Journal of Information Technology Management
ISSN-OnLine: 2980-7972
University of Tehran

A Trust-Aware and Fault-Tolerant Framework for Secure Data Sharing in Cloud–Edge Environments

Kalpana Dwivedi

Ph.D. Candidate, Department of Computer Science and Engineering, SSCSE, Sharda University, Greater Noida, Uttar Pradesh 201310. E-mail: 2020513259.kalpana@dr.sharda.ac.in

Gaurav Raj*

*Corresponding author, Associate Prof., Department of Computer Science and Engineering, SSCSE, Sharda University, Greater Noida, Uttar Pradesh 201310. E-mail: gaurav.raj@sharda.ac.in

Journal of Information Technology Management, 2026, Vol. 18, Issue 4, pp. 1-17.
Published by the University of Tehran, College of Management
doi: https://doi.org/10.22059/jitm.2026.108912
Article Type: Research Paper

Received: June 21, 2026
Received in revised form: July 28, 2026
Accepted: August 14, 2026
Published online: October 01, 2026

Abstract

The accelerated growth of cloud edge computing has increased the need for secure, reliable, and efficient data exchange in distributed and heterogeneous environments. However, node failures, untrustworthy edge devices, information corruption, key destruction, and excessive communication overhead are significant issues that significantly affect the performance and security of the system. To address these challenges, this paper proposes a trust-aware, fault-tolerant, and secure data-sharing framework for cloud–edge environments. The framework combines the use of dynamic trust evaluation, lightweight cryptography, and erasure codes for data distribution to ensure data confidentiality, integrity, availability, and trustworthiness. By a behavior-based trust model, the edge nodes are continually evaluated on parameters such as reliability, history of previous interactions, and consistency of services, allowing for intelligent decisions to be made regarding data placement and recovery. Erasure coding allows for reconstructing data against multiple node failures at a low storage overhead cost, which enables greater fault tolerance. Within the same framework, secure key management and integrity checks are provided against eavesdropping, replay attacks, insider attacks, and malicious node actions. Simulation results demonstrate that the proposed framework significantly outperforms conventional cloud–edge data-sharing solutions in terms of data availability, fault tolerance, latency, and bandwidth efficiency. The results demonstrate the suitability of the proposed framework for large-scale and security-sensitive applications, such as IoT data aggregation, healthcare, and enterprise cloud computing.

Keywords: Cloud–Edge Computing, Secure Data Sharing, Trust Management, Fault Tolerance, Erasure Coding, Data Security

Introduction

The key architectural concept in hosting latency-sensitive, data-intensive, and large-scale applications, such as the Internet of Things (IoT), novel healthcare, intelligent transportation, and industrial automation, has been identified as the concept of cloud-edge computing. Edge computing minimizes the latency of communication and the consumption of bandwidth, while also increasing the real-time response by minimizing the distance between data sources and data processors. However, this secure and reliable distribution of information between the cloud and edge nodes remains a significant challenge due to the heterogeneous, distributed, and dynamic nature of the edge nodes. The cloud-edge scenarios are characterized by the frequent use of other intermediate nodes, and thus result in vulnerability to security risks, including unauthorized access, eavesdropping, replay attacks and malicious behavior of nodes Compared to a standard cloud system, edge nodes may be placed in untrusted or semi-trusted locations with minimal computation and storage capacity; thus, conventional centralized security systems cannot be applied (Ganesan & Umadevi, 2024).

This has led to trust management being seen as a vital component in the endeavor to create a secure collaboration among distributed entities. Models of trust evaluation dynamics are now being applied to determine the reliability of nodes based on previous behavior, the success of communication, and the consistency of the services (Gupta et al., 2022).

Fault tolerance is another crucial issue when it comes to sharing data on cloud edge networks. Edge nodes are also vulnerable to intermittent connectivity, energy limitations, mobility, and targeted attacks, which can cause data loss and disrupt certain services (Kanchanadevi et al., 2020). The current fault-tolerant methods, such as data replication, have high storage and communication overheads, thereby restricting scalability (Mercy & Srikanth, 2014). According to recent research, there is a focus on the importance of erasure code techniques in achieving fault tolerance with lower redundancy and high data availability (Gupta et al., 2022). Nevertheless, there is a lack of research on combining erasure coding and trust-aware security (Khan et al., 2022). More recent studies have proposed blockchain and decentralized security frameworks to enhance the transparency and trustworthiness of cloud–edge systems (Li & Wang, 2023; Verma & Nand, 2025).

Literature Review

However, the decentralized character of edge and fog environments introduces additional security risks, as edge nodes can be compromised and act maliciously (Zhou & Shi, 2025). The current literature tends to focus on encryption and authentication without considering the trustworthiness or reliability of the involved nodes (Sugino et al., 2018). It is due to this that trust management has become a critical subject of study, providing support for security schemes in distributed systems. Trust-based models consider the behavior of nodes based on the measures of communication success rate, service availability, and previous interactions to

make a secure cooperation decision (Hörandner et al., 2020). Probabilistic trust models and reputation-based trust structures have been suggested to identify malicious or selfish nodes in distributed networks (Karame et al., 2017). Even though these methods enhance the accuracy of decision-making, most of them are developed without secure data storage and sharing mechanisms, which constrain their applicability in the end-to-end cloud-to-edge data sharing context (Qin et al., 2015).

Another important factor of safe data sharing in a distributed environment is fault tolerance. Thus, classical fault-tolerant storage methods are based on storing data in a replica, which guarantees its availability; however, this approach incurs a communication overhead (Li et al., 2013). To eliminate such constraints, data distribution schemes based on erasure coding have been suggested that can provide better storage performance and node failure resilience (Samuel & Kasturi, 2024).

