IoT Privacy Protection Scheme Based on Hyperledger Fabric

Weiyan Feng*, Li Zhang, Xin Zhao
School of Information Engineering, Weifang Engineering Vocational College
Weifang 262500, China
E-mail: kapibala121@163.com
*Corresponding author

Keywords: Hyperledger fabric, internet of things privacy protection, distributed storage, access control, secret sharing mechanism

Received: January 16, 2025

The increasing use of Internet of Things devices has exacerbated the contradiction between data sharing and privacy protection, while existing blockchain-based solutions still have significant limitations in scalability, computational efficiency, and access control. Therefore, this study proposes a Hyperledger Fabric privacy protection framework for the Internet of Things, and constructs a smart contract system that includes device management, policy management, and access control. By combining advanced encryption standard algorithms, interstellar file system storage, and Shamir secret sharing mechanism, secure storage and flexible sharing of data have been achieved. The experiment shows that the throughput of the research scheme is stable between 370 TPS and 420 TPS on both the training and testing sets, with an average delay controlled between 50 ms and 72 ms, which is significantly better than other comparative schemes. The utilization rate of the central processing unit in terms of resource overhead is maintained at around 50%, with a minimum memory consumption of 505 MB. In addition, in terms of decryption performance, the decryption success rate of this scheme under various attacks is between 98% and 99%, with an average decryption cost of 45 ms. The research scheme balances performance and efficiency while ensuring privacy and security, demonstrating its application advantages in Internet of Things privacy protection scenarios.

Povzetek: Raziskava predlaga učinkovito ogrodje za zaščito zasebnosti v IoT z uporabo Hyperledger Fabric, ki omogoča varno deljenje podatkov ter dosega visoko zmogljivost in zanesljivost ob nizkih stroških virov.

1 Introduction

The development of the Internet of Things (IoT) has brought about exponential growth in data scale, and the interconnectivity between devices has driven the rapid evolution of application scenarios such as smart healthcare, smart manufacturing, and smart cities [1]. To achieve privacy protection in the IoT environment, it is necessary to simultaneously address three key issues: secure data storage, flexible access control, and trusted sharing [2]. The traditional centralized storage method is difficult to ensure data integrity and tamper resistance, and relying solely on symmetric or asymmetric encryption can also bring high computational and key management overhead in multi-user scenarios. Many scholars have conducted research on privacy protection in the IoT environment. Among them, Hua et al. analyzed security threats in international trade transactions, evaluated the effectiveness of privacy data protection using additive aggregation functions, and explored relevant core issues. The study also analyzed privacy protection models based on IoT blockchain and proposed corresponding international trade privacy data protection strategies. Among the five interviewed companies, the level of data auditing and backup was higher than the standard requirements [3]. Xie et al. proposed a privacy protection framework for edge face recognition systems. This framework adopted a local differential privacy algorithm based on the proportion difference of feature information to protect privacy in data transmission and training models, and combined identity authentication and hashing technology to ensure the legitimacy of terminal devices and the integrity of facial images. Theoretical analysis showed that this method achieved better privacy protection while ensuring availability [4]. Kong et al. proposed a privacy protection scheme that combines smart contracts, key isolation, and uncertified anonymous signatures to ensure data privacy and maintain user anonymity during data sharing. This scheme utilized smart contracts to achieve fair and automatic key distribution, replacing traditional key generation centers. The security of this scheme has been verified under the random oracle model, and by optimizing the elliptic curve points, the signature length has been shortened, significantly reducing communication overhead [5].

Table 1: Comparative analysis of related blockchain-based IoT privacy protection approaches

Reference	Core Technique	Strengths	Focus Layer	Limitations
Hua et al. [3]	Additive aggregation + IoT blockchain	Improves auditability	Data & policy layer	Limited encryption–performance analysis
Xie et al. [4]	Local differential privacy + authentication	Strong edge privacy	Application / edge layer	Scenario-specific, weak cross-domain support
Kong et al. [5]	Smart contract + key isolation + certificateless signature	Efficient key management	Cryptographic layer	No scalability evaluation
Abbas et al. [7]	Post-quantum cryptography in Fabric	Stronger crypto security	Encryption & certificate layer	Focused only on certificate security
Qingan et al. [8]	National crypto + dual-signature	Reliable identity verification	Identity layer	Limited storage/access analysis
Mahdi et al. [9]	Hyperledger telehealth framework	Practical deployment validation	Application layer	No fine-grained sharing design
Proposed Scheme	AES + IPFS + Shamir + contract access	Balanced privacy & scalability	On/off-chain collaborative architecture	Slight computational overhead

In recent years, blockchain technology has gradually become an important support platform for IoT data management due to its decentralized and traceable characteristics. Hyperledger Fabric, as a permissioned blockchain, can support multi-organization participation and fine-grained access control, making it particularly suitable for privacy protection research in consortium chain environments [6]. Abbas et al. delved into the integration of quantum encryption algorithms in Hyperledger Fabric to address security threats posed by quantum computing. Meanwhile, an improved Cryptogen tool has been developed to generate X.509 certificates that support both classical and post-quantum cryptography. Through empirical analysis using Hyperledger Caliper and Prometheus, this hybrid method effectively improved security while ensuring system performance [7]. Qingan et al. proposed an optimization and application scheme for Hyperledger Fabric, which integrates national encryption standards into the Fabric architecture through its pluggable mechanism to enhance cryptographic components, and explores key technologies in blockchain security application protocols. A blockchain-based identity authentication protocol was proposed, which combines blockchain certificates with Fabric Certificate Authority (Fabric CA) to achieve identity verification, and adopts a dual signature method to enhance security and reliability [8]. Mahdi et al. proposed a framework based on Hyperledger, which utilizes blockchain technology to provide healthcare workers with a secure and reliable platform for remote interaction and transaction processing with patients. Blockchain could become an integrated digital service solution, effectively protecting patient data privacy and supporting medical payments. This framework was built on the existing Hyperledger platform and aimed to achieve a secure blockchain-assisted remote medical environment [9]. The literature comparison table is shown in Table 1.

In summary, existing research mainly relies on static small-scale evaluations, leaving a key knowledge gap: the performance degradation caused by the dynamic coupling of device size, policy complexity, and transaction frequency in real IoT deployments. Previous studies often regarded security, latency, and energy efficiency as isolated indicators, ignoring how the cumulative computational burden of continuous encryption and consensus directly competes with scalability in resource-constrained environments. Based on this, this study proposes a privacy protection method based on Hyperledger Fabric to ensure the security of IoT data. This solution innovatively integrates Advanced Encryption Standard (AES), Interplanetary File System (IPFS) distributed storage, and Shamir secret sharing mechanism on the license chain, achieving secure data storage and flexible sharing. By comprehensively evaluating these intertwined factors and proposing a collaborative framework for jointly optimizing endorsement strategies, network sharding, and energy consumption perception mechanisms, the study aims to bridge this gap and enhance the scalability and long-term sustainability of the system. At the same time, contract modules such as device management, key management, and access control are designed to ensure the reliability of the system in a multi-organizational collaborative environment. The innovation is not simply transferring federated learning to underwater sensor networks, but rather constructing a collaborative optimization mechanism for high latency, low bandwidth, and highly energy constrained environments: ① The adaptive layered training is proposed to dynamically couple update frequency with link quality and remaining energy; ② Dual weight scheduling of energy and task importance are introduced to balance overhead and model contribution; ③ Median aggregation is improved by combining gradient distribution characteristics to enhance robustness and convergence stability under high packet loss and strong noise. The technological increment is reflected in the mechanism coupling and robust optimization for complex underwater constraints, rather than the scenario-based application of existing methods. This study aims to ensure data privacy and security while improving operational efficiency, providing secure and practical solutions for IoT privacy protection.

