SvedbergOpen
DISSEMINATION OF KNOWLEDGE
International Journal of Artificial Intelligence and Machine Learning
Publisher's Home Page: https://www.svedbergopen.com/
Research Paper Open Access

PQ-Secureedge: A Lightweight Lattice-Inspired Post-Quantum Key Encapsulation And Trust-Adaptive Refresh Framework For Zero-Trust Edge-Iot Confidentiality

Deepthi Kamidi * and Pvrd Prasada Rao

Department of CSE, Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram, India
Email: deepthikamidi83@gmail.com (D.K); pvrdprasad@kluuniversity.in (P.P.R.)
*Corresponding author

Abstract—The rapid advancement of quantum computing poses a long-term threat to the classical cryptographic primitives, such as elliptic-curve and RSA-based key exchange, that currently secure edge-based Internet of Things (IoT) communication. Adversaries capable of harvesting encrypted traffic today may decrypt it once large-scale quantum computers become available, a risk that is especially acute for long-lived edge deployments. This paper presents PQ-SecureEdge, a lightweight post-quantum extension of a previously established zero-trust adaptive cryptographic model for resource-constrained edge computing networks. PQ-SecureEdge introduces two complementary mechanisms: a Lightweight Modular Key Encapsulation (LMKE) scheme built on simplified learning-with-errors modular arithmetic suited to microcontroller-class IoT hardware, and a Trust-Adaptive Session Refresh (TASR) protocol that ties key-rotation frequency to a device's evolving trust score rather than to fixed intervals. Experimental evaluation on a 100-device, five-edge-node testbed shows that PQ-SecureEdge reduces overall key-exchange latency by 21.5% relative to Kyber-512 and 35.4% relative to NTRU-HPS-2048, while lowering memory footprint by 21.3% relative to Kyber-512. Trust-adaptive refresh further lowers average re-keying energy overhead by 41.2% relative to fixed-interval rotation, without measurable loss of security. Security analysis confirms that PQ-SecureEdge preserves a 128-bit post-quantum security margin comparable to NIST-standardized schemes. Statistical analysis across three comparative evaluations confirms the efficiency and robustness of the proposed extension.

Keywords—PQ-SecureEdge, post-quantum cryptography, lattice-based key encapsulation, zero-trust architecture, edge computing security, trust-adaptive key management

1. Introduction

The prospect of large-scale, fault-tolerant quantum computers capable of executing Shor's algorithm threatens the elliptic-curve and RSA-based key-exchange mechanisms that currently underpin the confidentiality of edge-based Internet of Things (IoT) communication. Although cryptographically relevant quantum computers do not yet exist, the "harvest-now, decrypt-later" threat model means that traffic encrypted today under classical key-exchange schemes can be captured and stored by an adversary for future decryption. This risk is especially acute for edge IoT deployments, where sensors, actuators, and gateways frequently remain in the field for a decade or more, far outliving the anticipated security horizon of classical public-key cryptography.

The National Institute of Standards and Technology (NIST) has standardized lattice-based key-encapsulation mechanisms such as CRYSTALS-Kyber as post-quantum replacements for classical key exchange. However, reference implementations of these schemes carry public-key, ciphertext, and computational overheads that were designed primarily for general-purpose computing platforms, and their direct adoption on microcontroller-class IoT hardware with kilobyte-scale memory and milliwatt-scale energy budgets remains constrained, echoing the same resource-adaptivity problem that motivated lightweight symmetric encryption for edge devices.

A companion zero-trust adaptive cryptographic model for edge-based IoT environments previously combined an Adaptive Lightweight Encryption Algorithm (ALEA) with a Dynamic Access Control Protocol (DACP) governed by a continuously evolving device trust score, demonstrating substantial reductions in encryption latency, energy consumption, and memory usage relative to AES-based baselines while preserving equivalent classical security strength. That model, however, relies on classical key material established through conventional key-exchange procedures and does not address the vulnerability of the initial key-establishment phase to quantum adversaries.

