metrics.

The main contributions of this work are: (i) a risk-adaptive policy-generation mechanism that maps risk levels to CP-ABE policy strength in real time; (ii) an edge-assisted enforcement workflow that performs lightweight context extraction and risk scoring close to IoT devices while retaining ciphertext in cloud storage; and (iii) a structured evaluation comparing static CP-ABE with risk-adaptive CP-ABE under credential misuse, suspicious login, compromised-device, and replay scenarios. The framework extends prior work that primarily addresses efficiency, policy hiding, revocation, or authorization-layer risk assessment in isolation [5], [6], [8]-[10], [12]-[15], [19], [20].

2. LITERATURE REVIEW

A. CP-ABE for Cloud-IoT: Efficiency, Outsourcing, and Edge/Fog Assistance

CP-ABE enables fine-grained access control over cloud-hosted IoT data, but conventional implementations can impose substantial computation on constrained devices. Recent studies therefore emphasize fog or edge assistance, decentralized authority management, precomputation, and outsourced encryption or decryption [3], [5], [12], [13]. These techniques reduce device-side cost and improve scalability; however, their primary objective is cryptographic efficiency rather than automatic adaptation of ciphertext policies to real-time contextual risk.

B. Policy Privacy, Auditing, and Stronger Threat Models

Policy leakage can reveal sensitive information embedded in an access structure. Policy-hiding, cloud-auditing, and hidden updatable-policy mechanisms reduce this disclosure and support integrity verification [9], [10], [19]. Revocation is another major operational requirement, and fog-IoT studies examine attribute revocation, key updates, and their efficiency-usability trade-offs [5], [6], [20]. These approaches strengthen privacy and lifecycle management but generally do not regenerate ciphertext policies automatically from live risk signals.

C. Dynamic Policies in ABE versus Truly Risk-Adaptive Policies

Several ABE frameworks support dynamic attribute or policy updates [7], [9], [19]. In most of this literature, however, dynamic refers to changes initiated by an administrator or data owner. It does not denote automatic tightening or relaxation of the encryption policy in response to a continuously computed risk score.

D. Zero Trust and Risk/Context-Aware Access Control

Zero Trust access-control research increasingly emphasizes continuous verification, contextual signals, and risk scoring [16], [17]. Under this model, every request is evaluated using current evidence rather than being trusted solely because it originates inside a network boundary.

Risk-aware zero-trust and adaptive edge-security frameworks use trust or contextual scores to modify authorization decisions [8], [15]. Huang et al. combine zero-trust access control with ABE for Cloud-IoT environments [14]. Nevertheless, explicit real-time regeneration of the ciphertext policy from an evolving risk score remains limited.

E. Research Gap Motivating the Proposed Work

Across these strands, a consistent gap emerges:

- ZT/risk engines typically enforce decisions at the *access layer* (gateway/service) and do not guarantee cryptographic persistence of risk decisions once data is copied/outsourced.
- CP-ABE systems largely optimize efficiency, revocation, privacy, or traceability, but rarely make the *ciphertext policy itself* automatically risk-adaptive in real time.

These observations motivate the proposed Risk-Adaptive CP-ABE framework, which connects real-time risk assessment directly to ciphertext-policy generation and enforcement.

Table-1. Comparison of related literature.

Ref	Main focus	Edge/Fog assist/outsourcing	Dynamic policy/attribute updates	Policy hiding/privacy	Revocation / Traceability	Explicit risk / ZT scoring	Main limitation relative to proposed work
Xu et al. [9], 2025	Dynamic attribute updates and policy hiding for IoT	Not primary	Yes	Yes	Not primary	No	Updates are administrative, not risk-driven
Gasmi et al. [12], 2024	Load-balanced outsourced encryption for constrained IoT	Yes	No	Not primary	No	No	Efficiency focus; no adaptive policy generation
Wang et al. [10], 2023	Policy-hiding CP-ABE with cloud auditing	Cloud auditing	No	Yes	No	No	Static policy; no context-based adaptation
Mahdavi et al. [13], 2024	Precomputed and outsourced ABE encryption	Yes	No	Not primary	No	No	Reduces computation but does not adapt policy to risk
Sarma and Barbhuiya [5], 2022	Fog-IoT ABE with revocation and attribute merging	Yes	Partial	Not primary	Yes	No	Lifecycle management without real-time risk scoring
Peñuelas-Angulo et al. [6], 2023	Survey of ABE revocation for fog-IoT	Yes	Partial	Partial	Yes	No	Survey only; no risk-adaptive implementation
Routray and Bera [15], 2024	Risk-aware zero-trust access control	N/A (access layer)	Yes	N/A	N/A	Yes	Risk is not cryptographically bound to ciphertext
Halgamuge and Niyato [8], 2025	Adaptive edge-security policies	Edge-centric	Yes	N/A	N/A	Yes	Not CP-ABE; lacks ciphertext-level enforcement
Huang et al. [14], 2023	Zero-trust access control combined with ABE	Cloud-assisted	Partial	Not primary	Not primary	Yes	No explicit real-time risk-to-policy regeneration
Arshad et al. [17], 2023	Automated user-centric zero-trust access control	N/A	Yes	N/A	N/A	Yes	Access-control framework without CP-ABE
Chegenizadeh et al. [19], 2021	Hidden updatable access policies	Not primary	Yes	Yes	No	No	Policy updates are not driven by live risk
Li et al. [20], 2025	Revocable ABE with data integrity	Not primary	Partial	Not primary	Yes	No	Revocation and integrity focus; no adaptive strengthening

3. RESEARCH GAPS

G1. CP-ABE in Cloud-IoT Is Predominantly Static Rather Than Risk-Adaptive

Recent CP-ABE work for IoT improves efficiency through outsourcing and improves practicality through revocation and policy privacy. However, the ciphertext policy is typically fixed at encryption time or changed administratively rather than being strengthened or relaxed automatically according to real-time risk.