2 Methods

2.1 Privacy preserving data sharing scheme based on hyperledger fabric

Hyperledger Fabric is an open-source distributed ledger framework contributed by IBM, which adopts an innovative execution sorting verification architecture. Its

highly modular and scalable design aims to provide a core foundation for deploying a permissioned blockchain. The ledger structure is shown in Figure 1 [10].

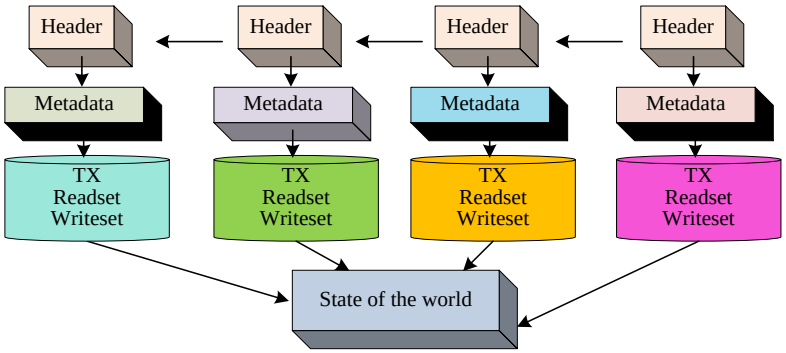


Figure 1: Hyperledger fabric ledger structure

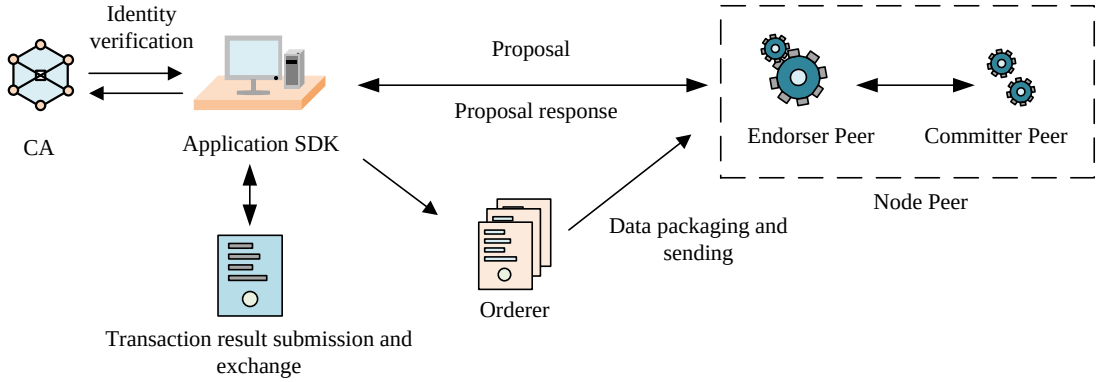


Figure 2: Fabric transaction process

In Figure 1, the ledger consists of two core parts: a blockchain that stores all transaction history and a world state database that maintains the latest state of data objects. When executing transactions with chaincode, to obtain the latest data, the key values in the world state database can be directly read. The transaction process of Fabric is shown in Figure 2 [11].

In Figure 2, the client application first obtains an authentication certificate through a Certification Authority (CA) and initiates a transaction proposal request using a Software Development Kit (SDK) [12]. The proposal is sent to the endorsement node, which executes the chain code and generates a response, and returns it to the client. After collecting sufficient transaction endorsements, the client submits them to the sorting node, packages them into blocks, and broadcasts them to the submitting node. Finally, the verification, ledger writing, and status update are completed, thus ending the entire transaction processing flow. Furthermore, the trusted execution of blockchain is combined with distributed storage to achieve secure storage and sharing of data in the IoT environment. The data security, storage, and sharing solution based on Hyperledger Fabric is shown in Figure 3.

In Figure 3, the data owner first encrypts the original data using the AES algorithm and stores the encrypted file in the IPFS network, obtaining the corresponding hash value as the unique identifier of the data. The pseudo-code of AES algorithm is as follows.

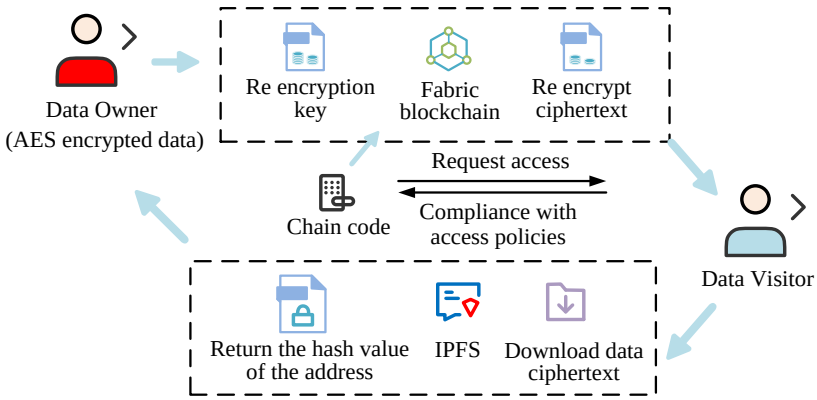


Figure 3: Data security storage and sharing solution

Algorithm 1: AES Encryption
Input: Master key K, Plaintext P, Associated data A
Output: Ciphertext C, Initialization vector IV, Authentication tag T
1. (K_enc, K_auth) ← HKDF(K, 384) // 256-bit for encryption, 128-bit for auth
2. IV ← GenerateRandomNonce(96 bits) // e.g., Timestamp + Counter
3. C, T ← AES-256-GCM_Encrypt(K_enc, IV, P, A)
4. return (IV, C, T)

Key encapsulation technology is utilized to re-encrypt data encryption keys and upload ciphertext and access policies together to the Fabric blockchain to achieve secure key sharing and policy execution. When a data visitor initiates a request, the chaincode verifies their permissions based on a preset policy. Subsequently, the blockchain returns the re-encryption key and related policy information. Visitors generate decryption keys based on their own credentials, restore AES keys, and ultimately download ciphertext data from IPFS and decrypt it. This scheme is built on the Hyperledger Fabric license chain, and all organizations joining the network must hold a valid certificate issued by a CA. CA generates a key pair for the data owner when issuing certificates, as shown in equation (1) [13].

$$\begin{cases} P_O = H_1(ID_O) \\ S_O = H_1(ID_O)^s \end{cases} \tag{1}$$

In equation (1), P_O, S_O, and ID_O are the public key, private key, and identity identifier of the data owner. s is the system master key. $H_1(\cdot)$ is a hash function that maps identity identifiers to a point or group element on an elliptic curve, ensuring uniqueness and collision resistance. The public-private key pairs of data visitors are shown in equation (2) [14].

$$\begin{cases} P_V = H_1(ID_V) \\ S_V = H_1(ID_V)^s \end{cases} \tag{2}$$

In equation (2), P_V is the public key of the data visitor, S_V is the private key of the visitor, and ID_V is their identity identifier. The data owner generates a random symmetric key, encrypts the original data into a ciphertext, uploads it to IPFS, and obtains the corresponding hash address. On this basis, the data owner combines with the symmetric key and uses P_V for re encryption to generate a two-layer ciphertext. Randomness is introduced during re-encryption to restrict the proxy's authority and ensure secure, valid ciphertext transformation. The resulting combined message hash supports subsequent access and verification. The calculation of the message hash is shown in equation (3) [15].