Recent investigations have shown that erasure coding can significantly enhance the availability of data in cloud and edge storage systems, while also reducing redundancy (Zheng et al., 2023). However, most available fault-tolerant systems lack the element of trust evaluation, which means that unreliable or malicious nodes can engage in data storage and recovery techniques (Velmurugan et al., 2024). Although secure data sharing, trust management, and fault tolerance have been improved, the current solutions mainly focus on these issues individually. Frameworks built on blockchains aim to combine trust and security by offering decentralized verification and immutability (Pasquier et al., 2017). Still, they often incur high computational and communication overheads, making them unsuitable for large-scale and resource-constrained edge cases. Hence, there is an urgent research gap regarding the development of a unified system that integrates trust-conscious decision-making, fault-tolerant data allocation, and minimal security applications across cloud-edge conditions. This loophole motivates the proposed trustworthy, fault-tolerant, and secure data sharing framework (Li & Wang, 2023).

Methodology

Cloud-Edge System Architecture

The discussed system follows a hierarchical cloud-edge architecture that incorporates three primary layers: data sources, edge layer, and cloud layer (Attasena et al., 2017). The data sources comprise IoT devices, sensors, and user applications that continuously generate large amounts of data. The edge layer consists of several distributed edge nodes, which have limited computation, storage, and energy resources and are placed near data sources to allow low-latency processing and some local data aggregation (Grolinger et al., 2013).

The cloud layer is a capability that offers operational capacities such as robust computation, long-term storage, and global coordination. Information created in the source layer is encoded, encrypted, and shared with selected edge nodes, whereas the cloud handles trust, metadata, and recovery (Kumar et al., 2018).

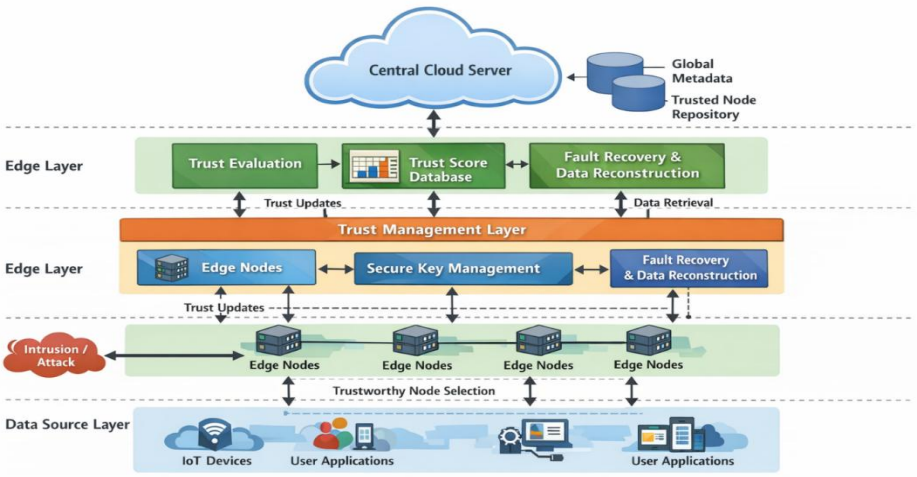

Figure 1. Architecture of the proposed trust-aware fault-tolerant secure data sharing framework for cloud–edge environments

Network and Communication Model

The network model assumes heterogeneous communication connections between data sources, edge nodes, and the cloud, utilizing both wired and wireless connections. The communication between edge nodes and the cloud is carried out using secure channels, whereas communication between edge nodes can vary in latency and have intermittent connectivity. The model accounts for dynamic network states, where edge nodes can be added or removed from the system due to mobility or failures. Some of the message exchanges involve trust updates, encrypted data fragments, integrity verification messages, and recovery requests (Rastogi et al., 2017).

Trust Model Description

A dynamically applied trust model is used to determine the trustworthiness of edge nodes involved in data storage and sharing. The trust score is allocated to each edge node depending on the history of interaction, the fact of data forwarding, response, and integrity checking results. The trust scores are updated periodically in accordance with changes in node behavior. During data storage and recovery operations, highly trusted nodes are preferentially selected for use. Conversely, those nodes with a low trust score are excluded, as they increase security and reliability threats (Wang et al., 2010).

Threat and Adversary Model

The system considers external and internal adversaries. When transmitting data, external attackers can eavesdrop, launch replay attacks, or alter messages. Internal adversaries include compromised or malicious edge nodes that may drop data fragments, distort stored content, or provide incorrect answers during recovery. It is assumed that the attacker has limited computational resources and can break standard cryptographic primitives but may also exploit system or unreliable node vulnerabilities.

The assumption behind the model is that the cloud server is a totally trusted server with adequate computing resources and storage space. Edge nodes may be considered semi-trusted, though this may lead to unreliable or malicious behavior. Light cryptographic functions are preferable because of the resource constraints of the edge. The assumption is that time synchronization among nodes is loosely maintained, and a secure key distribution system is provided at the system's inception (Yu et al., 2010).

Design Constraints and Assumptions

The assumption behind the model is that the cloud server is a totally trusted server with adequate computing resources and storage space. Edge nodes may be considered semi-trusted, though this may lead to unreliable or malicious behavior. Light cryptographic functions are preferable because of the resource constraints of the edge. The assumption is that time synchronization among nodes is loosely maintained, and a secure key distribution system is provided at the system's inception (Yu et al., 2010).

Problem Definition

With the range of edge nodes having different trust levels and failure probabilities, it becomes a challenge to form a secure data sharing scheme that guarantees data confidentiality, integrity, and availability with the lowest communication overhead, recovery latency, and other related characteristics. The system must also be smart enough to identify nodes for the distribution and sharing of data reliably, remain resilient to partial node failure, and recover reliable data. The goal is to ensure the highest reliability and security of the system, regardless of varying cloud-edge conditions, without compromising scalability and efficiency (Ferreira et al., 2017).