G2. Dynamic Updates in ABE Do Not Constitute Continuous Risk-Driven Policy Regeneration

Some IoT ABE frameworks support dynamic attribute updates and even policy-hiding with update mechanisms, but they do not formalize a risk score → policy strength mapping (e.g., LOW risk uses baseline policy; HIGH risk adds MFA/time-window/location constraints).

G3. Zero-Trust and Risk-Aware Access Control Is Often Limited to the Authorization Layer

Risk-aware / Zero Trust access control systems compute context/trust/risk scores and adapt decisions continuously, but enforcement is usually done at the gateway/service layer. If data is copied or accessed later, the decision may not be cryptographically bound to the data itself.

G4. Missing End-to-End Edge-Cloud Workflow Linking Context, Risk Scoring, and ABE Policy Enforcement

Edge security frameworks can adapt policies, and CP-ABE can enforce fine-grained cryptographic access, but there is limited integrated design showing how edge context extraction + real-time scoring triggers CP-ABE policy updates/encryption in an edge–cloud pipeline suitable for IoT.

G5. Limited Evaluation of the Security–Performance Trade-Off Under Adaptive Policies

Efficiency-focused ABE papers evaluate encryption/decryption cost and sometimes outsourcing gains, while ZT/risk papers evaluate decision effectiveness; fewer studies evaluate adaptive CP-ABE on both sides together: (i) security improvement under suspicious contexts and (ii) latency/overhead of frequent policy strengthening (e.g., time-window + MFA attributes).

G6. Policy Privacy and Auditability under Frequent Updates Remain Under-addressed

Policy-hiding and auditing exist for CP-ABE, but when policies are regenerated frequently (due to changing risk), additional leakage risks and audit requirements arise (e.g., observers inferring user state from policy changes). This "adaptive-policy privacy" angle is not fully treated in most current designs.

G7. Revocation and Short-Lived Constraints Require an Integrated Adaptive Strategy

Revocation is well studied in fog and IoT ABE, but combining revocation with short-lived, risk-strengthened constraints such as time windows, verified MFA, and location restrictions requires a cohesive strategy so that security does not depend on a single mechanism.

4. PROBLEM FORMULATION

a) System Model for Edge-Cloud IoT

The system consists of IoT devices or sensors, an edge gateway, a Policy and Risk Engine, a CP-ABE encryption service, cloud storage, an Attribute Authority, and authorized users. A user can decrypt a ciphertext only when the attributes in the user key satisfy the active ciphertext policy.

Let:

- $D = \{d_1, d_2, ..., d_n\}$: set of IoT devices that generate data, such as healthcare, smart-home, or smart-grid devices.
- $U = \{u_1, u_2, ..., u_m\}$: set of users, such as doctors, engineers, or administrators, who possess attribute keys.
- AA: Attribute Authority that runs CP-ABE setup and issues attribute-based secret keys.
- C: cloud storage that retains ciphertext and associated metadata.

b) Attribute Universe and Access Policy

Define the attribute universe $A = A_u \cup A_c$, where:

- User attributes A_u include Role, Department, Organization, and ClearanceLevel.
- Context attributes A_c include TimeWindow, LocationZone, DeviceTrustLevel, and RiskLevel.

A ciphertext policy P is a Boolean formula over these attributes using AND/OR operators. A baseline policy is:

$$P\_base = (Role = Doctor \wedge Department = Cardiology) \vee (Role = Admin \wedge Clearance = High)$$

c) CP-ABE Workflow (Cryptographic Enforcement)

CP-ABE is used as the data-layer access-control primitive:

- $Setup(1^{\lambda}) \rightarrow (PK, MSK)$, where PK denotes the public parameters and MSK denotes the master secret key.
- $KeyGen(MSK, S_u) \rightarrow SK_u$, where SK_u is bound to the user attribute set S_u.
- $Encrypt(PK, m, P_t) \rightarrow CT_t$, where CT_t is generated under policy P_t.
- $Decrypt(PK, CT_t, SK_u) \rightarrow m$ if and only if $S_u \models P_t$.

d) Risk Scoring Model (Real-Time Context → Risk)

At each access time t, the edge Policy and Risk Engine computes a risk score from the context vector:

$$x(t) = [f\_failed, f\_trust, f\_location, f\_time, f\_frequency, ...]$$

Features include failed login count, device trust score (0–1), unusual location, access time, request frequency anomaly, etc.

A rule-based or ML classifier produces:

$$RiskLevel(t) \in \{LOW, MEDIUM, HIGH\}$$

e) Dynamic Policy Generator (Risk → Policy Strength)

A policy generator g(.) maps the current risk level and context to an encryption policy:

$$P_t = g(RiskLevel(t), x(t))$$

Risk-to-policy mapping:

- LOW → baseline policy

- MEDIUM → add one constraint (e.g., LocationZone=HospitalNetwork)
- HIGH → strict policy + short-lived access (e.g., TimeWindow=15min AND DeviceTrustLevel=High; MFA attribute as needed)

f) Threat Model (What the System Must Resist)

The adversary attempts unauthorized access through common Cloud-IoT attack patterns, including stolen credentials used from an abnormal location or time, repeated failed attempts, compromised low-trust devices, and replay or odd-hour access. Elevated risk automatically strengthens the ciphertext policy so that unauthorized decryption fails.

g) Problem Definition (What Must Be Achieved)

Given a stream of IoT data items and access requests over time, design a Risk-Adaptive CP-ABE scheme that:

Security requirements:

a) Confidentiality against untrusted cloud storage and unauthorized users: decryption succeeds only when $S_u \models P_t$.

b) Context-aware enforcement: suspicious contexts automatically trigger a stricter policy P_t.

c) Attack resilience under the stated threat model (credential misuse, replay, compromised device, abnormal login behavior).

Performance requirements (measurable in evaluation):

- minimize encryption/decryption latency and end-to-end latency
- minimize policy update overhead
- limit cloud storage overhead (ciphertext + policy metadata)
- maximize correct allow/deny behavior (legitimate allowed, attacker blocked)

Objectives

Objective 1: Build a secure CP-ABE based access control layer for Cloud–IoT data.