$$\begin{cases} C_1 = G^r \\ C_2 = H_M \cdot e(g^s, H1(ID_O))^r \\ C_3 = G^{rx} \\ C = (C_1, C_2, C_3) \end{cases} \tag{3}$$

In equation (3), G is the generator in the bilinear group. r is a temporary random number randomly selected by the data owner. H_M is the message hash, and $e(\cdot)$ is the bilinear mapping operation. x is another random number used to limit the agent's ability in ciphertext conversion. C is the complete ciphertext. C_1, C_2, and C_3 are the three components of ciphertext. The data visitor acts as a proxy and performs a re-encryption operation on the ciphertext locally using an identity-based conditional proxy re-encryption key, as shown in equation (4).

$$\begin{cases} C_1' = C_1 \\ C_2' = C_2 \cdot \dfrac{e(C_1, r_2)}{e(C_3, r_1)} \\ C_3' = r_1 \\ C_4 = r_2 \end{cases} \tag{4}$$

In equation (4), C_1' inherits the first part C_1 of the original ciphertext. C_2' is the re encrypted core ciphertext part. C_3' and C_4 are newly introduced random elements as one of the decryption auxiliary components. r_1 and r_2 are random values introduced by agents during the re-encryption process to enhance security and prevent ciphertext reuse attacks. The new ciphertext is represented as shown in equation (5).

$$C_V = (C_1, C_2', C_3', C_4) \tag{5}$$

In equation (5), C_V is the transformed ciphertext that can be successfully decrypted by the designated data visitor. The sequence diagram of the data sharing scheme is shown in Figure 4.

In Figure 4, the data owner generates a random symmetric key to encrypt the data, uploads the ciphertext to IPFS, and obtains its hash. The hash and symmetric key are then used to define an access policy via the Fabric blockchain and encrypted with the visitor's public key before being recorded on-chain. During access, if chaincode verification satisfies the policy, a proxy re-encryption key is returned, allowing the proxy to transform the ciphertext for the visitor. The visitor decrypts it with a private key to obtain the symmetric key and hash, retrieves the ciphertext from IPFS, and recovers the plaintext.

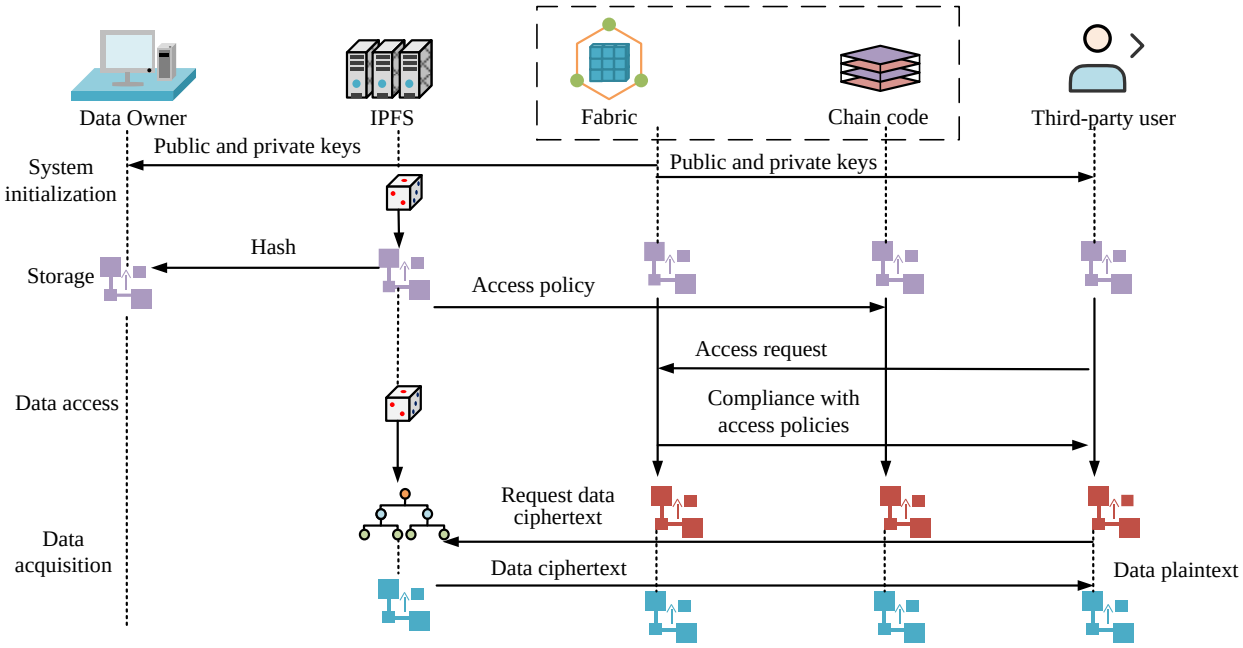


Figure 4: Sequence diagram of privacy protection data sharing scheme

2.2 IoT privacy protection scheme based on hyperledger fabric

This study constructs a data security storage and sharing solution based on Hyperledger Fabric, which combines blockchain and distributed storage to achieve trusted interaction of data on and off the chain. However, relying solely on secure storage and sharing mechanisms is still insufficient to fully address the complex privacy protection requirements in IoT environments, especially in terms of access control and key management. On this basis, this study integrates data encryption, access control, and key management mechanisms within the same architecture to form a complete privacy protection closed loop. The data are encrypted and stored in a distributed system, with hash and access policies recorded on the chain. During the access phase, identity authentication and permission verification are completed through smart contracts, and controlled decryption is achieved through a key mechanism. The overall framework is shown in Figure 5.