Trust-Aware Secure Data Sharing Framework

Framework Overview

The proposed model of trust-based secure data sharing aims to ensure the confidentiality, integrity, and availability of data in the cloud-edge network, which can also withstand server failure and malicious control. The model is a synthesis of four key components: dynamic trust

assessment, lightweight encryption, key management for security, data encryption using erasure codes, and trust-sensitive information distribution. Data generated on the source layer is coded and encrypted using an algorithm that breaks it down into several fragments. The fragments are handed around smartly (using calculated trust scores) with the chosen edge nodes. A cloud layer containing global metadata performs trust and data recovery operations, enabling the effective and secure sharing of data in dynamic situations.

System for Trust Assessment and Revision.

To assess the reliability of edge nodes involved in sharing and storing data, a dynamic trust evaluation system is considered. A baseline trust score is fed to all edge nodes at the system's inception. The trust scores are updated periodically, depending on the perceived behavioral attributes of the node, such as success in forwarding data, promptness of response, data integrity results, and dependability in recovery requests. Nodes whose behavior is similar and stable are rated higher on the trust aspect, whereas those that are malicious or unreliable are punished. Such an updated adaptive trust process guarantees the reflective accuracy of trust assessments concerning the current node's behavior.

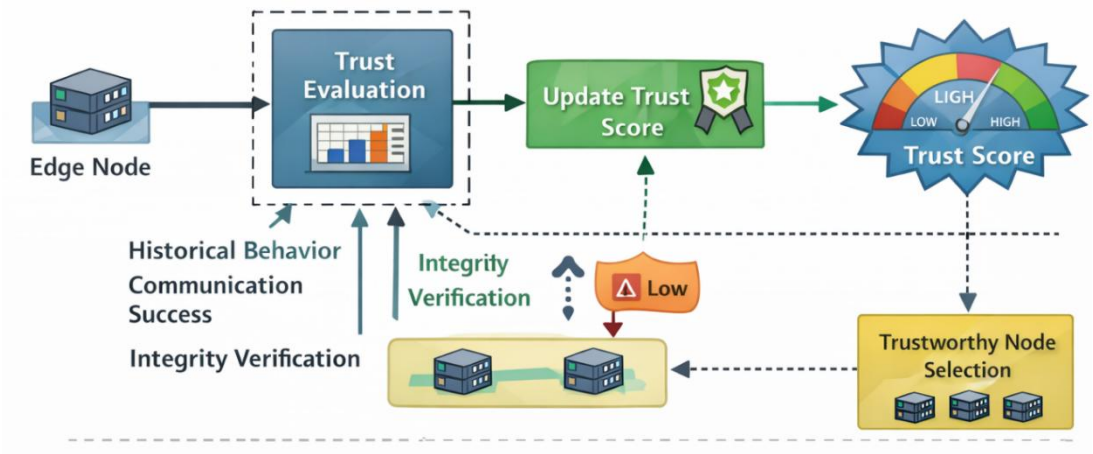

Figure 2. Trust evaluation and update mechanism for edge nodes in the proposed framework

Authorized Data Encryption and Secrecy

Lightweight encryption is used to ensure that data is protected during its distribution. Data content is encrypted symmetrically because it is efficient, whereas the exchange of keys is done asymmetrically. The cloud layer features a secure key management mechanism that generates, distributes, and periodically refreshes cryptographic keys. Only trusted edge nodes are shared keys with, and this minimizes the probability of key compromise. This will provide safe access to data while minimizing computational costs on resource-constrained edge devices.

Data Encoding Strategy using Erasure Codes

This is done to increase fault tolerance by encoding the encrypted data using an erasure coding method, which divides the information into several fragments. It is coded to give an original data object. n fragments, of which any k fragments can be used to reconstruct the original data. This strategy can recover the data despite the failure of a section of the edge nodes or their unavailability. Erasure coding is an efficient storage method, as it incurs low storage overhead compared to traditional replication with high data availability.

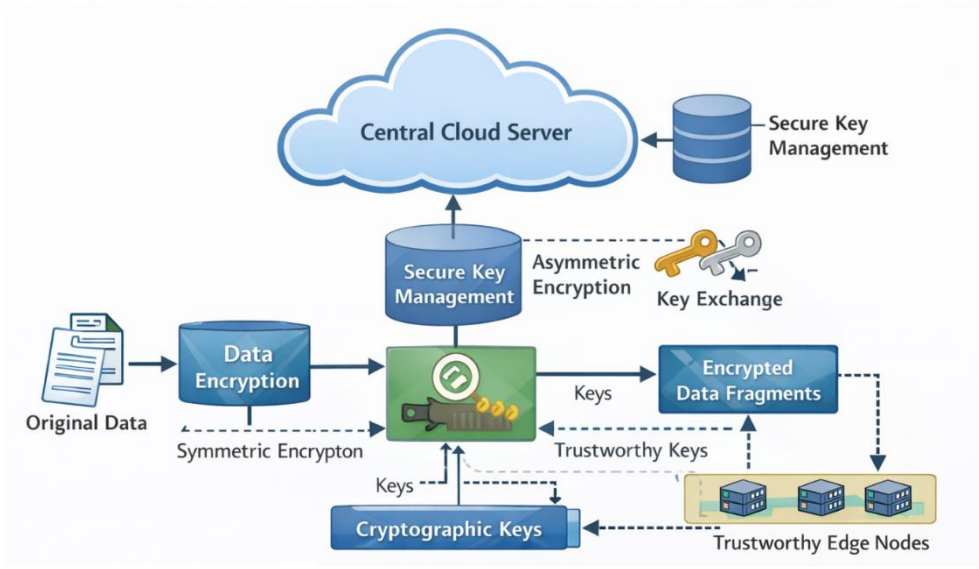

Figure 3. Secure data encryption and key management process in the proposed framework