Objective 2: Develop an intelligent risk/context engine that dynamically updates access policies (Low/Medium/High risk → relaxed/strict ABE policy).

Objective 3: Evaluate security and performance trade-offs of adaptive policies versus static ABE (encryption/decryption time, policy update time, correctness of allow/deny, resilience to common attacks, end-to-end latency, storage overhead).

5. PROPOSED METHODOLOGY

This section presents the end-to-end Risk-Adaptive CP-ABE methodology that integrates the CP-ABE access-control layer, the risk and context engine, and the security-performance evaluation.

A. System Architecture and Workflow

Figure-1 shows the closed-loop edge–cloud workflow: IoT data is produced, the edge extracts context, a risk engine computes RiskLevel ∈ {LOW, MEDIUM, HIGH}, a policy generator creates a dynamic CP-ABE policy, encryption is performed, and ciphertext is stored in the cloud; users decrypt only if their attributes satisfy the current policy.

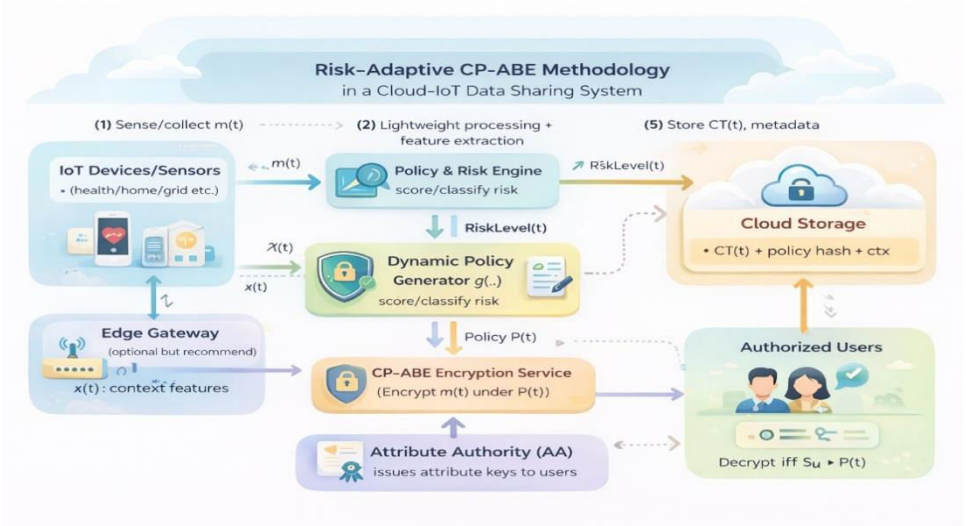

Figure-1. Risk-Adaptive CP-ABE methodology in an edge-cloud Cloud-IoT pipeline.

The architecture comprises IoT devices, an edge gateway, a Policy and Risk Engine, a dynamic policy generator, a CP-ABE encryption service, cloud storage, an Attribute Authority, and authorized users.

B. Attribute Universe and Policy Templates

The attribute universe is defined as:

$$A = A_u \cup A_c$$

User attributes A_u include Role, Department, Organization, and Clearance, whereas context attributes A_c include TimeWindow, LocationZone, DeviceTrustLevel, and RiskLevel.

An example baseline low-risk access policy is:

$$P\_base = (Role = Doctor \wedge Department = Cardiology) \vee (Role = Admin \wedge Clearance = High)$$

C. Risk Scoring Model (Real-Time Intelligence)

At each data or access event time t, the edge gateway extracts the context vector:

$$x(t) = [f\_fail(t), f\_trust(t), f\_loc(t), f\_time(t), f\_freq(t), f\_IP(t)]$$

where typical inputs include failed logins (last 10 min), device trust score (0–1), unusual location, access time (working/odd hours), request frequency anomaly, and optional IP reputation.

Two implementation options are considered: (i) a lightweight rule-based risk engine and (ii) an ML-based classifier. The reported prototype uses the rule-based normalized score described below, while the ML formulation is retained as an extensible alternative.

Option 1: Rule-Based Risk Score (Normalized to 0-1)

The normalized score is computed as a weighted combination of the context features:

$$r(t) = clip(\Sigma_k w_k x_k(t), 0, 1), \text{ with } \Sigma_k w_k = 1$$

The score is mapped to risk classes using two thresholds:

$$RiskLevel(t) = LOW \text{ if } r(t) < \tau_1; MEDIUM \text{ if } \tau_1 \le r(t) < \tau_2; HIGH \text{ if } r(t) \ge \tau_2.$$

Thus, the risk engine returns one of three levels: LOW, MEDIUM, or HIGH.

Option 2: ML-Based Classifier

An alternative classifier, such as logistic regression, a decision tree, or a random forest, can predict the risk class from x(t).

For logistic regression, an illustrative high-risk probability is:

$$p(HIGH \mid x(t)) = \sigma(w^T x(t) + b)$$

The final RiskLevel(t) is assigned using the class with the highest predicted probability.

D. Risk-Adaptive Policy Generation

The policy generator maps LOW risk to the baseline policy, MEDIUM risk to the baseline policy plus one contextual constraint, and HIGH risk to a strict, short-lived policy.

Formally:

$$P(t) = g(RiskLevel(t), x(t)) = P\_base \wedge \Delta(RiskLevel(t), x(t))$$

The policy-strengthening template used in the prototype is summarized in Table-2.

Table-2. Risk-to-policy strengthening templates.

Risk Level(t)	Added constraints Δ(.)	Example
LOW	None	P(t) = P_base
MEDIUM	One contextual constraint	P_base ∧ (LocationZone = HospitalNetwork)
HIGH	Strict, time-bounded, and high-trust constraints	P_base ∧ LocationZone ∧ MFA ∧ TimeWindow ∧ DeviceTrustLevel

E. CP-ABE Cryptographic Workflow

The implementation follows the standard CP-ABE pipeline: Setup, KeyGen, Encrypt, and Decrypt.