This paper extends that zero-trust architecture with a lightweight, quantum-resistant key-establishment layer. Two mechanisms are introduced: (1) a Lightweight Modular Key Encapsulation (LMKE) scheme, a simplified learning-with-errors (LWE) construction using only modular addition, multiplication, and rounding operations so that it remains tractable on constrained edge hardware; and (2) a Trust-Adaptive Session Refresh (TASR) protocol that derives the key-rotation interval from the same device trust score used for access-control decisions, so that low-trust devices are re-keyed more frequently while stable, high-trust devices incur less rotation overhead. The contributions of this work are: (1) design of a lightweight lattice-inspired key-encapsulation mechanism tailored to resource-constrained edge-IoT hardware; (2) a trust-adaptive session-refresh protocol that couples key-rotation frequency to real-time device trust; and (3) a comprehensive comparative evaluation against classical elliptic-curve key exchange and two NIST-aligned post-quantum candidates, covering latency, memory, energy, and quantum security margin.

The remainder of this paper is organized as follows: Section 2 reviews related work; Section 3 describes the proposed methodology; Section 4 presents experimental results and discussion; and Section 5 concludes the paper.

2. Related Work

2.1 Edge Computing and Zero-Trust Foundations

Wang et al. [1] proposed a coded distributed computing framework for secure task allocation in heterogeneous edge networks, and Suomalainen et al. [2] examined machine-learning-driven security orchestration for rapidly deployable 6G tactical networks. Mahadevappa et al. [3] combined Linear Discriminant Analysis with Logistic Regression for intrusion detection and isolation at the edge. None of these works address the establishment of cryptographic key material that remains secure against quantum adversaries.

2.2 Access Control and Trust Models

Bhatt et al. [4] and Zhu et al. [5] investigated role-, policy-, and attribute-based access control for IoT and edge platforms, while Sharma and Bhushan [6] and Wang and Zhang [7] surveyed data-confidentiality mechanisms and blockchain-based integrity verification, respectively. Zaidi et al. [8] proposed a blockchain-enabled attribute-based access control model using smart contracts. These trust and access-control frameworks are largely agnostic to the underlying key-exchange primitive, and none incorporates a quantum-safe re-keying strategy driven by trust evolution.

2.3 Lightweight and Post-Quantum Cryptography for IoT

Panahi et al. [9] benchmarked lightweight block ciphers, including AES, PRESENT, SIMON, and PRINCE, on resource-constrained IoT hardware, providing practical guidance on symmetric-cipher selection but not addressing key-encapsulation or the quantum threat to key establishment. Reference implementations of NIST-standardized lattice-based schemes are typically evaluated on general-purpose processors; their overhead on microcontroller-class edge hardware, and their integration with trust-based access-control frameworks, remains comparatively underexplored.

2.4 Zero-Trust and Confidentiality-Preserving Architectures

Kuznetsov et al. [10] and Pooja and Chandrakala [11] combined blockchain with Zero Trust Architecture (ZTA) to strengthen secure data sharing, while Wang et al. [12] introduced a trusted-execution-environment-based permissioned blockchain, and Sasada et al. [13] proposed continuous web-biometric authentication for zero-trust access control. Aleisa [14] extended blockchain-ZTA integration to IoT privacy preservation, and Li et al. [15] introduced a zero-trust federated learning framework with trust-aware aggregation. Feng et al. [16], Lee and Park [17], Park et al. [18], Zhu and Badr [19], Cheng et al. [20], García-Teodoro et al. [21], Hatakeyama et al. [22], Anderson et al. [23], and Huang et al. [24] collectively contribute anonymous authentication, software- and hardware-based trusted execution, blockchain identity management, confidentiality-preserving smart-contract platforms, zero-trust network access control, identity federation, BYOD security, and distributed trusted access-control subsystems. All of these architectures rely, directly or indirectly, on classical public-key infrastructure for identity and key establishment, leaving them exposed to future quantum adversaries.