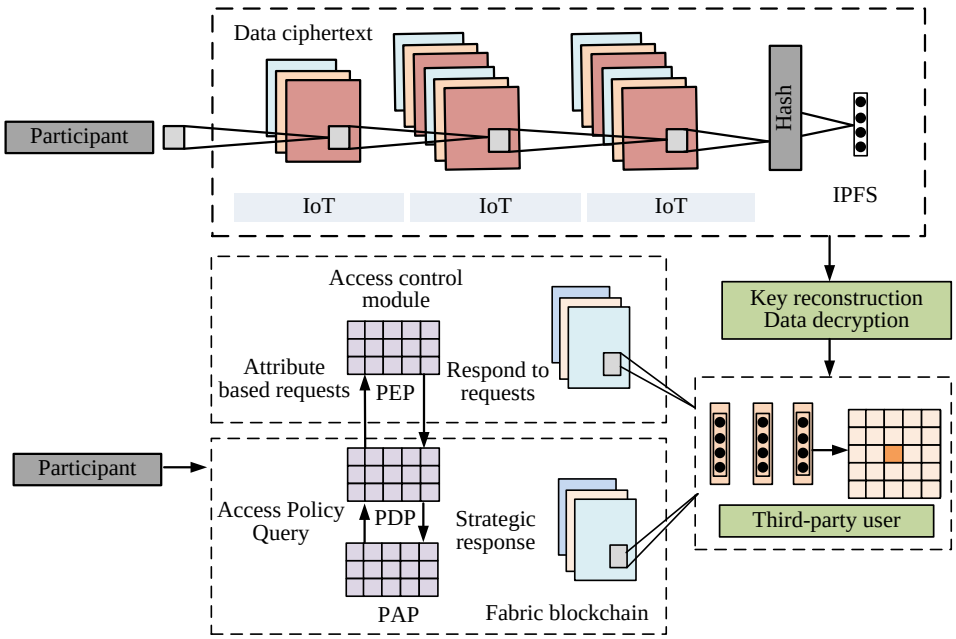


Figure 5: Privacy protection scheme for the IoT based on hyperledger fabric

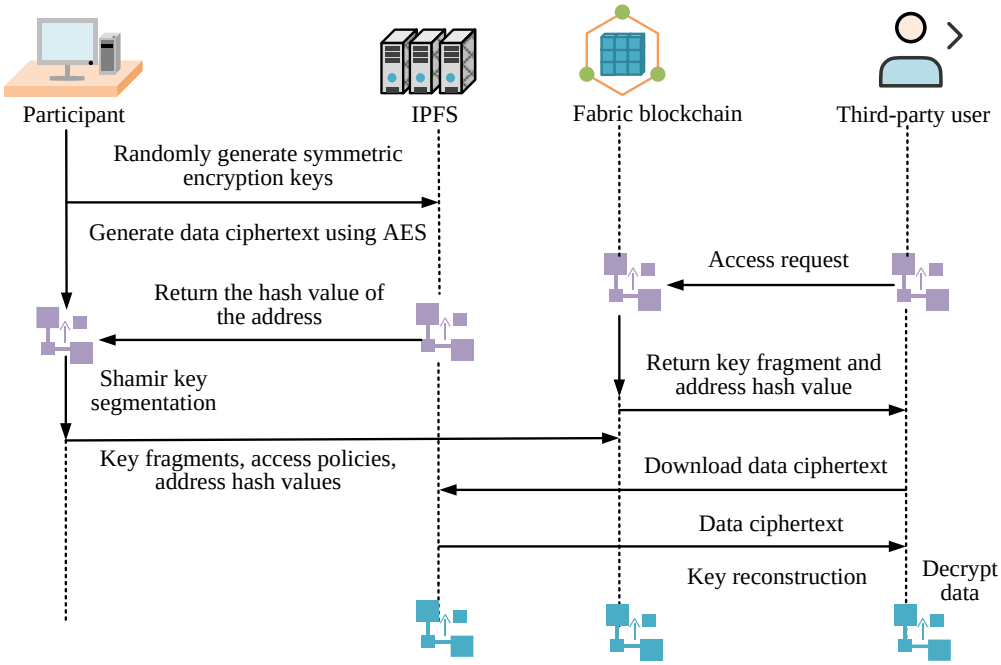


Figure 6: Time sequence diagram of IoT privacy protection scheme

In Figure 5, IoT devices first complete data collection through a gateway, and then the data owner encrypts the data using the AES algorithm and uploads the encrypted data to IPFS to obtain the corresponding address hash value and key information. The hash value and key will be written into the Fabric blockchain through a chain code to ensure traceability and verifiability. When third-party users need to access data, they will initiate requests to the access control module, including address hash and key credentials [16]. After verification is passed, the system generates and returns an access key, allowing users to obtain encrypted data from IPFS based on the address hash and decrypt it using the key to obtain the original plaintext. The timing diagram of the IoT privacy protection scheme based on Hyperledger Fabric is shown in Figure 6.

In Figure 6, participants randomly generate symmetric encryption keys and use the AES algorithm to encrypt the original data to obtain a ciphertext. Then, the ciphertext is uploaded to IPFS, and the corresponding address hash value is obtained. To enhance security, the generated key will be divided into multiple shares through the Shamir secret sharing scheme, and the hash values of the shares will be written into the blockchain together with the access policy. When a third-party user needs to access data, they initiate a request to the blockchain. The blockchain code will verify the legitimacy of the request based on a preset policy. If the conditions are met, the corresponding key share and address hash value will be returned. The user obtains the ciphertext of the data from IPFS and reconstructs it by combining it with the key share, ultimately decrypting the original plaintext data. Shamir secret sharing is divided into two core stages: distribution and reconstruction. In the distribution phase, the system first obtains the master key and constructs a $k-1$ degree polynomial based on it. Then, it uses this polynomial to generate n key shares, and finally distributes these shares to n endorsement nodes for distributed storage. In the reconstruction phase, the system collects at least k shares provided by nodes, restores the original polynomial through Lagrange interpolation algorithm, extracts the constant term or master key from it, and finally uses this key to decrypt the data stored off chain. In Fabric blockchain, users need to hold a certificate to access the network. CA is responsible for issuing identity certificates and user certificates, and generating public-private key pairs based on the system master key and device identifier, as shown in equation (6) [17].

$$\begin{cases} P_{IoT} = H_1(DeviceId) \\ S_{IoT} = H_1(DeviceId) \end{cases} \tag{6}$$

In equation (6), P_{IoT} and S_{IoT} are the public and private keys corresponding to the IoT device. When executing the Shamir secret sharing reconstruction algorithm locally, third-party users need to meet access policy requirements. If the number of key fragments obtained is less than the threshold value, the key cannot be recovered. If it is not less than the threshold value, the decryption key can be reconstructed using the Lagrange interpolation formula, as shown in equation (7) [18].

$$\begin{cases} f(\alpha) = \sum_{i=1}^{t} S_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{t} \dfrac{\alpha - \alpha_j}{\alpha_i - \alpha_j} \\ K = \sum_{i=1}^{t} S_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{t} \dfrac{\alpha_j}{\alpha_j - \alpha_i} \end{cases} \tag{7}$$

In equation (7), K is the original symmetric key, α is the coordinate point, S_i is the i-th key share, and t is the threshold value. t is empirically determined through sensitivity analysis to optimize the trade-off between security strength and computational overhead. The study evaluates the key reconstruction delay of the system under different t-values. Preliminary experiments have shown that although theoretically higher t-values can provide

stronger resistance to collusion attacks, they exponentially increase the computational cost and communication epochs required for polynomial interpolation during key reconstruction. Setting $t = 3$ in a 4-node network achieves the lowest overall latency while meeting the minimum-security requirement of tolerating at least one node being compromised ($t > n/2$). Therefore, $t = 3$ is selected as the optimal working point in the experiment. After obtaining K , third-party users can decrypt the ciphertext to restore the data. In terms of node participation strategy, the study introduces an energy perception model based on the nonlinear growth law of underwater energy consumption and communication distance. In terms of cluster head selection, to address the problem of static or random strategies being difficult to adapt to dynamic channels, multiple indicators, such as link stability, remaining energy, and node connectivity, are used to evaluate and improve upload success rates and aggregation efficiency. Topology stability is incorporated into training scheduling. For non-independent identically distributed and asynchronous updates, improved median aggregation is adopted to enhance the robustness against abnormal gradients, and gradient validity screening and time decay weights are introduced to alleviate the impact of expired gradients on convergence. Each mechanism constructs a collaborative optimization framework around three core issues: energy constraints, communication limitations, and data heterogeneity, to ensure theoretical consistency between training strategies and network operational characteristics.