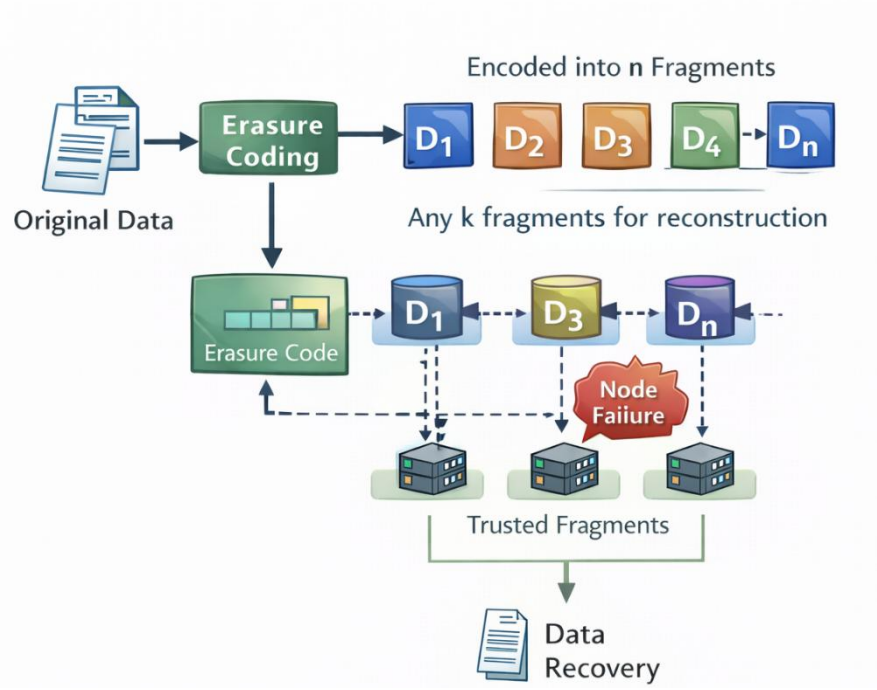

Figure 4. Erasure-coded data encoding and recovery strategy for fault-tolerant storage

Trust-Sensitive Data Distribution and Placement

The packets created by erasure coding are assigned to the edge nodes based on their trust scores. The data is placed on the nodes that have trust scores over a set threshold. The presence of the trust-aware placement strategy ensures that sensitive data is stored on stable nodes, minimizing the risk of data loss or compromise. Issues of load balancing are also taken into consideration to prevent overloading of highly trusted nodes and to provide system scalability.

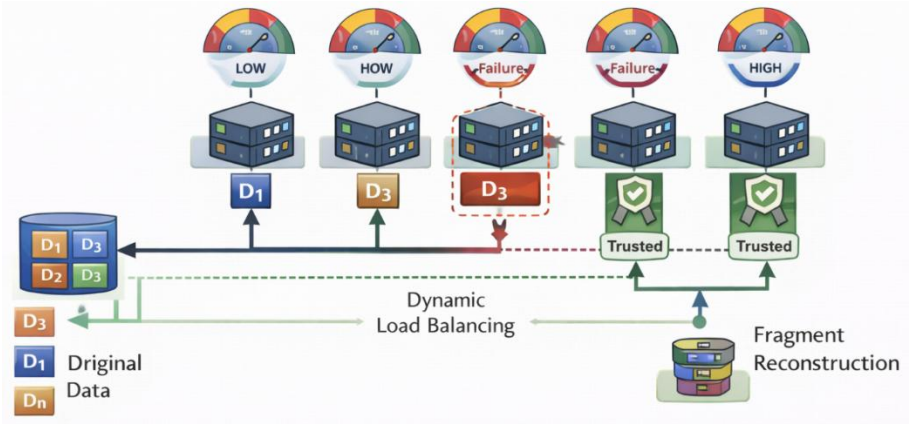

Figure 5. Trust-aware data distribution and placement across edge nodes

Checking Integrity and Mitigation of Attacks

To ensure the integrity of the data, cryptographic hash functions and message authentication codes are used to verify the stored and relayed pieces of data. Integrity checks are performed during the retrieval or recovery of data to identify any unauthorized changes. The system is effective in preventing the most frequent attacks, which include eavesdropping attacks, replay attacks, malicious node attacks, and node selection and integrity checks based on trust. Nodes that fail multiple integrity checks are tagged as malicious and will not participate in any further data sharing processes.

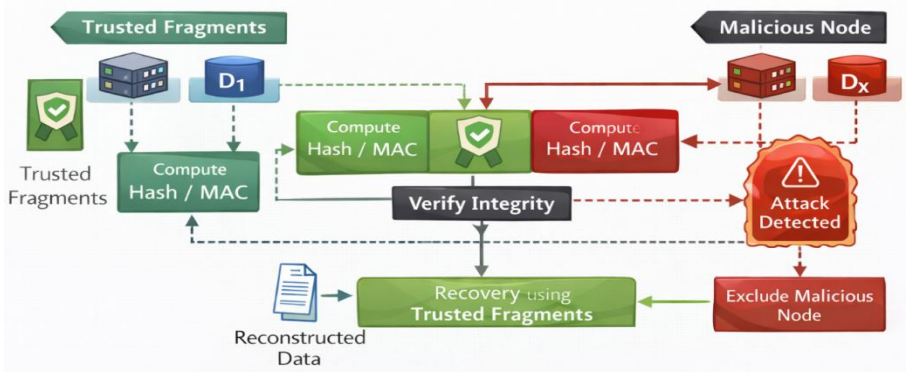

Figure 6. Integrity Check & Attack Mitigation in the proposed framework

Mathematical Model and Formulation

This document provides the mathematical foundation for the proposed trust-sensitive, fault-tolerant, and secure data sharing model. The formulation incorporates trust computation, erasure storage, security guarantee, data availability, and communication overhead.