- Setup by the Attribute Authority:

$$(PK, MSK) \leftarrow Setup(1^\lambda)$$

- Key generation for user u with attribute set S_u:

$$SK_u \leftarrow KeyGen(MSK, S_u)$$

- Encryption of IoT message m(t) under dynamic policy P(t):

$$CT(t) \leftarrow Encrypt(PK, m(t), P(t))$$

- Decryption succeeds only when the user attributes satisfy the policy:

$$m(t) \leftarrow Decrypt(PK, CT(t), SK_u) \text{ if and only if } S_u \vDash P(t)$$

Cloud stores ciphertext plus policy hash/context metadata.

F. Algorithm

Algorithm: Risk-Adaptive Encryption

a) The IoT device generates data m(t) and telemetry.

b) The edge gateway extracts context features x(t).

c) The risk engine computes RiskLevel(t) ∈ {LOW, MEDIUM, HIGH}.

d) The dynamic policy generator produces P(t) = g(RiskLevel(t), x(t)).

e) The system encrypts m(t) under P(t) and uploads CT(t) to the cloud.

f) On access, the user decrypts only if $S_u \vDash P(t)$.

G. Threat-Driven Validation and Evaluation Plan

The evaluation tests stolen credentials used from the wrong location or time, excessive failed attempts, compromised low-trust devices, and replay at odd hours. The expected response is automatic policy strengthening followed by denial of unauthorized decryption.

Static ABE and Risk-Adaptive ABE are compared using encryption and decryption time, policy-update time, access-decision correctness, storage overhead, end-to-end latency, and attack resilience.

6. RESULTS AND DISCUSSIONS

A. Experiment Context and Threat Coverage

The proposed Risk-Adaptive CP-ABE framework was evaluated in an edge-cloud Cloud-IoT pipeline comprising 10 IoT nodes, an edge gateway, and cloud storage.

The test suite covered brute-force, stolen-credential, and replay attacks and applied time-bounded semantics, including a 15-minute access window, in strict policies.

B. Risk-Driven Policy Tightening (Core Objective Validation)

The logs confirm that the edge gateway dynamically switches policies based on computed risk:

- Low risk (0.1) uses the baseline CP-ABE policy: Role = Doctor and Department = Cardiology.
- High risk (1.0) escalates to a stricter policy by adding Location = Hospital Zone, MFA = Verified, and TimeWindow = 15 min.

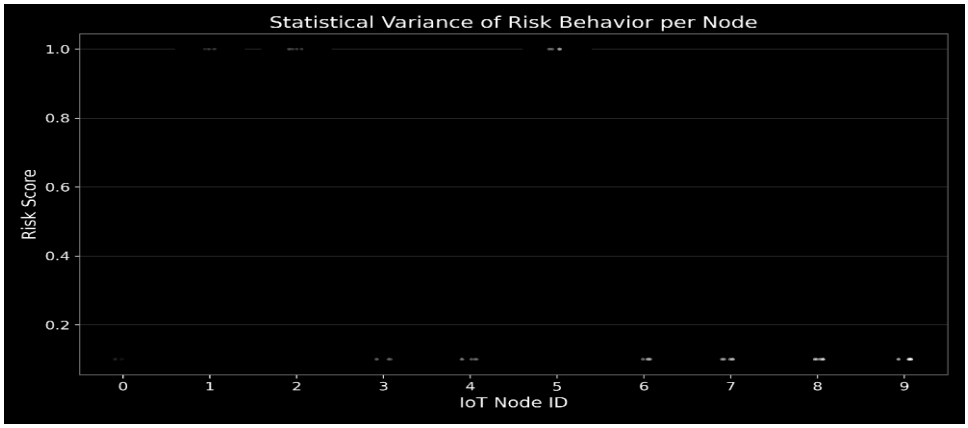

Figure-2. Statistical variance of risk behavior per node.

Discussion. The per-node view shows that only a subset of nodes repeatedly exhibits high risk, whereas the remaining nodes stay in a low-risk state. This behavior is consistent with the attack-injection scenario, in which selected nodes represent adversarial activity.

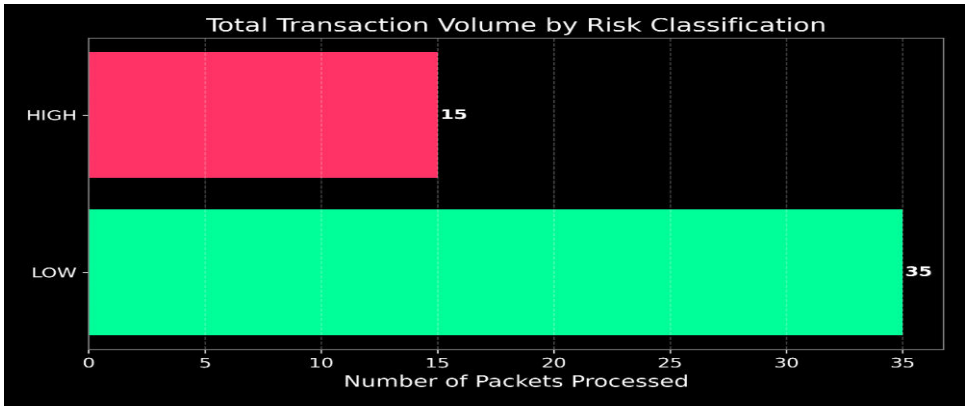

Figure-3. Total transaction volume by risk classification.

From the recorded metrics for 50 encryption events, the system processed 35 LOW-risk and 15 HIGH-risk transactions; no MEDIUM-risk samples occurred in this run. Policy escalation is therefore selective rather than continuously active, as intended.