3 Results

3.1 Performance Comparison Test of IoT Privacy Protection Solutions

To verify the feasibility and efficiency of the proposed solution in the context of the IoT, performance tests are conducted based on the Ubuntu 22.04 operating system. The server is configured with an Intel Core i7-12700K processor, 32GB of memory, 1TB of NVMe SSD storage, and equipped with an NVIDIA RTX 3080 GPU for acceleration-related calculations. The blockchain framework adopts Hyperledger Fabric v2.5.1, Docker version 24.0, Go language version 1.20, and the block size is set to 100 transactions. The SWaT dataset is selected. This dataset is sourced from a real industrial control system testing platform and contains operational data from various sensors and actuators in the water treatment process. The dataset contains the operating data of core edge devices, such as programmable logic controllers, and communicates using industry-standard protocols to truly reflect the behavioral characteristics of the underlying IoT devices. It also covers normal operation data collected over multiple days and 36 different types of attack scenarios, providing rich annotations for evaluating the system's robustness under security threats. As an international standard benchmark dataset in industrial IoT security, SWaT's widespread recognition in academia and industry further validates its authority as representative testing data. The schemes based on Hyperledger Fabric, Multi-Authority Attribute-Based Encryption (MA-ABE) [19], and Lightweight Privacy-preserving Blockchain Federated Learning (LPBFL) [20] are compared. It is divided into training and testing sets in a 7:3 ratio. The throughput changes of different methods on the dataset are shown in Figure 7.

Figures 7 (a) and (b) show the throughput performance of three schemes on the dataset. In Figure 7 (a), there is a significant difference in throughput performance among the three schemes under the training set conditions. The TPS of the Hyperledger Fabric solution consistently remains around 370-420, with minimal overall fluctuations, demonstrating high stability and processing capability. The TPS of MA-ABE is stable in the range of 320-360, slightly lower than Fabric, but still able to maintain good performance, indicating that it can balance safety and efficiency during the training data stage. LPBFL is concentrated between 270-310, with a significant decrease compared to the previous two, which is related to the additional overhead in encryption, signature, and batch verification. However, the overall trend remains stable, indicating that its lightweight design has controllable performance costs during the training phase. In Figure 7 (b), the TPS distribution of the Hyperledger Fabric case is around 360-410, maintaining the highest performance. MA-ABE is located in the range of 310-350, slightly lower than Fabric. LPBFL is maintained between 260 and 300, with the lowest throughput. Compared to the training phase, the overall TPS of the three schemes slightly decreases on the test set, but the difference is not significant, indicating that different methods have a certain degree of robustness when facing unknown test data. The average delay time of different methods is shown in Figure 8.

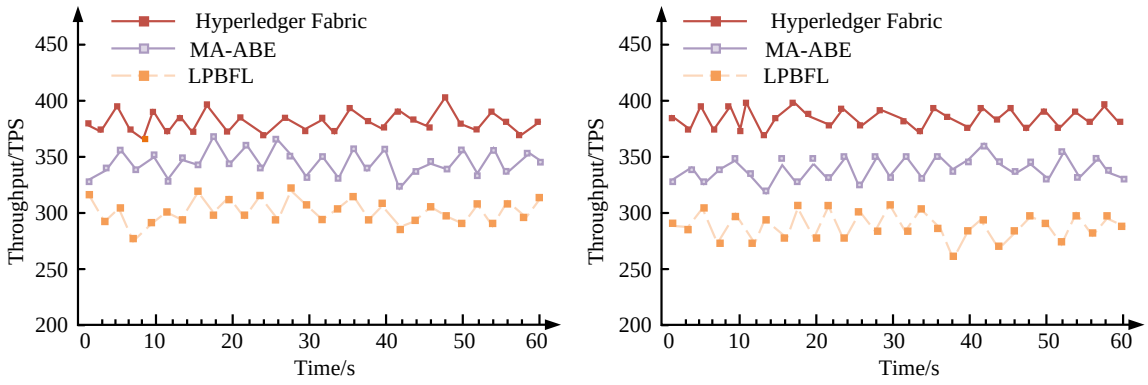


Figure 7: Throughput changes of different methods

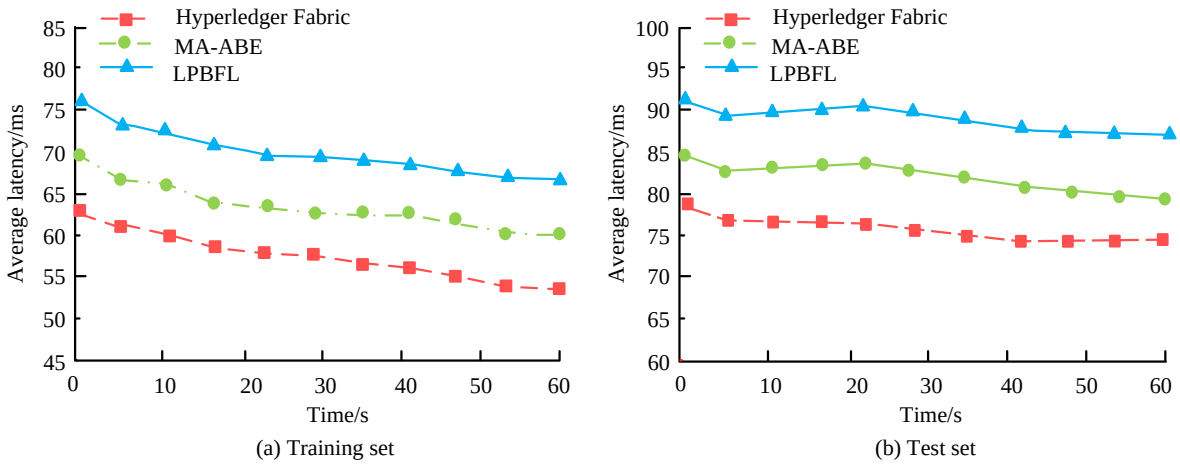


Figure 8: The average delay time of different methods

In Figure 8 (a), the Hyperledger Fabric scheme has the lowest latency, with an initial value of about 60 ms, gradually decreasing to nearly 50 ms over time, demonstrating extremely high response efficiency. The delay level of MA-ABE is in the middle range, starting at about 70 ms and finally stabilizing at around 60 ms, which is about 10 ms higher than Fabric, but still maintains a relatively stable trend. The delay of LPBFL is the highest, with an initial value of around 80 ms, which slowly decreases over time and eventually stabilizes at around 70 ms, significantly higher than the other two schemes. This indicates that under the training set conditions, LPBFL brings higher system latency while ensuring privacy protection and security. In Figure 8 (b), the delay of the three schemes in the test set stage is higher than that in the training set, but the overall trend remains stable. The average latency of the Hyperledger Fabric solution starts at 78 ms, gradually decreases, and eventually stabilizes at around 72 ms, still showing the best performance among the three. LPBFL has the highest latency, starting at nearly 90 ms and ultimately maintaining around 85 ms, with a difference of 10-15 ms compared to Hyperledger Fabric. Under the test set conditions, the delay differences among the three schemes are further amplified, indicating that privacy protection mechanisms have a significant impact on access delay, especially LPBFL, which sacrifices some performance while enhancing security. To comprehensively evaluate the security performance and system stability of the proposed solution under privacy enhancement and performance optimization mechanisms, the study further selects Privacy Preserving Video Using Hyperledger Fabric (P2V Fabric) [21] and High Concurrency Cross-Sharding on Permissioned Blockchains-Fabric (Hicocs-Fabric) [22] for comparative testing. The resource cost results of different methods on the training and testing sets are shown in Table 2.