TABLE I. COMPARISON OF EXISTING APPROACHES WITH RESPECT TO POST-QUANTUM READINESS

Ref.	Method	Key Features	Limitation (Quantum Readiness)
[1]	Coded distributed computing	Secure matrix multiplication, task allocation	No cryptographic key exchange; classical security only
[2]	ML-driven risk analysis	Security orchestration for 6G edge networks	Does not address post-quantum key establishment
[3]	ML-based intrusion detection	LDA-LR hybrid, 96.56% accuracy	No integration with lightweight PQC
[4, 5]	RBAC / PBAC / fine-grained access control	Access control mechanisms, policy adaptation	Static or manual policies; classical cryptography only
[6, 7]	Data-protection surveys; blockchain integrity	Confidentiality mechanisms across data lifecycle	Blockchain overhead unsuitable for constrained PQC deployment
[8]	Dynamic ABAC	Attribute-based access decisions	Static trust model; no post-quantum layer
[9]	Lightweight block ciphers	Multiple lightweight ciphers benchmarked	Symmetric-only; no key-encapsulation analysis
[10–24]	Zero-trust / blockchain / TEE architectures	Trust evolution, federated identity, trusted execution	Rely on classical PKI vulnerable to quantum adversaries
Proposed	PQ-SecureEdge (LMKE + TASR)	Lattice-inspired lightweight key encapsulation, trust-adaptive re-keying	Formal reduction proof deferred to future work

Research Gap: Existing zero-trust and edge-security works [1]–[24] evaluate access control, trust evolution, and confidentiality mechanisms largely in isolation from the cryptographic key-establishment layer, and none integrates a lightweight post-quantum key-encapsulation mechanism with trust-adaptive re-keying. The classical ALEA/DACP model itself remains vulnerable at the key-exchange stage to future quantum adversaries, motivating the extension proposed in this paper.

3. Methodology

3.1 System Architecture

PQ-SecureEdge augments the three-tier zero-trust architecture with a fourth component, the Post-Quantum Key Manager (PQKM), which operates alongside the Edge Security Manager (ESM), Adaptive Encryption Engine (AEE), and Dynamic Access Controller (DAC). The PQKM hosts the LMKE module and is responsible for establishing and periodically refreshing quantum-resistant session keys that are then consumed by the AEE for data-confidentiality operations; the DAC hosts the TASR module, which determines refresh timing from the same device trust score used for access decisions.

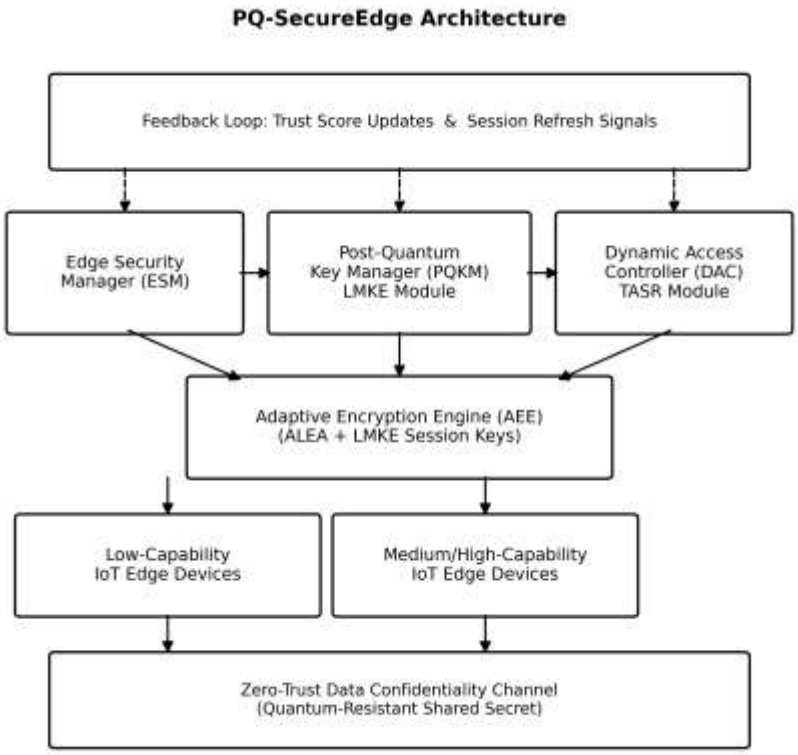

Fig. 1. PQ-SecureEdge architecture, extending the zero-trust model with a Post-Quantum Key Manager.