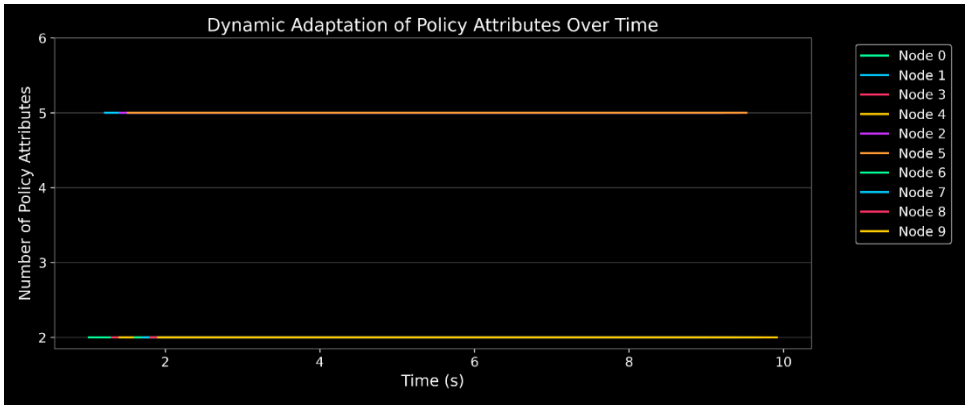

Figure-4. Dynamic adaptation of policy attributes over time.

Discussion. The step plot shows that policy complexity remains at two attributes during LOW risk and increases to five attributes during HIGH risk, confirming that runtime risk changes directly trigger policy strengthening.

Table-3. Policy adaptation and ciphertext complexity (computed from metrics log).

Risk Level	Policy attribute count	Packet Size (Bytes)	Interpretation
LOW	2	1152	Baseline access policy (minimal constraints)
HIGH	5	1344	Tightened policy (+location +MFA +time window)

C. Attacker Node Profiling and Spatiotemporal Risk Localization

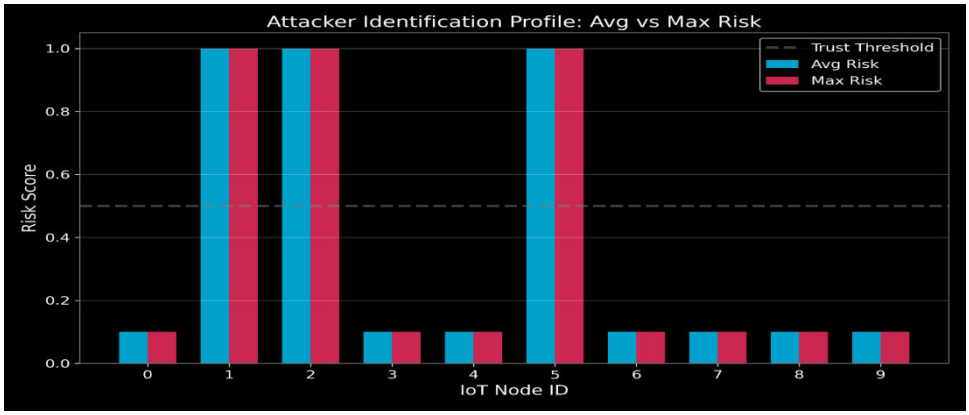

Figure-5. Attacker identification profile (avg vs. max risk).

A trust-threshold line at 0.5 visually separates benign and suspicious node behavior. Nodes that repeatedly cross the threshold are clearly identified as high-risk candidates, consistent with the repeated HIGH-risk pattern in the recorded log.

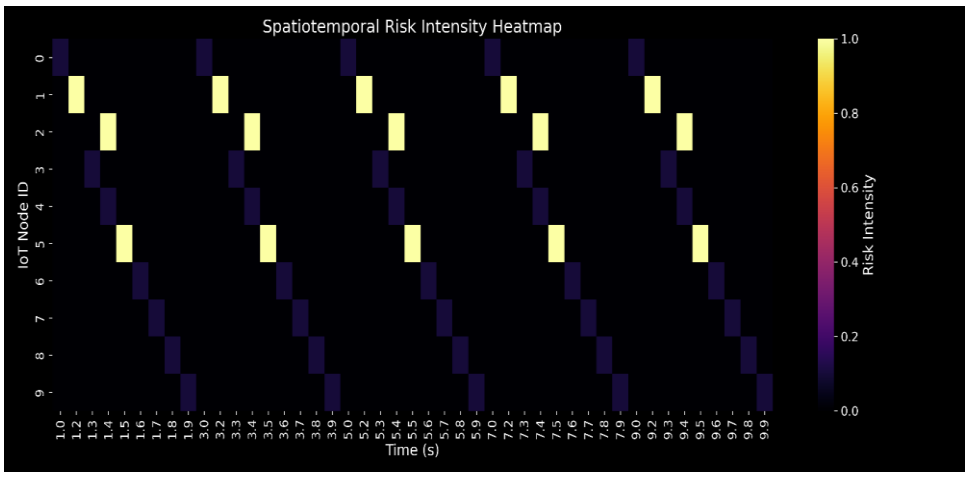

Figure-6. Spatiotemporal risk intensity heatmap.

Discussion. The heatmap concentrates high-risk intensity within specific node-time regions rather than across all nodes, indicating that the risk engine can localize suspicious activity for edge enforcement and auditing.

D. Communication/Storage Overhead Under Adaptive Enforcement

The final evaluation yielded an average encryption latency of 12.1 ms, an average decryption latency of 21.6 ms, and cloud storage usage of 59.0625 KB.

Table-4. Performance overhead measured in the final evaluation.