Table 2: Calculate the cost results

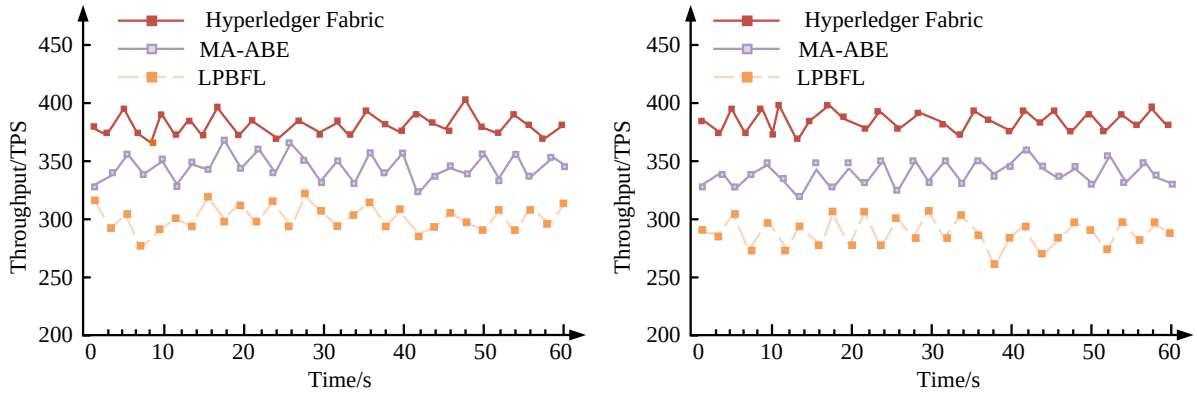
Data set	Method	CPU (%)	Memory (MB)	Storage growth rate (MB/day)	Estimated Energy (W)	Energy per TPS (W/TPS)
Training set	LPBFL	42	435	18	52.5	0.19
	MA-ABE	62	625	26	77.5	0.24
	P2V-Fabric	57	560	24	71.3	0.21
	Hicocs-Fabric	55	540	22	68.8	0.15
	Hyperledger Fabric	53	515	22	66.3	0.17
Test set	LPBFL	40	415	17	50.1	0.20

	MA-ABE	60	605	25	75.0	0.25
	P2V-Fabric	54	545	23	67.5	0.22
	Hicocs-Fabric	52	525	21	65.0	0.16
	Hyperledger Fabric	50	505	21	62.5	0.18

In Table 2, there are significant differences in resource consumption among the three methods in the training and testing stages. MA-ABE has the highest CPU usage, reaching 62% and 60%, while also having the highest memory consumption, maintaining at 625 MB and 605 MB. Its storage growth rate is also at the highest level, indicating that it has brought significant system load while introducing multiple attribute encryption and complex policy verification. In contrast, LPBFL has the lowest resource overhead, with a CPU usage rate of 42% in the training set and 40% in the test set. Its memory consumption remains at 435 MB and 415 MB, and its storage growth rate is the lowest, only 18 MB/day and 17 MB/day. This indicates that its lightweight design effectively reduces the additional burden of computation and storage while ensuring privacy protection. The resource consumption of P2V Fabric and Hicocs Fabric is between MA-ABE and LPBFL. The resource consumption of the Hyperledger Fabric solution is between the two, with CPU usage rates of 53% and 50% in the training and testing sets, memory consumption of 515 MB and 505 MB, and storage growth rates of 22 MB/day and 21 MB/day. This reflects that the resource burden is relatively moderate when complex encryption mechanisms are not introduced. In terms of energy consumption indicators, MA-ABE has the highest power consumption, at 77.5W and 75.0W, respectively, while LPBFL has the lowest power consumption, at 52.5W and 50.1W, respectively. The other schemes show a stepwise decreasing distribution. In terms of unit throughput energy consumption indicators, Hicocs Fabric exhibits the best energy efficiency, with training and testing sets of 0.15 W/TPS and 0.16 W/TPS, respectively, indicating that it achieves a better throughput energy consumption ratio while ensuring functional enhancement. Overall, Hicocs Fabric performs the best in terms of unit performance and energy efficiency, maintaining relatively reasonable energy efficiency while ensuring stable system operation. It should be pointed out that the experiment is based on a standard server platform, and the result of CPU usage of about 50% mainly reflects the overall resource utilization of the system in a fully functional operating state. This load is acceptable for strong computing devices such as edge nodes or gateways, but may be high for lightweight IoT terminals, especially increasing energy consumption during continuous or high-frequency interactions. Therefore, the framework is suitable for a layered deployment mode dominated by gateway/edge nodes, rather than directly running on ultra-low power terminals.

3.2 Analysis of the application Effectiveness of IoT privacy protection solutions

To comprehensively evaluate the security and reliability of the proposed solution in practical application environments, this study simulates four typical attack scenarios in the experimental environment: unauthorized access, replay attacks, data tampering, and access after revocation. To verify the defense capability and stability of the system under different attack conditions, the success rates of data integrity verification and data leakage rates of different methods under four different attacks are shown in Figure 9.



(a) Training set (b) Test set

Figure 9: Success rate of data integrity verification and data leakage rate results

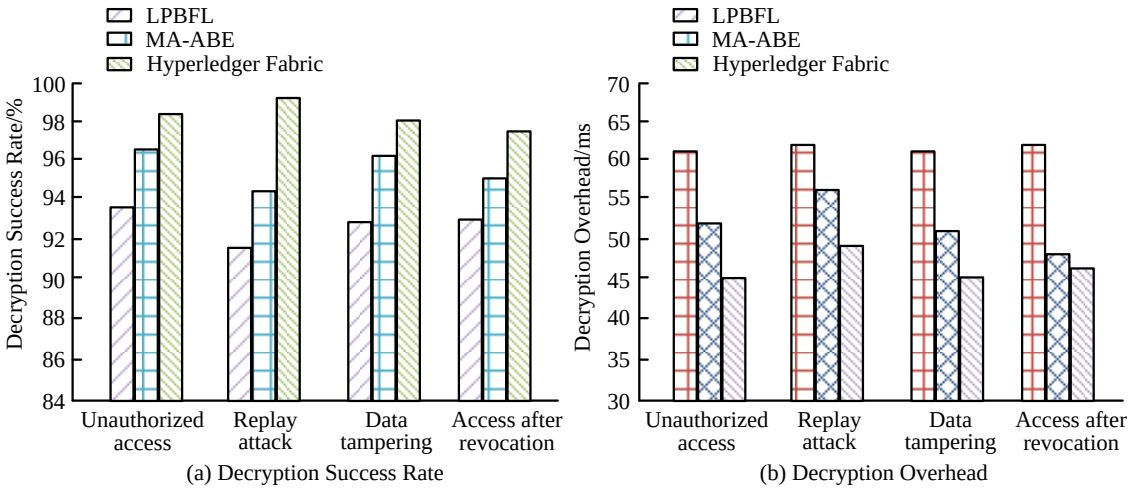


Figure 10: Decryption success rate and average decryption cost results

In Figure 9(a), the Hyperledger Fabric scheme achieves the highest integrity verification rate, remaining above 80% under unauthorized access and replay attacks. MA-ABE follows at around 70%, while LPBFL falls below 70% in these scenarios. Under data tampering and access-after-revocation attacks, verification rates improve overall: Hyperledger Fabric approaches 100%, MA-ABE exceeds 90%, and LPBFL reaches about 85%, highlighting the effectiveness of blockchain-based certificate storage and hash verification against tampering. In Figure 9(b), LPBFL shows the highest data leakage rate, nearing 8% during unauthorized access and replay attacks. MA-ABE keeps leakage below 5% across all scenarios, demonstrating stronger access control. Hyperledger Fabric performs best, limiting leakage to around 3% under unauthorized access and replay attacks and nearly 0% under tampering and post-revocation access, indicating enhanced privacy protection with minimal overhead. The decryption success rates and average decryption costs of different methods under four types of attacks are shown in Figure 10.