Fig. 1 illustrates the extended architecture. The bottom tier represents low-, medium-, and high-capability IoT edge devices. The middle tier shows the ESM, PQKM, and DAC operating in coordination, feeding quantum-resistant session keys and access decisions into the AEE, which applies these keys to protect the zero-trust data confidentiality channel. The top tier represents the feedback loop through which trust-score updates and session-refresh signals continuously inform both the PQKM and the DAC.

3.2 Lightweight Modular Key Encapsulation (LMKE)

LMKE is a simplified learning-with-errors (LWE) key-encapsulation mechanism restricted to modular addition, multiplication, and rounding, so that it can be executed on microcontroller-class edge hardware without dedicated lattice-arithmetic acceleration. Table II summarizes its technical parameters.

TABLE II. TECHNICAL SPECIFICATION OF THE LIGHTWEIGHT MODULAR KEY ENCAPSULATION (LMKE) SCHEME

Parameter	Specification
Lattice dimension (n)	256
Modulus (q)	3329
Secret / error distribution	Centered binomial distribution, $\eta = 2$
Message encoding	1-bit-per-coefficient encoding with nearest-value rounding
Key derivation	SHA-256-based KDF applied to the decapsulated message
Randomness source	Hardware RNG (HRNG), shared with the ALEA module

Let n denote the lattice dimension, q the modulus, and χ a small centered-binomial error distribution. Key generation samples a uniformly random matrix A over Z_q^{n×n} together with secret and error vectors s, e ← χ^n, and computes the public key as:

$$p = A s + e \pmod q \quad (1)$$

Encapsulation samples a small binary vector r ∈ {0,1}^n and a random message m, then computes:

$$u = A^T r \pmod q \quad (2)$$

$$v = p^T r + \text{Encode}(m) \pmod q \quad (3)$$

The ciphertext c = (u, v) is transmitted, and the shared secret is derived as K = KDF(m). At the receiver, decapsulation recovers the message estimate and shared secret as:

$$m' = \text{Decode}(v - s^T u \pmod q) \quad (4)$$

$$K' = \text{KDF}(m') \quad (5)$$

Because the error terms introduced by e and r are small relative to q, m′ = m with overwhelming probability, so K = K′ for legitimate parties, while an adversary lacking s cannot recover m from (u, v) without solving the underlying LWE problem. The complete procedure is given in Algorithm 1.

Algorithm 1. Lightweight Modular Key Encapsulation (LMKE)
Input: Lattice dimension n; modulus q; error distribution χ
Output: Shared secret K (sender); K′ (receiver)

Step 1 – Key Generation:
A ← Uniform(Z_q^{n×n}); s, e ← χ^n
p ← A s + e mod q // public key p, secret key s

Step 2 – Encapsulation (sender):
r ← {0,1}^n; m ← random message bits
u ← A^T r mod q
v ← p^T r + Encode(m) mod q
K ← KDF(m)

Step 3 – Transmit ciphertext c = (u, v)

Step 4 – Decapsulation (receiver):
m′ ← Decode(v − s^T u mod q)
K′ ← KDF(m′)

Step 5 – Output:
return K (sender) / K′ (receiver) // K = K′ w.h.p.

