TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 08/20 pp736–752
DOI: 10.26599/TST.2023.9010049
Volume 29, Number 3, June 2024

Effective Identity Authentication Based on Multiattribute Centers for Secure Government Data Sharing

Meiquan Wang, Junhua Wu*, Tongdui Zhang, Junhao Wu, and Guangshun Li

Abstract: As one of the essential steps to secure government data sharing, Identity Authentication (IA) plays a vital role in the processing of large data. However, the centralized IA scheme based on a trusted third party presents problems of information leakage and single point of failure, and those related to key escrow. Therefore, herein, an effective IA model based on multiattribute centers is designed. First, a private key of each attribute of a data requester is generated by the attribute authorization center. After obtaining the private key of attribute, the data requester generates a personal private key. Second, a dynamic key generation algorithm is proposed, which combines blockchain and smart contracts to periodically update the key of a data requester to prevent theft by external attackers, ensure the traceability of IA, and reduce the risk of privacy leakage. Third, the combination of blockchain and interplanetary file systems is used to store attribute field information of the data requester to further reduce the cost of blockchain information storage and improve the effectiveness of information storage. Experimental results show that the proposed model ensures the privacy and security of identity information and outperforms similar authentication models in terms of computational and communication costs.

Key words: blockchain; identity authentication; distribution; dynamic key generation

1 Introduction

The advent of the information age has promoted the development of data assets. It is a common goal of all government departments to accelerate interdepartmental data sharing and realize a digital service-oriented smart government[1, 2]. However, the efficiency of government data sharing is significantly affected because of data storage in separate departments, low security of shared information storage, and uncertainty of the sharer's identity[3]. Furthermore, it is difficult to assign responsibility for government data, which makes it challenging to share the data fully. Therefore, a secure and efficient government data-sharing solution is urgently needed.

Identity Authentication (IA) is the first step in securing government data sharing between departments. Security and privacy protection mechanisms are needed to restrict illegal access and use of valuable government data. However, traditional authentication mechanisms are generally based on trust provided by a third party[4], such as the well-known Public Key Infrastructure (PKI)[5], cloud-driven trusted certificate authority, and current internet address allocation strategy[6]. A conventional IA model, such as the simplified version presented in Fig. 1, typically comprises three parties: data requester, authorization

• Meiquan Wang, Junhua Wu, Junhao Wu, and Guangshun Li are with School of Computer Science, Qufu Normal University, Rizhao 276800, China. E-mail: wangmeiquan0409@163.com; shdwjh@163.com; wujunhao0716@163.com; guangshunli@qfnu.edu.cn.
• Tongdui Zhang is with Science and Technology Innovation Service Institution of Rizhao, Rizhao 276800, China. E-mail: rzkjhz@163.com.
* To whom correspondence should be addressed.
Manuscript received: 2022-12-19; revised: 2023-04-12; accepted: 2023-05-23

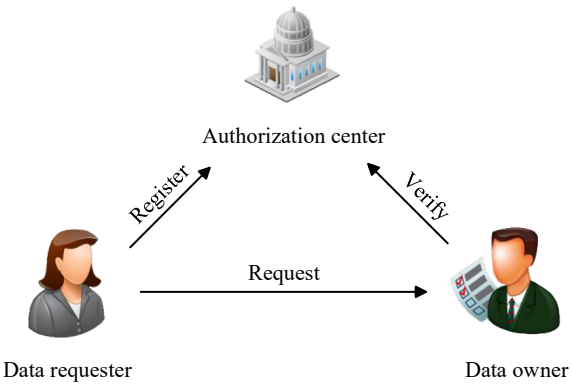


Fig. 1 Simplified conventional identity authentication process.

center, and data owner. The data requester requests identity registration from the identity authorization center. When data sharing is required, the data requester initiates a request to the data owner, who verifies the identity of the former through the authorization center. Conventional static password-based authentication methods are cost-effective, and their implementation is simple and fast because they use only passwords for authentication and do not require other resources; however, they are vulnerable to network attacks and have low security. Thus, multifactor authentication needs to be considered. Because the traditional IA is controlled by the central server rather than the users, the latter are forced to trust the authority[7]. A trusted authority controls the user's keys, which may lead to potential key escrow. Such a highly centralized feature can render the whole system unable to operate and cause a risk of a single point of failure in case the authority fails[8].

The digital identity management market is growing every year. The traditional centralized digital IA systems undergo frequent user data leakages even sales because several organizations lack privacy protection abilities. In October 2017, a former Uber executive paid $100 000 in Bitcoin to hackers for concealing a data breach involving 57 million accounts[9]. In June 2018, Acfun announced an attack by hackers and the leak of nearly 10 million pieces of user data[10]. In July 2018, the healthcare records of 1.5 million citizens of Singapore were leaked, including the personal data of prime minister Lee Hsienloong. New York-based entrepreneur and cryptocurrency investor Michael Terpin sued telecom operator AT&T (based in the United States) for fraud and gross negligence that led to the loss of cryptocurrency in personal accounts; the claim was set to $224 million[11]. These destructive security vulnerabilities remind us of the necessity of IA in data sharing; however, the current centralized authentication systems face severe challenges.