In Figure 10(a), the Hyperledger Fabric scheme maintains a decryption success rate of 98%–99% across unauthorized access, replay, tampering, and post-revocation scenarios, ensuring reliable plaintext recovery under attacks. MA-ABE achieves 94%–96%, sustaining high availability, while LPBFL remains below 94%, with a more noticeable drop after revocation, indicating reduced reliability. In Figure 10(b), Hyperledger Fabric shows the lowest decryption cost at 45–50 ms and remains stable across all attacks, reflecting lightweight efficiency. MA-ABE ranges from 50–58 ms, outperforming LPBFL, which exceeds 60 ms in all scenarios, suggesting higher computational overhead. The application effects of different methods under four types of attacks are shown in Table 3.

In Table 3, there are significant differences in the application effects of the three methods in the four types of attack scenarios. When unauthorized access occurs, the Hyperledger Fabric solution has the highest attack detection rate of 99.6%, the lowest false positive rate of only 0.5%, and a system availability of up to 99%, significantly better than MA-ABE and LPBFL. In replay attacks, the Hyperledger Fabric scheme also performs the best, with a detection rate of 99.8%, a false positive rate of only 0.4%, and an availability rate of 98%. The detection rate of MA-ABE is 97.5%, with a usability of 94%, while LPBFL is 95.2% and 91%, further highlighting the gap. In the scenario of data tampering, the detection rate of the Hyperledger Fabric solution is 99.4%, the false positive rate is only 0.3%, and the availability is as high as 99%. The detection rate of MA-ABE is 98.9%, and the availability is 96%, while LPBFL is 96.7% and 93%. The detection rates of P2V Fabric and Hicocs Fabric are 97.6% and 98.2%, respectively, with false positive rates of 1.5% and 1.3%. The system availability is 94% and 95%, indicating that both have strong recognition capabilities in integrity attack scenarios. In the case of revoked access, the detection rate of the Hyperledger Fabric scheme reaches 99.7%, the false positive rate remains at 0.6%, and the availability is 98%, which is better than MA-ABE's 97.1% detection rate and 94% availability, as well as LPBFL's 95.3% detection rate and 90% availability. In the threat model, it is assumed that the attacker has the ability to passively eavesdrop, replay, unauthorized access, and compromise some nodes, but cannot break the standard cryptography assumptions, and the blockchain consensus satisfies an honest majority, and the smart contract execution environment cannot be tampered with. The system must meet the requirements of data confidentiality, integrity, correct access control, and resistance to replay and forgery. The security of proxy re-encryption and key sharding is based on the bilinear mapping problem and the information theory security of Shamir secret sharing under an insufficient threshold, respectively. Under this assumption, attackers cannot break through the security boundaries of the system in polynomial time, so the framework theoretically meets the requirements of confidentiality, integrity, and access control.

Table 3: Comparison of security indicators of three methods under four types of attacks

Data set	Method	Attack detection rate (%)	False positive rate (%)	System availability under attack (%)
Unauthorized access	LPBFL	94.2	2.1	92
	MA-ABE	96.8	1.6	95
	P2V-Fabric	95.5	1.9	93
	Hicocs-Fabric	96.1	1.7	94
	Hyperledger Fabric	99.6	0.5	99
Replay attack	LPBFL	95.2	2.5	91
	MA-ABE	97.5	1.3	94
	P2V-Fabric	96.3	2.1	92
	Hicocs-Fabric	97.8	1.6	93
	Hyperledger Fabric	99.8	0.4	98
Data tampering	LPBFL	96.7	1.8	93
	MA-ABE	98.9	1.2	96
	P2V-Fabric	97.6	1.5	94
	Hicocs-Fabric	98.2	1.3	95
	Hyperledger Fabric	99.4	0.3	99
Access after revocation	LPBFL	95.3	2.2	90
	MA-ABE	97.1	1.4	94
	P2V-Fabric	96.3	1.8	92
	Hicocs-Fabric	96.7	1.6	93
	Hyperledger Fabric	99.7	0.6	98

4 Discussion

In response to the privacy protection requirements in IoT environments, this study proposed and implemented a privacy protection scheme based on Hyperledger Fabric. By introducing AES encryption and IPFS distributed storage, combined with Shamir secret sharing and policy management contracts, secure storage and controllable sharing of data have been achieved. In the experiment, this scheme had significant advantages in both performance and security, with a stable throughput of 370-420 TPS and an average latency as low as 50-72 ms. In terms of resource overhead, the CPU occupied about 50%, and the memory was about 510 MB, which was significantly better than the comparison method. In terms of attack defense, the detection rate was generally above 99%, the false positive rate was below 0.6%, the system availability exceeded 98%, the data leakage rate was close to 0, and the integrity verification rate remained at 100%. In terms of decryption performance, the success rate of decryption remained stable at 98-99%, with an average cost of only 45-50 ms. The research results fully verify the superiority and practical value of this scheme in IoT privacy protection scenarios. Compared with existing methods, the proposed approach improves system architecture rather than a single metric. Unlike schemes relying on multi-attribute encryption or on-chain centralized verification, it decouples large-scale storage and policy execution through on/off-chain collaboration, reducing consensus overhead and enhancing scalability. Compared to solutions focused only on lightweight computation or local security, it establishes a closed loop among access control, key management, and data verification, ensuring consistent and traceable policy enforcement in multi-organization settings. The layered protection mechanism also creates multiple security boundaries across identity, policy, and data integrity, offering greater stability than single-layer models. Overall, the contribution lies in architectural-level optimization rather than parameter tuning or local algorithm refinement.

Although the proposed scheme performs well under the current configuration, validation is mainly based on simulated data and idealized network models rather than real deployment environments, so its engineering applicability requires further evaluation. Simulation settings adopt controllable communication models and node parameters, which facilitate mechanism analysis but may underestimate uncertainties such as link fluctuations, hardware heterogeneity, and noise interference. Moreover, the current task scale is limited and does not cover large-scale concurrency or high-dimensional complex data scenarios, and the model's generalization ability under such conditions remains to be verified. In terms of scalability, as device numbers and on-chain interactions grow, the system may encounter accumulated communication delays, higher endorsement loads, and increased consensus overhead. Heterogeneous computing power and energy levels may further aggravate training imbalance. Although the proposed hierarchical structure and energy-aware mechanism can theoretically mitigate these pressures, their performance limits in ultra-large networks require system-level validation. Future work will conduct real hardware deployment and long-term testing, and integrate sharding or layered optimization, off-chain indexing, and dynamic scheduling to improve high-concurrency scalability. In large-scale IoT environments, continuous encryption, key updates, and consensus processes may also impose cumulative computational burdens, especially on resource-constrained nodes. Therefore, further studies will evaluate long-term energy consumption and explore lightweight cryptography, energy-aware scheduling, and model compression to reduce overhead while maintaining security and privacy.