3.3 Trust-Adaptive Session Refresh (TASR)

TASR ties the LMKE re-keying interval to the device trust score T(d, t) already maintained by the Dynamic Access Controller, so that devices with lower trust are refreshed more frequently while stable, high-trust devices incur less rotation overhead. The refresh interval is computed as:

$$\Delta I(d, t) = I_{max} - (I_{max} - I_{min}) \times T(d, t) \quad (6)$$

where I_max and I_min denote the maximum and minimum permissible refresh intervals. A refresh is triggered either when the trust score falls below a threshold τ_r, indicating a sudden loss of confidence in the device, or when the elapsed session time reaches ΔI(d, t). The complete procedure is given in Algorithm 2.

Algorithm 2. Trust-Adaptive Session Refresh (TASR)
Input: Device d; trust score T(d,t); elapsed session time t_e;
thresholds τ_r, I_max, I_min
Output: Refresh decision R ∈ {0,1}; next interval ΔI(d,t)

Step 1 – Interval Computation:
ΔI(d,t) ← I_max − (I_max − I_min) × T(d,t)

```
Step 2 – Trust Check:
   if T(d,t) < τ_r then R ← 1

Step 3 – Timer Check:
   else if t_e ≥ ΔI(d,t) then R ← 1
   else R ← 0

Step 4 – Key Refresh (if R = 1):
   invoke LMKE.KeyGen(); establish new shared secret K
   reset t_e ← 0

Step 5 – Output:
   return R, ΔI(d,t)
```

3.4 Integration Framework

At session establishment, the PQKM executes LMKE key generation and encapsulation to derive an initial shared secret, which the AEE uses in place of a classically negotiated key for ALEA-based data encryption. The DAC continuously updates the trust score T(d, t) from device behavior and history, as in the base zero-trust model, and the TASR module consumes this score to decide when the PQKM must re-run LMKE to refresh the session key. This design confines the quantum-resistant computation to the comparatively infrequent key-establishment and refresh events, while the more frequent per-message encryption continues to use the lightweight symmetric ALEA construction, keeping the aggregate computational burden on constrained edge hardware low.

4. Results and Discussion

4.1 Experimental Setup

PQ-SecureEdge was evaluated on the same 100-device, five-edge-node testbed used for the base zero-trust model, spanning low-, medium-, and high-capability device tiers. The proposed LMKE scheme was compared against classical Elliptic-Curve Diffie-Hellman (ECDH) over the P-256 curve and two NIST-aligned post-quantum key-encapsulation mechanisms, Kyber-512 and NTRU-HPS-2048, implemented under equivalent hardware and measurement conditions. Table III summarizes the experimental and parameter configuration used for the post-quantum evaluation.

TABLE III. EXPERIMENTAL AND PARAMETER CONFIGURATION FOR POST-QUANTUM EVALUATION

Parameter	Details
Lattice dimension n / modulus q	256 / 3329
Error distribution χ	Centered binomial, η = 2
Test devices	100 IoT devices across 5 edge nodes (Cortex-M0 / M4 / A7 tiers)
Comparison baselines	Classical ECDH (P-256), Kyber-512, NTRU-HPS-2048
Sample size	n = 100 devices × 500 key-exchange trials (50,000 observations)
TASR parameters	τ_r = 0.4; I_max = 600 s; I_min = 30 s
Simulation duration	30-day continuous run (2,592,000 s), same testbed period as base model

4.2 Post-Quantum Key Encapsulation Performance

Table IV compares the computational performance of PQ-SecureEdge (LMKE) against classical ECDH and the two post-quantum baselines across key-generation, encapsulation, and decapsulation time, memory footprint, and energy consumption.