Trust Score Computation Model

Let $E = \{e_1, e_2, \dots, e_N\}$ denote the set of edge nodes. Each edge node ei is associated with a dynamic trust score Ti (t) in [0,1] and is updated over time according to its observed behavior. The trust score is calculated as a weighted sum of several factors:$T_i(t) = \alpha B_i(t) + \beta C_i(t) + \gamma I_i(t)$
where:

- $B_i(t)$denotes historical behavior reliability,
- $C_i(t)$represents communication success ratio,
- $I_i(t)$reflects data integrity verification outcomes,
- $\alpha + \beta + \gamma = 1, \alpha, \beta, \gamma \in [0,1]$.

To be adaptable, trust scores are updated by means of exponential smoothing:

$$T_i(t+1) = \lambda T_i(t) + (1-\lambda)\hat{T}_i(t) \quad (1)$$

where $\hat{T}_i(t)$is the newly observed trust value and λcontrols sensitivity to recent behavior.

Erasure Coding Formulation

Let a data object Dbe divided into $k$original blocks and encoded into $n$fragments using an (k)erasure coding scheme, where $n > k$.

$$EC(D) = \{d_1, d_2, \dots, d_n\} \quad (2)$$

The original data can be reconstructed from any subset of $k$fragments:

$$D = EC^{-1}(\{d_i \mid i \in S, \mid S \mid \geq k\}) \quad (3)$$

This formulation provides fault tolerance, allowing data recovery even if up to n-k edge nodes fail or become unavailable.

Security and Confidentiality Model

To ensure confidentiality, the data object $D$was encrypted using symmetric encryption before the encoding process:$C = E_{K_s}(D)$.

where $K_s$is a symmetric session key. The session key is protected using asymmetric encryption:

$$K_s = E_{K_p}(K_s) \quad (4)$$

Integrity is then guaranteed by calculating a message authentication code (MAC) for each fragment:

$$MAC_i = H(d_i \parallel K_s) \quad (5)$$

During data retrieval, integrity is verified by recomputing the MAC values and comparing them with the previously calculated values, thereby countering tampering and replay attacks.

Data Availability and Reliability Analysis

Let $p_i$denote the availability probability of an edge node e_i. The probability that at least $k$fragments are accessible is given by:

$$P_{avail} = \sum_{j=k}^{n} \binom{n}{j} \bar{p}^{j}(1-\bar{p})^{n-j} \quad (6)$$

where $\bar{p}$ is the average availability of trusted nodes.

By placing high-trust nodes first, the proposed framework maximizes $\bar{p}$, thus improving overall data availability and system reliability.

Communication and Computation Cost Model

The total communication cost $C_{comm}$is expressed as:

$$C_{comm} = n \cdot | d_i | + C_{meta} \quad (7)$$

where $| d_i |$ is the fragment size and C_{meta}represents trust and key management overhead. The computational cost C_{comp}includes encryption, encoding, and verification operations:

$$C_{comp} = C_{enc} + C_{EC} + C_{ver} \quad (8)$$

Although the extra computation of erasure coding introduces some overhead, the framework minimizes retransmissions and recovery delays, resulting in less end-to-end latency during node failure conditions.

Experimental Setup and Performance Evaluation

An experimental analysis of the given trust-based fault-tolerant secure data sharing framework is conducted within a discrete-event simulation environment designed in Python, allowing for realistic modelling of interactions between the cloud and the edge. The simulated architecture will consist of a centralized cloud server and several distributed edge nodes, with the number of edge nodes ranging from 20 to 100 to assess scalability. Network latency, bandwidth limits, and communication delays are simulated to capture realistic conditions at cloud edges. The simulations are provided with node failures and malicious behavior that is randomly introduced to determine fault tolerance and security resilience. Artificial datasets are generated to simulate workloads of cloud-edge data sharing, including structured and semi-structured packets of data ranging from 128 KB to 5 MB in size, which replicate IoT sensor data, healthcare data, and enterprise data logs.

Table 2. Setting parameters and configuring the simulation

Parameter	Value
Number of edge nodes	20–100
Data size	128 KB–5 MB
Erasure coding (n, k)	(10, 6)
Trust threshold	0.6
Encryption	AES-128
Simulation time	1000 s

Moreover, behavioral information, including the success rate of communication, response time, and integrity check systems, is used to help determine dynamic trust scores. The data availability ratio, the rate of data loss, the average data distribution latency, the data recovery time, the trust accuracy, and the rate of communication overhead are the key measurements of the effectiveness of the proposed framework. The solution is compared to the representative baseline techniques, including traditional cloud-based secure storage, where no one knows about trust or is fault-tolerant, edge data sharing with lightweight encryption, replication-based fault-tolerant storage, erasure-coded storage where no one knows whether it is trust-based or not, and trust-based access control systems which do not have built-in fault-tolerance. Several experimental settings are established by varying the node failure rates, the ratio of malicious nodes, the network size, and the amount of data to analyze the resilience and scalability of the systems. The results show that the proposed framework is superior to the included base strategies, as it offers an exceptional view of data and lower data loss rates due to the choice of reliable nodes and the adoption of erasure coding, which allows for the successful recovery of data in case of partial node failures. Furthermore, the proposed framework offers lower data distribution latency and recovery time compared to replication-based solutions, while maintaining moderate communication overhead, which justifies its efficacy and relevance to large-scale and security-aware cloud-edge systems.