Metric	Value
Average encryption latency	12.1 ms
Average decryption latency	21.6 ms
Cloud storage usage	59.0625 KB

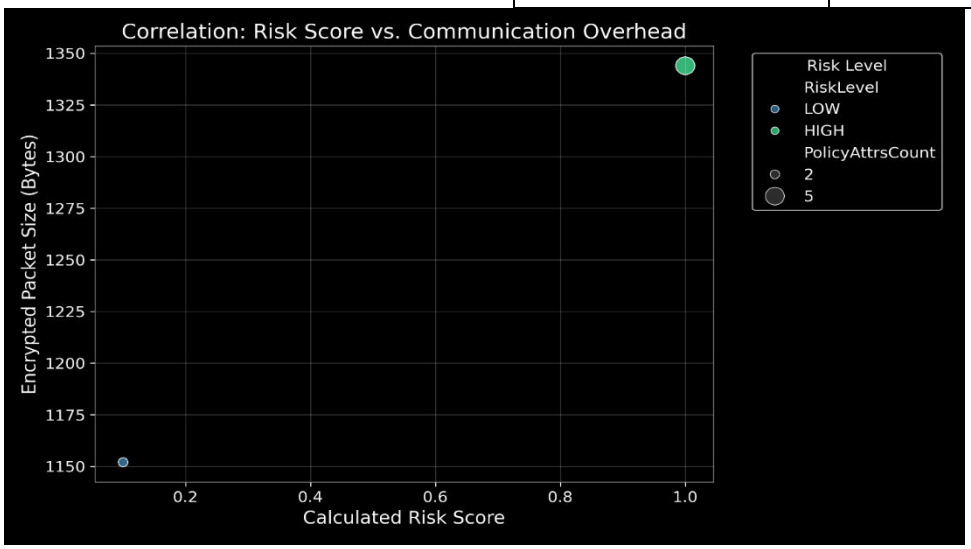

Figure-7. Correlation: risk score vs. communication overhead.

The correlation plot shows that ciphertext size increases with risk because high-risk events include more policy attributes. Marker size represents the policy-attribute count.

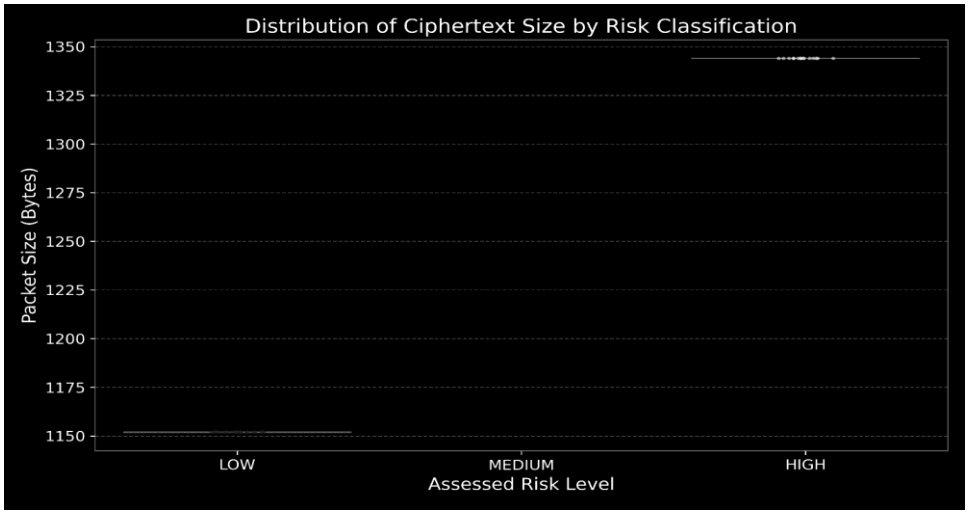

Figure-8. Distribution of ciphertext size by risk classification.

According to the metrics log, ciphertext size increases from 1152 bytes at LOW risk to 1344 bytes at HIGH risk. The communication increase is:

$$\Delta\_packet = ((1344 - 1152) / 1152) \times 100 = 16.67\%$$

This 16.67% increase quantifies the communication trade-off associated with enforcing additional constraints under suspicious contexts.

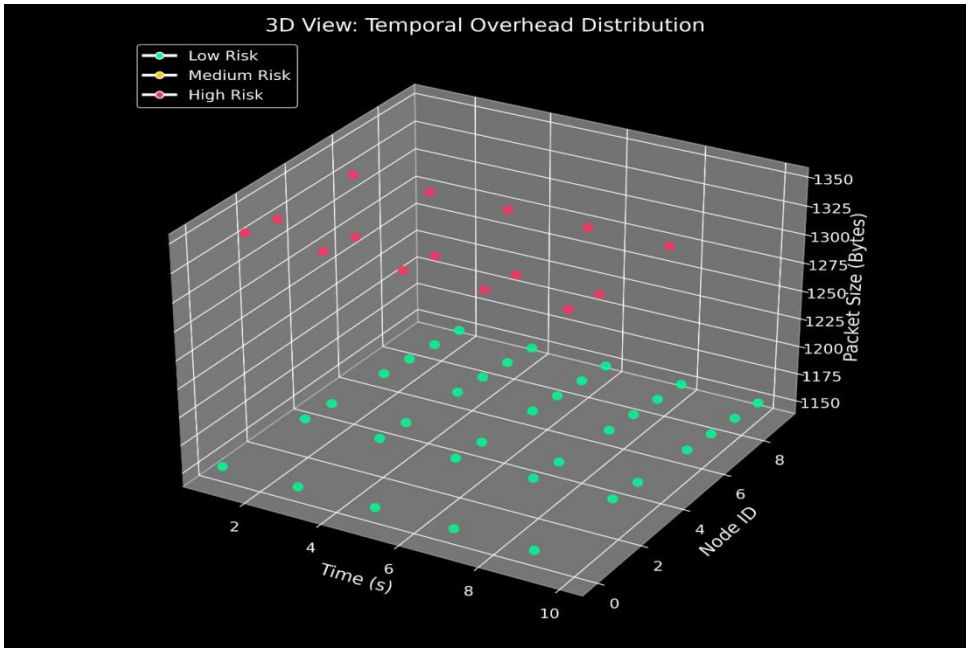

Figure-9. 3D view: temporal overhead distribution.