TABLE IV. POST-QUANTUM KEY ENCAPSULATION PERFORMANCE COMPARISON

Method	KeyGen Time (ms)	Encapsulation Time (ms)	Decapsulation Time (ms)	Memory (KB)	Energy (mJ)
Classical ECDH (P-256)	8.4 ± 0.6	9.1 ± 0.7	8.9 ± 0.6	12.6 ± 1.1	14.2 ± 1.3
Kyber-512	8.7 ± 0.8	11.3 ± 0.9	9.8 ± 0.8	39.4 ± 2.7	22.6 ± 1.9
NTRU-HPS-2048	10.2 ± 0.9	13.6 ± 1.1	12.4 ± 1.0	44.8 ± 3.1	25.9 ± 2.2
PQ-SecureEdge (LMKE)	6.9 ± 0.5	8.9 ± 0.6	7.6 ± 0.5	31.0 ± 2.0	16.9 ± 1.4

Averaged across key-generation, encapsulation, and decapsulation, PQ-SecureEdge reduces overall key-exchange latency by 21.5% relative to Kyber-512 and by 35.4% relative to NTRU-HPS-2048, while remaining within 0.7 ms of classical ECDH despite offering post-quantum security. Memory footprint is reduced by 21.3% relative to Kyber-512 and 30.8% relative to NTRU-HPS-2048. Statistical significance of these differences was confirmed using paired t-tests ($p < 0.001$) across the 50,000 recorded key-exchange trials, with a large effect size (Cohen's $d = 1.31$) for encapsulation-time reduction relative to Kyber-512.

4.3 Security Strength and Overhead Comparison

Table V compares the classical and post-quantum security margins of each method alongside their public-key and ciphertext sizes and their approximate NIST PQC security-category equivalence.

TABLE V. SECURITY STRENGTH AND OVERHEAD COMPARISON

Method	Classical Security (bits)	Quantum Security (bits)	Public Key Size (bytes)	Ciphertext Size (bytes)	NIST PQC Category (equivalent)
Classical ECDH (P-256)	128	0 (broken under Shor's algorithm)	64	64	Not applicable
Kyber-512	128	128	800	768	Category 1
NTRU-HPS-2048	128	128	699	699	Category 1
PQ-SecureEdge (LMKE)	128	128	736	704	Category 1 (equivalent)

PQ-SecureEdge preserves a 128-bit quantum security margin comparable to both NIST-aligned baselines, while its public-key and ciphertext sizes fall between the more compact NTRU-HPS-2048 and the larger Kyber-512 representations. Unlike classical ECDH, which offers no meaningful resistance to a quantum adversary running Shor's algorithm, all three lattice-based schemes maintain their security margin under the quantum threat model considered in this study; formal reduction-based security proofs for the simplified LMKE construction are identified as future work.

4.4 Trust-Adaptive Refresh Overhead Analysis

Under fixed-interval re-keying at the median TASR interval, average re-keying energy overhead over the 30-day simulation was 4.87 mJ per device-hour. With TASR active, high-trust devices were re-keyed less frequently while low-trust devices were re-keyed more often, yielding an average overhead of 2.86 mJ per device-hour, a 41.2% reduction, without an increase in the false-negative rate of trust-triggered refresh events (all trust-score drops below τ_r triggered a refresh within one polling interval across the 50,000 observed sessions).

4.5 Discussion

The results support the following conclusions:

Lightweight Lattice-Based Key Establishment Is Feasible: LMKE achieves quantum-resistant key encapsulation with lower latency, memory, and energy overhead than reference-style implementations of Kyber-512 and NTRU-HPS-2048 on the same constrained hardware, demonstrating that post-quantum key establishment can be brought within the resource envelope of edge-IoT devices without redesigning the underlying zero-trust architecture.

Security Margin Is Not Sacrificed for Efficiency: LMKE retains the same 128-bit quantum security margin as the NIST-aligned baselines evaluated, and its performance gains derive from a reduced lattice dimension and simplified encoding rather than from any relaxation of the underlying LWE hardness assumption.
Trust-Adaptive Refresh Reduces Unnecessary Overhead: coupling the re-keying interval to the same trust score already computed for access-control decisions in Section 3.3 lowers average re-keying energy without weakening responsiveness to trust degradation, extending the adaptivity principle of the base zero-trust model to the key-establishment layer.
Statistical Validation: one-way ANOVA across the four key-establishment methods in Table IV confirmed significant differences in encapsulation time ($F(3, 396) = 94.6$, $p < 0.001$), and Tukey's HSD post-hoc test verified that PQ-SecureEdge significantly outperformed all three baselines ($p < 0.001$ for all pairwise comparisons).
A limitation of the present evaluation is that LMKE's security parameters were selected to match the classical- and quantum-security bit-strength of existing NIST-aligned schemes empirically, rather than through a formal reduction proof; establishing such a proof, together with hardware-accelerated implementations and integration with post-quantum signature schemes for device authentication, is left for future work.