5 Conclusion

A collaborative framework integrating blockchain verification, distributed storage, and policy-driven access control is proposed to address the challenge of balancing

data sharing efficiency and privacy protection in the IoT. This framework achieves a structural balance between security and system performance. The results indicate that through the collaborative design of an on-chain trust mechanism and off-chain data management, data trust sharing and fine-grained access control can be achieved without centralized trust nodes. Unlike existing methods that only optimize a single security mechanism, this study shows that the performance improvement of privacy protection systems can be achieved through architectural optimization rather than relying solely on complex encryption algorithms. This discovery emphasizes that when designing distributed security systems, priority should be given to the collaborative relationship between structure and mechanism, optimizing performance and security trade-offs at the overall level. This achievement can be applied to cross-organizational collaboration scenarios such as medical data exchange and industrial IoT platforms. The method's framework has good transferability, supporting flexible replacement of encryption algorithms or access policy modules according to business needs. Future work will focus on stability verification in real environments and lightweight optimization for resource-constrained scenarios.

References

[1] Batamu Anderson Chiphiko, Hyunsung Kim, Patrick Ali, and Levis Eneya. Forward secrecy attack on privacy-preserving machine authenticated key agreement for internet of things. Archives of Advanced Engineering Science, 3(1):29-34, 2025. https://doi.org/10.47852/bonviewAAES32021937

[2] Minfeng Qi, Ziyuan Wang, Qinglong Han, Jun Zhang, Shiping Chen, and Yang Xiang. Privacy protection for blockchain-based healthcare IoT systems: A survey. IEEE/CAA Journal of Automatica Sinica, 11(8):1757-1776, 2022. https://doi.org/10.1109/JAS.2022.106058

[3] Xia Hua, and Hongzhen Zhang. International trade privacy data management system combining internet of things blockchain. Intelligent Decision Technologies, 18(1):211-222, 2024. https://doi.org/10.3233/IDT-230393

[4] Yun Xie, Peng Li, Nadia Nedjah, Brij B. Gupta, David Taniar, and Jindan Zhang. Privacy protection framework for face recognition in edge-based internet of things. Cluster Computing, 26(5):3017-3035, 2023. https://doi.org/10.1007/s10586-022-03808-8

[5] Nana Kong, Zhifeng Wan, Cui Xu, Xukai Liu, Yixin Yuan, and Shuanggen Liu. An efficient certificateless key-insulated anonymous signature scheme based on smart contract for data sharing in industrial internet of things. IEEE Internet of Things Journal, 12(6):6930-6942, 2025. https://doi.org/10.1109/JIOT.2024.3491136

[6] Tatsuya Sato, Taku Shimosawa, Yuki Kondo, and Nao Nishijima. Toward fully-decentralized system with hyperledger fabric. IEICE Communications Express, 12(5):223-229, 2023. https://doi.org/10.1587/comex.2023XBL0011

[7] Shahroz Abbas, Ajmery Sultana, and Georges Kaddoum. Quantum-safe blockchain in hyperledger fabric. IEEE Networking Letters, 7(1):61-65, 2024. https://doi.org/10.1109/LNET.2024.3522966

[8] Qingan Zheng, Jialin Meng, Junjie Wu, Jingtao Li, and Haonan Lin. Research and implementation of trusted blockchain core technology based on state secret algorithm. China Communications, 22(4):143-160, 2025. https://doi.org/10.23919/JCC.fa.2024-0370.202504

[9] Syed Sarosh Mahdi, Zaib Ullah, Gopi Battineni, Muneer Gohar Babar, and Umer Daood. The telehealth chain: A framework for secure and transparent telemedicine transactions on the blockchain. Irish Journal of Medical Science, 193(5):2129-2137, 2024. https://doi.org/10.1007/s11845-024-03728-z

[10] Vidhi Thakkar, and Vrushank Shah. A privacy-preserving framework using hyperledger fabric for EHR sharing applications. International Journal of Electrical and Computer Engineering Systems, 14(6):667-676, 2023. https://doi.org/10.32985/ijeces.14.6.6

[11] Jianguo Yu, Lin Ge, and Minghui Wu. Proposal distribution optimization for endorsement strategy in hyperledger fabric. The Journal of Supercomputing, 80(10):15038-15065, 2024. https://doi.org/10.1007/s11227-024-06056-2

[12] Zuqiang Ke, and Nohpill Park. Performance modeling and analysis of hyperledger fabric. Cluster Computing, 26(5):2681-2699, 2023. https://doi.org/10.1007/s10586-022-03800-2

[13] Rafah Amer Jaafar, and Saad Najim Alsaad. Enhancing educational certificate verification with blockchain and IPFS: A decentralized approach using hyperledger fabric. TEM Journal, 12(4):2385-2395, 2023. https://doi.org/10.18421/TEM124-5134TU

[14] Yongxin Zhang, Jiacheng Yang, Hong Lei, Zijian Bao, Ning Lu, and Wenbo Shi. PACTA: An IoT data privacy regulation compliance scheme using TEE and blockchain. IEEE Internet of Things Journal, 11(5):8882-8893, 2023. https://doi.org/10.1109/JIOT.2023.3321308

[15] Panjun Sun, Shigen Shen, Yi Wan, Zongda Wu, Zhaoxi Fang, and Xiaozhi Gao. A survey of IoT privacy security: Architecture, technology, challenges, and trends. IEEE Internet of Things Journal, 11(21):34567-34591, 2024. https://doi.org/10.1109/JIOT.2024.3372518

[16] Rajdeep Chatterjee, Rohit Halder, Tanmoy Maitra, and Santosh Pani. A computer vision-based perceived attention monitoring technique for smart teaching. Multimedia Tools and Applications, 82(8):11523-11547, 2023. https://doi.org/10.1007/s11042-022-14283-z

[17] Shahnawaz Ahmad, Mohd Arif, Javed Ahmad, and Shabana Mehfuz. A TOTP-based secure data storage system in the cloud environment using the JWT

token approach. International Journal of System Assurance Engineering and Management, 16(4):1565-1578, 2025. https://doi.org/10.1007/s13198-025-02775-8

[18] Zouhaier Brahmia, Fabio Grandi, and Rafik Bouaziz. τJOWL: A systematic approach to build and evolve a temporal OWL 2 ontology based on temporal JSON big data. Big Data Mining and Analytics, 5(4):271-281, 2022. https://doi.org/10.26599/BDMA.2021.9020019

[19] Chenhao Xu, Youyang Qu, Yong Xiang, Tom H. Luan, and Longxiang Gao. An optimized privacy-protected blockchain system for supply chain on internet of things. IEEE Internet of Things Journal, 11(5):9019-9030, 2023. https://doi.org/10.1109/JIOT.2023.3321889

[20] Mochan Fan, Kailai Ji, Zhaofeng Zhang, Hongfang Yu, and Gang Sun. Lightweight privacy and security computing for blockchained federated learning in IoT. IEEE Internet of Things Journal, 10(18):16048-16060, 2023. https://doi.org/10.1109/JIOT.2023.3267112

[21] Muhammad Saad, and Ki-Woong Park. P2V-Fabric: Privacy-preserving video using hyperledger fabric. Computers, Materials & Continua, 83(2):1881-1900, 2025. https://doi.org/10.32604/cmc.2025.061733

[22] Lingxiao Yang, Xuewen Dong, Zhiguo Wan, Di Lu, Yushu Zhang, and Yulong Shen. Hicocs: High concurrency cross-sharding on permissioned blockchains. IEEE Transactions on Computers, 74(7):2168-2182, 2025. https://doi.org/10.1109/TC.2025.3558001