The three-dimensional plot shows that larger ciphertexts align with high-risk events at specific nodes and times, confirming that the additional overhead is event-driven rather than constant.

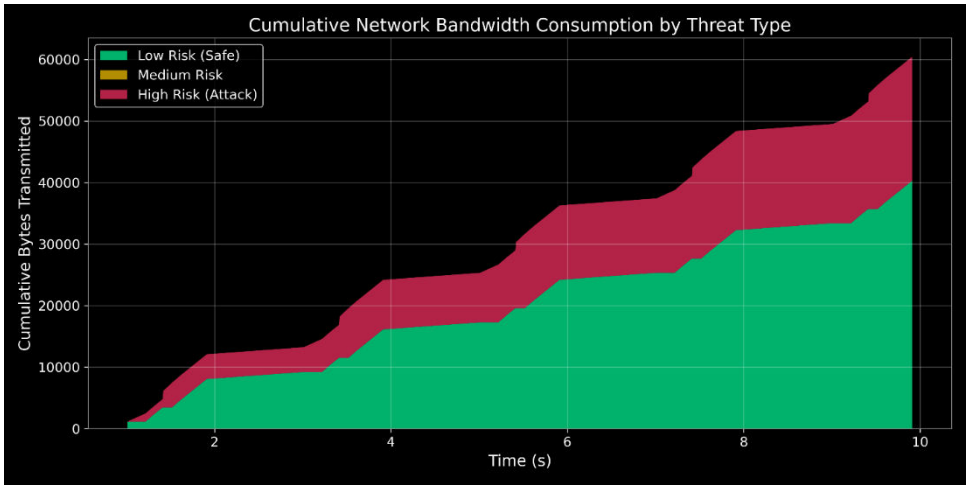

Figure-10. Cumulative network bandwidth consumption by threat type (risk).

Cumulative transmitted bytes increase more rapidly during high-risk intervals because those ciphertexts contain stricter policy material. The system therefore incurs additional bandwidth only when the estimated threat level rises.

E. System Stability (Risk Smoothing and Burst Behavior)

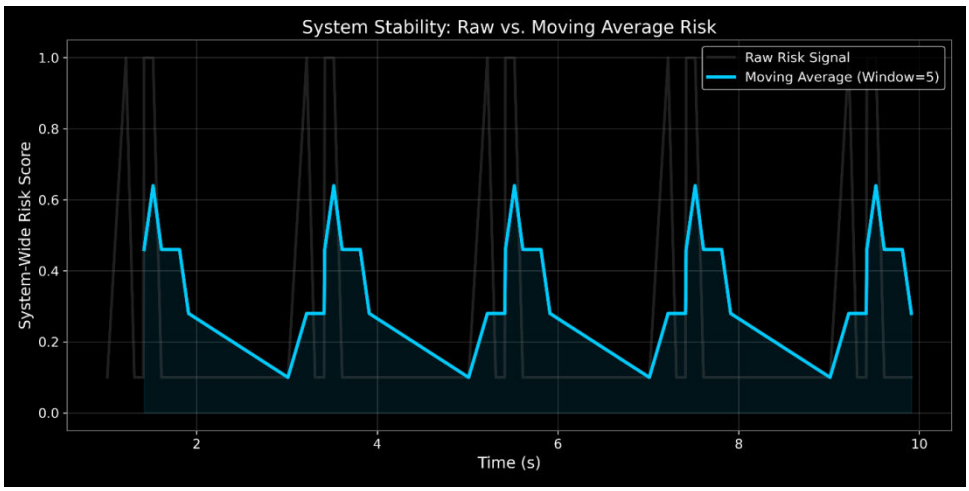

Figure-11. System stability: raw vs. moving average risk (Window = 5).

The raw risk signal contains short attack-related spikes, whereas the moving average reduces volatility. The moving-average peak remains below the high-risk threshold of 0.7, indicating bursty rather than persistent high-risk activity.

F. Security Effectiveness and Baseline Comparison

The system mitigated all 15 evaluated attacks, corresponding to 100% adaptive defense accuracy.

Table-5. Security results.

Measure	Value
Total attacks generated	15
Attacks mitigated	15
Adaptive defense accuracy	100%

Table-6. Comparative analysis of the proposed and static ABE configurations.

Metric	Static ABE (Low)	Static ABE (High)	Proposed (Edge)
Threat mitigation rate	0% (Fail)	100%	100%
Legitimate latency	10.0 ms	17.5 ms	10.0 ms
Average system latency	10.0 ms	17.5 ms	12.1 ms

Static-LOW minimizes latency but fails to mitigate the evaluated threats, whereas static-HIGH mitigates the threats at the cost of consistently higher latency. The proposed risk-adaptive approach achieves 100% mitigation while preserving a 10.0 ms legitimate-access latency and a moderate 12.1 ms average system latency. Security hardening is therefore activated only when the estimated risk justifies it.

7. CONCLUSIONS

This work presented a Risk-Adaptive CP-ABE framework for Cloud-IoT in an edge-cloud architecture, where the edge gateway computes real-time risk and dynamically strengthens the CP-ABE access policy. In the experiment with 10 IoT nodes, LOW risk (0.1) activated the baseline policy, Role = Doctor and Department = Cardiology, whereas HIGH risk (1.0) added Location = Hospital Zone, MFA = Verified, and TimeWindow = 15 min.

KEY FINDINGS

- Under the evaluated threat set (brute force, stolen credentials, replay), the proposed system mitigated 15 out of 15 attacks, achieving 100% adaptive defense accuracy.
- Performance results show that the adaptive security comes with manageable overhead: 12.1 ms average encryption latency, 21.6 ms average decryption latency, and 59.0625 KB cloud storage usage for the recorded run.
- The comparative analysis further indicates that risk-adaptive enforcement attains the security level of an "always strict" static configuration while avoiding its constant latency penalty, since tightening is activated only when risk increases.