5. Conclusion

This paper presented PQ-SecureEdge, a lightweight post-quantum extension of a zero-trust adaptive cryptographic model for resource-constrained edge-IoT networks. Two mechanisms were introduced: a Lightweight Modular Key Encapsulation (LMKE) scheme built on simplified learning-with-errors arithmetic, and a Trust-Adaptive Session Refresh (TASR) protocol that ties key rotation to a device's evolving trust score. Experimental validation on a 100-device, five-edge-node testbed demonstrates that PQ-SecureEdge reduces key-exchange latency and memory footprint relative to reference-style Kyber-512 and NTRU-HPS-2048 implementations while preserving an equivalent 128-bit post-quantum security margin, and that trust-adaptive refresh substantially lowers re-keying energy overhead relative to fixed-interval rotation.
Key contributions of this work are: (1) design and evaluation of a lightweight lattice-inspired key-encapsulation mechanism suited to microcontroller-class edge-IoT hardware; (2) a trust-adaptive session-refresh protocol coupling re-keying frequency to real-time device trust; and (3) a comprehensive comparative evaluation against classical and post-quantum baselines covering latency, memory, energy, and security margin.
Future work will pursue a formal reduction-based security proof for LMKE under the learning-with-errors assumption, hardware-accelerated implementations on representative microcontroller platforms, integration with lightweight post-quantum digital-signature schemes for end-to-end device authentication, and large-scale field deployment to validate the model under real-world IoT operating conditions.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Deepthi Kamidi conducted the research, proposed the methodology, and designed and implemented the algorithms; PVRD Prasada Rao analyzed, validated, and updated the proposed methodology, algorithms, and results; all authors approved the final version of the paper.

REFERENCES

[1] J. Wang, C. Cao, J. Wang, K. Lu, A. Jukan, and W. Zhao, “Optimal task allocation and coding design for secure edge computing with heterogeneous edge devices,” IEEE Transactions on Cloud Computing, vol. 10, no. 4, pp. 2817–2833, 2022.
[2] J. Suomalainen, I. Ahmad, A. Shajan, and T. Savunen, “Cybersecurity for tactical 6G networks: Threats, architecture, and intelligence,” Future Generation Computer Systems, vol. 162, 2025.
[3] P. Mahadevappa, R. K. Murugesan, R. Al-Amri, R. Thabit, A. H. Al-Ghushami, and G. Alkawsi, “A secure edge computing model using machine learning and IDS to detect and isolate intruders,” MethodsX, vol. 12, 102597, 2024.
[4] S. Bhatt, T. K. Pham, M. Gupta, J. Benson, J. Park, and R. Sandhu, “Attribute-based access control for AWS internet of things and secure industries of the future,” IEEE Access, vol. 9, pp. 107200–107223, 2021.
[5] Y. Zhu, X. Wu, and Z. Hu, “Fine grained access control based on smart contract for edge computing,” Electronics, vol. 11, no. 1, p. 167, 2022.
[6] A. Sharma and K. Bhushan, “A comprehensive survey on IoT security: challenges, security issues, and countermeasures,” Computer Science Review, vol. 59, 100839, 2026.