Table 3. Performance comparison of proposed framework with baseline methods

Method	Availability (%)	Loss (%)	Latency (ms)	Recovery Time (ms)
Proposed	98.6	1.2	210	180
Replication	95.1	3.4	350	260
Erasure (No Trust)	96.0	2.8	290	230
Edge Encryption	91.7	5.9	240	N/A

Results and Discussion

Here, the results of the experiment are presented, and the work of the proposed trust-aware fault-tolerant secure data sharing framework is evaluated in terms of its security, fault tolerance, efficiency, trust sensitivity, and scalability.

Data Availability Analysis and Fault Tolerance

Fault tolerance is a crucial requirement in cloud-edge systems due to the high frequency of node failures and network instability. The results show that the proposed framework has a significantly better data availability ratio compared to replication-based and non-trust-aware erasure coding methods. Erasure coding is also implemented to provide data reconstruction in the event of disk failures of multiple edge nodes and to provide trust-wise data placement guarantees, i.e., data fragments will be stored at the discretion of trustworthy nodes. As such, the proposed structure has minimized data loss rates and demonstrated its ability to withstand growing node failure conditions, which is a testament to its success in maintaining data availability under dynamic conditions.

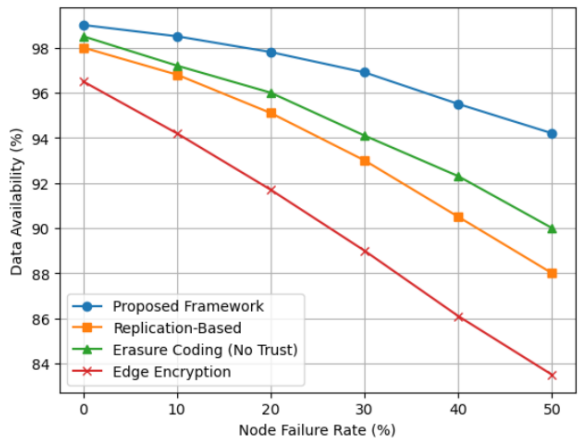

Figure 7. Data availability comparison under varying node failure rates

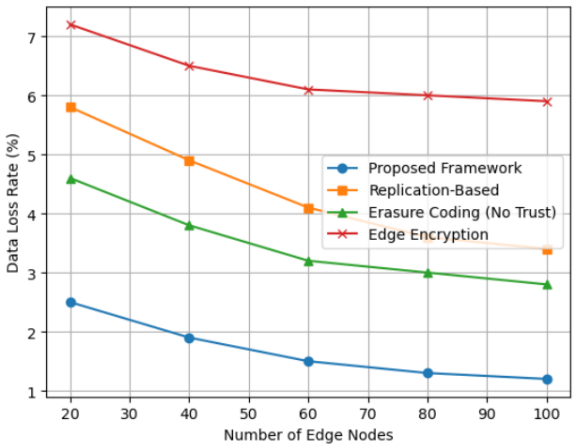

Figure 8. Data loss rate versus number of edge nodes

Latency and Communication Overhead Analysis

The latency analysis shows that the proposed framework has reduced data distribution and recovery latency compared to replication-based fault-tolerant systems. This has been enhanced primarily by minimizing redundancy and strategically positioning fragments based on trust scores. Although the framework will add more overhead to the evaluation of trust and integrity checking, the overall communication overhead is moderate and significantly lower than that of replication-based methods.

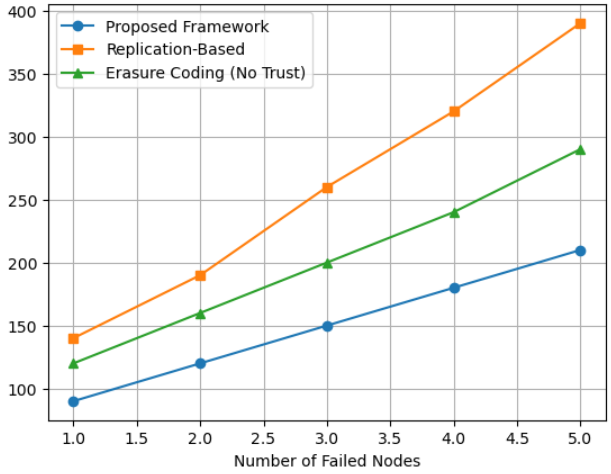

Figure 9. Data recovery time under varying numbers of failed nodes

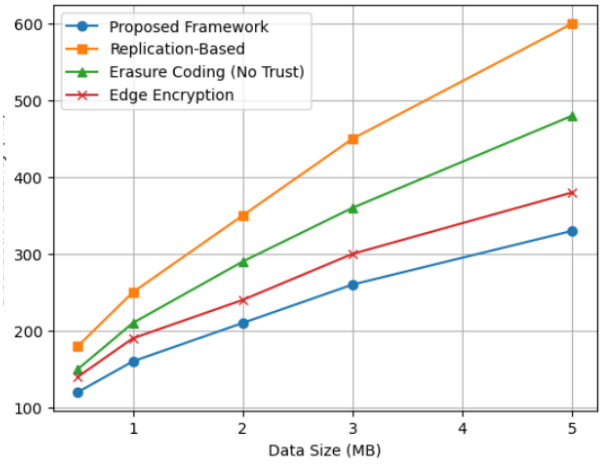

Figure 10. Average data distribution latency for different data sizes

The experimental results of the findings show that latency and availability degradation are minimal as the scale increases from small to large systems. The obtained results confirm that the suggested framework is suitable for a large-scale cloud-edge system that needs to support various and data-intensive applications.