FUTURE WORK

a) Richer Risk Modeling (Beyond Low/High): Extend the evaluation to include a well-populated MEDIUM-risk regime and study policy transitions (hysteresis/threshold tuning) to reduce unnecessary oscillations while preserving fast reaction.

b) Learning-Based Risk Engine: Replace or augment the current scoring with an ML model trained from labeled access/session behavior to better generalize across attack types and reduce false alarms, while keeping edge inference lightweight.

c) Scalability and Deployment Realism: Evaluate at larger scale (more nodes, higher traffic rates) and under more diverse network conditions (delays/loss), reporting throughput and edge CPU/memory impact.

d) Revocation and Key Lifecycle under Adaptive Policies: Integrate efficient attribute and key revocation with short-lived credentials so that risk-based policy strengthening, including time windows, operates seamlessly during user and device churn.

e) Policy Privacy and Metadata Leakage: Investigate policy-hiding or encrypted policy metadata so that frequent policy strengthening does not unintentionally reveal sensitive operational context (e.g., which nodes are under suspicion).

f) Broader Threat Coverage and Formal Assurance: Expand attack coverage (insider misuse, collusion, compromised gateway) and provide formal security arguments (or proofs) for the adaptive construction under the chosen threat model.

REFERENCES

[1] J. Bethencourt, A. Sahai and B. Waters. 2007. Ciphertext-policy attribute-based encryption. in Proc. IEEE Symp. Security and Privacy (SP), Oakland, CA, USA, pp. 321-334, doi: 10.1109/SP.2007.11.

[2] A. Sahai and B. Waters. 2005. Fuzzy identity-based encryption. in Advances in Cryptology - EUROCRYPT 2005, LNCS 3494, Springer, pp. 457-473, doi: 10.1007/11426639_27.

[3] A. Lewko and B. Waters. 2011. Decentralizing attribute-based encryption. in *Proc. EUROCRYPT*, pp. 568-588, doi: 10.1007/978-3-642-20465-4_31.

[4] B. Waters. 2011. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. in Proc. PKC, 2011, pp. 53-70, doi: 10.1007/978-3-642-19379-8_4.

[5] R. Sarma and F. A. Barbhuiya. 2022. A secure and efficient access control scheme with attribute revocation and merging capabilities for fog-enabled IoT. Computers and Electrical Engineering, vol. 104, pt. B, art. 108449, doi: 10.1016/j.compeleceng.2022.108449.

[6] A. Peñuelas-Angulo, C. Feregrino-Uribe and M. Morales-Sandoval. 2023. Revocation in attribute-based encryption for fog-enabled internet of things: A systematic survey. Internet of Things, vol. 22, art. 100827, doi: 10.1016/j.iot.2023.100827.

[7] Y. Lou *et al*. 2023. DACP: Enforcing a dynamic access control policy in cross-domain environments. Computer Networks, vol. 234, art. 110049, doi: 10.1016/j.comnet.2023.110049.

[8] D. Halgamuge and D. Niyato. 2025. Adaptive edge security framework for dynamic IoT security policies in diverse environments. *Computers & Security*, vol. 146, art. 104128, doi: 10.1016/j.cose.2024.104128.

[9] Z. Xu *et al*. 2025. A secure and scalable IoT access control framework with dynamic attribute updates and policy hiding. Scientific Reports, vol. 15, art. 11913, doi: 10.1038/s41598-025-80307-3.

[10] L. Wang *et al*. 2023. Ciphertext-policy attribute-based encryption supporting policy-hiding and cloud auditing in smart health. Computer Standards & Interfaces, vol. 86, art. 103696, doi: 10.1016/j.csi.2023.103696.

[11] S. Li, Y. Duan, J. Cao, J. Qin and D. Zheng. 2023. Online/Offline MA-CP-ABE with cryptographic reverse firewalls. *Entropy*, vol. 25, no. 4, art. 616, doi: 10.3390/e25040616.

[12] M. Gasmi *et al*. 2024. Load-balanced attribute-based outsourced encryption for constrained IoT devices. Computers and Electrical Engineering, vol. 112, art. 109424, doi: 10.1016/j.compeleceng.2023.109424.

[13] M. Mahdavi, R. Mokhtari, and A. Pashazadeh. 2024. IoT-friendly attribute-based encryption with pre-computed and outsourced encryption. Future Generation Computer Systems, 150: 359-374, doi: 10.1016/j.future.2023.08.015.

[14] W. Huang *et al*. 2023. A combination of zero trust access control with attribute-based encryption for secure cloud-IoT. *Ad Hoc Networks*, vol. 150, art. 103161, doi: 10.1016/j.adhoc.2023.103161.

[15] K. Routray and P. Bera. 2024. Risk-Aware Lightweight Data Access Control for Cloud-Assisted IIoT: A Zero-Trust Approach. in Proc. SIGCOMM Workshop on Zero Trust Architecture for Next Generation Communications (ZTA-NextGen), Sydney, NSW, Australia, pp. 40-42, doi: 10.1145/3672200.3673880.

[16] S. Rose, O. Borchert, S. Mitchell and S. Connelly. 2020. Zero Trust Architecture. NIST Special Publication 800-207, doi: 10.6028/NIST.SP.800-207.

[17] M. Arshad *et al*. 2023. AU-ZTAC: Automated and user-centric zero trust access control framework for IoT networks. Electronics, vol. 12, no. 21, art. 4608, doi: 10.3390/electronics12214608.

[18] J. Cui *et al*. 2020. BC-SABE: Blockchain-based searchable attribute-based encryption for industrial IoT. IEEE Internet of Things Journal, 7(10): 9550-9561, doi: 10.1109/JIOT.2020.2993231.

[19] M. Chegenizadeh, M. Ali, J. Mohajeri and M. R. Aref. 2021. HUAP: Practical attribute-based access control supporting hidden updatable access policies for resource-constrained devices. arXiv: 2107.10133.

[20] Y. Li et al. 2025. Fast revocable attribute-based encryption with data integrity for Internet of Things. arXiv: 2509.20796.