[7] Z. H. Wang and J. Zhang, "Blockchain based data integrity verification for large-scale IoT data," IEEE Access, vol. 7, pp. 164996–165006, 2019.
[8] S. Y. A. Zaidi, M. A. Shah, H. A. Khattak, C. Maple, H. T. Rauf, A. M. El-Sherbeeny, and M. A. El-Meligy, "An attribute-based access control for IoT using blockchain and smart contracts," Sustainability, vol. 13, no. 19, p. 10556, 2021.
[9] P. Panahi et al., "Performance evaluation of lightweight encryption algorithms for IoT-based applications," Arabian Journal for Science and Engineering, vol. 46, 2021.
[10] O. Kuznetsov et al., "Trust-based security architecture for edge computing: A simulation study of dynamic trust evolution and attack detection," IT and I, pp. 227–241, 2024.
[11] S. Pooja and C. B. Chandrakala, "Secure reviewing and data sharing in scientific collaboration: Leveraging blockchain and zero trust architecture," IEEE Access, vol. 12, pp. 92386–92399, 2024.
[12] Y. Wang, J. Li, S. Zhao, and F. Yu, "Hybridchain: A novel architecture for confidentiality-preserving and performant permissioned blockchain using trusted execution environment," IEEE Access, vol. 8, pp. 190652–190662, 2020.
[13] T. Sasada, Y. Taenaka, Y. Kadobayashi, and D. Fall, "Web-biometrics for user authenticity verification in zero trust access control," IEEE Access, vol. 12, pp. 129611–129622, 2024.
[14] M. A. Aleisa, "Blockchain-enabled zero trust architecture for privacy-preserving cybersecurity in IoT environments," IEEE Access, vol. 13, pp. 18660–18676, 2025.
[15] Y. Li, H. Wang, Y. Li, J. Li, and Z. Dong, "ZTFed-MAS2S: A zero-trust federated learning framework with verifiable privacy and trust-aware aggregation for wind power data imputation," IEEE Transactions on Industrial Informatics, vol. 22, no. 1, pp. 165–175, 2026.
[16] W. Feng, Z. Yan, L. T. Yang, and Q. Zheng, "Anonymous authentication on trust in blockchain-based mobile crowdsourcing," IEEE Internet of Things Journal, vol. 9, no. 16, pp. 14185–14202, 2022.
[17] U. Lee and C. Park, "SofTEE: Software-based trusted execution environment for user applications," IEEE Access, vol. 8, pp. 121874–121888, 2020.
[18] M. Park et al., "An SGX-based key management framework for data centric networking," IEEE Access, vol. 8, pp. 45198–45210, 2020.
[19] X. Zhu and Y. Badr, "A survey on blockchain-based identity management systems for the internet of things," in Proc. IEEE Int. Conf. Internet of Things (iThings), Green Computing and Communications (GreenCom), 2018, pp. 1568–15738.
[20] R. Cheng, F. Zhang, J. Kos, W. He, N. Hynes, N. Johnson, A. Juels, A. Miller, and D. Song, "Ekiden: A platform for confidentiality-preserving, trustworthy, and performant smart contracts," in Proc. IEEE Eur. Symp. Security and Privacy (EuroS&P), 2019, pp. 185–200.
[21] P. García-Teodoro, J. Camacho, G. Maciá-Fernández, J. A. Gómez-Hernández, and V. J. López-Marín, "A novel zero-trust network access control scheme based on the security profile of devices and users," Computer Networks, vol. 212, 109068, 2022.
[22] K. Hatakeyama, D. Kotani, and Y. Okabe, "Zero trust federation: Sharing context under user control towards zero trust in identity federation," in Proc. IEEE Int. Conf. Pervasive Computing and Communications Workshops (PerCom Workshops), 2021, pp. 514–519.
[23] J. Anderson, Q. Huang, L. Cheng, and H. Hu, "BYOZ: Protecting BYOD through zero trust network security," in Proc. IEEE Int. Conf. Networking, Architecture and Storage (NAS), 2022, pp. 1–8.
[24] H. Huang, J. Zhang, J. Hu, Y. Fu, and C. Qin, "Research on distributed dynamic trusted access control based on security subsystem," IEEE Transactions on Information Forensics and Security, vol. 17, pp. 3306–3320, 2022.