Security and Trust Analysis

The proposed secure data sharing model, based on trust, offers comprehensive security protection against a wide range of security threats in cloud-edge settings, encompassing trust evaluation, cryptographic security, and fault-tolerant information storage procedures. Secure encryption ensures data confidentiality and provides a key distribution mechanism under the control of the authors, allowing them to prevent unauthorized access even in the event of a compromised edge node. Cryptographic hashing and fragment validation are two integrity checker systems that facilitate the determination of data alteration, replay attacks, and malicious behavior, which can be consistent when storing data as well as during message transmission. The dynamic trust evaluation model is a process in which the behavior of nodes is continuously monitored in terms of reliability, interaction track record, and consistency of service, allowing the system to isolate malicious or unreliable nodes and prevent their participation in the data storage and recovery process. The sensitive data location also

minimizes exposure, as sensitive fragments are only placed on highly trusted devices, thereby reducing the attack surface. The scheme is also resistant to insider attacks, collusion attacks, node impersonation, and denial-of-service attacks, as low-trust nodes are eventually reduced and locked out. Besides this, the adaptive trust update mechanism lends the mechanism its resilience in a highly dynamic environment, as it reacts to environmental changes in real-time.

Finally, although the analysis conducted by simulation results shows good performance, a large-scale implementation in the real world should be carried out to test the trust dynamics over the long term, its interoperability with existing cloud platforms, and its cost implications on a large scale. These practical limitations should be addressed to achieve the goal of successfully implementing real-world cloud-edge systems.

IT Management Implications

The suggested trust-aware fault-tolerant secure data-sharing framework offers some practical implications for managing information technology in the modern cloud-edge computing environment. The growing adoption of distributed cloud and edge computing applications across enterprises, smart city services, healthcare, and Internet of Things (IoT) applications is creating a significant management challenge to ensure secure, reliable, and efficient data management. The proposed framework allows the trusted IT administrator or cloud service manager to dynamically determine whether the edge node is trusted before assigning it any data storage or computation tasks, minimizing the risk of data leakage, malicious edge nodes, and service disruption. Combining trust-aware decision-making with erasure-coded fault-tolerant storage reduces the amount of data being stored and brings high ability to serve the data and continuity of services, in turn reducing expenses and burdens on infrastructure and operations.

In the operational sense, the proposed idea helps the cloud administrator with the selection of nodes that are highly trusted for critical workloads and the selection of nodes that are not trusted for critical workloads. This intelligent resource management enhances the reliability of the overall system, decreases the recovery time in case of node failures, and eases compliance with the system's security policies or data governance needs. Moreover, its lightweight encryption and speedy trust evaluation mechanisms can reduce computational burden, thus being suitable for the resource-constrained environment of the edge without security compromises. The proposed model is particularly suitable for enterprise decision-makers, aiming at avoiding the trade-off between security, performance, and operational cost for the deployment of scalable and resilient cloud-edge services. Therefore, the framework provides tools to help organizations implement better services, increase business continuity and user confidence in the use of distributed data sharing platforms, and streamline cybersecurity governance.

Conclusion

In this paper, a trust-aware, fault-tolerant, and secure data sharing framework is presented for a cloud-edge environment, addressing the major concerns associated with security, reliability, and trust in a distributed system. The proposed structure ensures the confidentiality, integrity, and availability of data through the application of dynamic trust evaluation, lightweight encryption, erasure-coded data distribution, and verification of data integrity, even in the event of node failure and malicious attacks. The results of the experiments have proved that the framework can create a considerable positive effect on data availability, data loss, and latency and have a superior effect on system reliability, as compared to the current cloud-edge data sharing frameworks. A trust-conscious data placement method proves useful in lessening the danger of security attacks as well as assuring the scalability of a highly dynamic and large environment. Despite these advantages, future work will be aimed at optimizing trust computation by resource-constrained edge devices, generalizing the framework to fully decentralized and blockchain-aided trust management, and testing performance with real-world deployments and heterogeneous workloads. There is also the application of machine learning-predicted trust and adaptive security policies, which is the best next step to achieving robustness and autonomy in the next-generation cloud-edge systems.

Conflicts of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

The author's received no financial support for the research, authorship, and/or publication of this article.

References

Attasena, V., Darmont, J., & Harbi, N. (2017). Secret sharing for cloud data security. *Future Generation Computer Systems, 78*, 365–381.

Ferreira, B., Rodrigues, J., Leitão, J., & Domingos, H. (2017). Practical privacy-preserving content-based retrieval in cloud image repositories. *IEEE Transactions on Cloud Computing, 5*(3), 382–395.

Ganesan, V., & Umadevi, N. (2024). Group key management and sharing protocol in cloud computing environment. In *Proceedings of the 2024 8th International Conference on Inventive Systems and Control (ICISC)*. IEEE.

Grolinger, K., Higashino, W. A., Tiwari, A., & Capretz, M. A. M. (2013). Data management in cloud environments: NoSQL and NewSQL data stores. *Journal of Cloud Computing, 2*(22), 1–24.

Gupta, A. K., Singh, D., Singh, K., & Verma, L. P. (2022). STCP: A novel approach for congestion control in IoT environment. *Journal of Information Technology Management, 14*(Special Issue on

Security and Resource Management Challenges for Internet of Things), 44–51. https://doi.org/10.22059/jitm.2022.85648

Gupta, I., Singh, A. K., Lee, C.-N., & Buyya, R. (2022). Secure data storage and sharing techniques for data protection in cloud environments: A systematic review, analysis, and future directions. *IEEE Access, 10*, 1–1.

Hörandner, F., Ramacher, S., & Roth, S. (2020). Selective end-to-end data-sharing in the cloud. *Journal of Banking and Financial Technology, 4*(2), 139–157.

Kanchanadevi, P., Raja, L., Selvapandian, D., & Dhanapal, R. (2020). An attribute-based encryption scheme with dynamic attributes supporting in the hybrid cloud. In *Proceedings of the 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)*. IEEE.

Karame, G. O., Soriente, C., Lichota, K., & Capkun, S. (2017). Securing cloud data under key exposure. *IEEE Transactions on Cloud Computing, 5*(3), 472–484.

Khan, T., Singh, K., Gupta, S., & Manjul, M. (2022). Multi trust-based secure trust model for WSNs. *Journal of Information Technology Management, 14*(Special Issue on Security and Resource Management Challenges for Internet of Things), 147–158. https://doi.org/10.22059/jitm.2022.86657

Kumar Jangir, M., Singh, K., & Khan, T. (2025). HybridTouch: A robust framework for continuous user authentication by GAN-augmented behavioral biometrics on mobile devices. *Journal of Information Technology Management, 17*(Special Issue on Intelligent Security and Management), 108–129. https://doi.org/10.22059/jitm.2025.102924

Kumar, K. S., Chythanya, K. R., Kumar, N. V., & Siruvoru, V. (2018). An enhanced distributed accountability for data sharing in cloud computing technologies. *International Journal of Engineering and Technology, 7*(1), 1–6.

Li, M., Yu, S., Zheng, Y., Ren, K., & Lou, W. (2013). Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption. *IEEE Transactions on Parallel and Distributed Systems, 24*(1), 131–143.

Li, X., & Wang, M. (2023). Trusted sharing of data under cloud-edge-end collaboration and its formal verification. In *Proceedings of the 2023 IEEE 48th Conference on Local Computer Networks (LCN)*. IEEE.

Mercy, S. S., & Srikanth, G. U. (2014). An efficient data security system for group data sharing in cloud system environment. In *Proceedings of the International Conference on Information Communication and Embedded Systems (ICICES 2014)*. IEEE.

Pasquier, T. F. J.-M., Singh, J., Eyers, D., & Bacon, J. (2017). CamFlow: Managed data-sharing for cloud services. *IEEE Transactions on Cloud Computing, 5*(3), 472–484.

Qin, B., Deng, R. H., Liu, S., & Ma, S. (2015). Attribute-based encryption with efficient verifiable outsourced decryption. *IEEE Transactions on Information Forensics and Security, 10*(7), 1384–1393.

Raj, D., Sagar, A. K., & Ather, D. (2025). Optimizing vertical handover using multi-criteria decision making in heterogeneous networks. *Journal of Information Technology Management, 17*(Special Issue on Intelligent Security and Management), 63–86. https://doi.org/10.22059/jitm.2025.102922

Rastogi, N., Gloria, M. J. K., & Hendler, J. (2017). Security and privacy of performing data analytics in the cloud: A three-way handshake of technology, policy, and management. *Information Systems Management, 34*(4), 331–343.

Samuel, B., & Kasturi, K. (2024). A secure authentication and collaborative data-sharing model based on a blockchain network in the cloud. *Journal of Control and Decision, 11*(4), 730–745.

Sugino, H., Nakano, T., & Sugawara, S. (2018). Efficient content sharing by multiple users with RAID-based multi-cloud storages. In *Proceedings of the 2018 IEEE 7th International Conference on Cloud Networking (CloudNet)*. IEEE.

Tyagi, J., & Sharma, M. (2025). Optimising resource sharing algorithms for efficient cloud computing. In *Proceedings of the 2025 IEEE 11th Conference on Big Data Security on Cloud (BigDataSecurity)*. IEEE.

Velmurugan, S., Prakash, M., Neelakandan, S., & Radhakrishnan, A. (2024). Provably secure data selective sharing scheme with cloud-based decentralized trust management systems. *Journal of Cloud Computing, 13*(1), 1–20.

Verma, M., & Nand, P. (2025). Adaptive differential privacy for protecting user confidential information on Android devices. *Journal of Information Technology Management*, *17*(Special Issue on Intelligent Security and Management), 155–167. https://doi.org/10.22059/jitm.2025.102933

Wang, C., Ren, K., Lou, W., & Li, J. (2010). Toward publicly auditable secure cloud data storage services. *IEEE Network, 24*(4), 19–24.

Wang, X., Zhang, J., Meng, Z., Yi, B., Hu, A., & Gu, M. (2023). Research on symmetric encryption and decryption of multi-user shared data for cloud storage environment. In *Proceedings of the 2023 IEEE 6th International Conference on Information Systems and Computer Aided Education (ICISCAE)*. IEEE.

Yu, S., Wang, C., Ren, K., & Lou, W. (2010). Achieving secure, scalable, and fine-grained data access control in cloud computing. In *Proceedings of IEEE INFOCOM* (pp. 1–9).

Zheng, T., Wu, J., & Li, G. (2023). IoV data sharing scheme based on the hybrid architecture of blockchain and cloud-edge computing. *Journal of Cloud Computing, 12*(99), 1–20.

Zhou, M., & Shi, M. (2025). Construction of a business management education resource sharing platform in the cloud computing environment. In *Proceedings of the 2025 International Conference on Algorithm, Artificial Intelligence and Computer Vision (AAICV)*. IEEE.

Bibliographic information of this paper for citing:

Dwivedi, Kalpana & Raj, Gaurav (2026). A Trust-Aware Fault-Tolerant Secure Data Sharing Framework for Cloud–Edge Environments. *Journal of Information Technology Management*, 18 (4), 1-17. https://doi.org/10.22059/jitm.2026.108912