Achieving secure authentication in current data-sharing processes is challenging due to the following reasons: (1) Centralized IA systems are vulnerable to single point of failure attacks. Further, an attacker only needs to attack the IA server to threaten the entire IA platform. (2) The privacy of identity data storage is prone to threats: storing user identity data in a local database has an associated risk of privacy leakage. (3) Tracing the identity of both sharing parties is difficult, and in the event of a data breach, it results in difficulties in tracing responsibility. These challenges have been addressed by technologies such as distributed accounting, point-to-point transmission, openness, and transparency of blockchain[12]. To overcome the above challenges, we propose an effective IA based on multiattribute centers for secure data sharing. The main contributions of this study are as follows:

- We design the architecture of multiattribute authorization centers to replace the traditional centralized counterpart for addressing the problem of single point of failure. The corresponding authorization center generates the private key of each attribute for the data requester, who then generates a personal key that is unknown to other third-party organizations, thereby ensuring the security of the key.
- We propose a dynamic key generation algorithm that combines blockchain and smart contracts to periodically update the data requester's key to prevent theft by external attackers. The combination of blockchain and InterPlanetary File System (IPFS) is used to store the identity information of data requesters to achieve trusted data storage and reduce the leakage of private information.
- We propose a NonInteractive Zero-Knowledge Proof (NIZKP) authentication scheme for validating the identity of a data requester. A lightweight elliptic curve cryptography method is adopted to improve authentication efficiency.

The rest of this article is organized as follows: Section 2 briefly reviews related works. Section 3 presents the preliminaries. The system design and main algorithm are discussed in Section 4. Further, Section 5 introduces the blockchain-based IA model. Section 6

presents the experimental results and main findings. Finally, Section 7 summarizes our work.

2 Related Work

In this section, we review previous researches relevant to this article; Table 1 presents the summary.

Traditional IA is mainly based on PKI, wherein specialized authorities are established for issuing and managing digitally issued certificates. Given the security issues of these systems, several researchers have provided solutions.

To reduce the enormous cost of certificate revocation lists in intelligent transportation systems, Asghar et al.[13] proposed a scalable PKI-based vehicle ad-hoc network authentication protocol. Thousands of certificates belonging to revoked vehicles are replaced with a single common link key, thereby achieving efficiency. Marino et al.[14] introduced the PKI architecture of the Internet of Things (IoTs) to achieve certificate-based IA and reduce the consumption of transmission time in the IoTs applications. Qiu et al.[15] proposed a scheme for the intelligent distribution of authentication keys based on PKI and edge computing in vehicle to everything networks. The key is pre-distributed at the future location of the vehicle, reducing the delay caused by the vehicle request. Arm et al.[16] proposed offline vehicle access through PKI-based IA to solve the problem of access control in shared cycling. The scheme does not require physical keys to ensure the reliable and secure operation of critical control management.

The schemes presented in the aforementioned article still have certain problems that need to be addressed. A fully trusted key distribution center dispenses keys in the system architecture design. This partially centralized structure presents potential threats and does not consider the problem of the abuse of power. The emergence of blockchain provides a convenient platform for data protection. In the blockchain, miners cooperate to create blocks as a publicly distributed ledger for verifying and recording transactions, providing new ideas for addressing issues related to centralized certification centers[17].

Table 1 Summary of performance metrics discussed in this article.

Reference	Unlinkability	Tamper resistant	Workload	Resistant to external attacks
[13–17]	No	No	No	No
[18–26]	Yes	Yes	No	No
[27–29]	Yes	Yes	Yes	No
Our work	Yes	Yes	Yes	Yes

Feng et al.[18] proposed a 5G data-sharing model based on blockchain to address the security concerns of unmanned aerial vehicles in an open and untrusted environment. A smart contract is used for IA and access control to prevent invasion by malicious users. Guo et al.[19] established a secure and trusted access system for information isolation between different platforms by combining blockchain and edge computing to ensure traceability of activities while achieving trusted authentication, good fault tolerance, and anti-attack ability. Barnawi et al.[20] proposed a blockchain-based demand response management method by analyzing the privacy of information exchange between different entities in the vehicle-to-grid environment, in which the miner node is responsible for IA. This scheme can effectively reduce the transmission delay but does not consider large-scale decentralized energy transactions. Garg et al.[21] designed a key agreement protocol based on mutual authentication by fully hashing the Menezes-Qu-Vanstone key exchange mechanism combined with an elliptic curve cryptosystem, which can effectively resist denial of service and replay attacks. Thus, communication costs and expenses are considerably reduced. Shin et al.[22] proposed a distributed privacy protection scheme to solve the high dependency problem of traditional centralized key management centers. This solution provides unlinkability to protect the anonymity of users with blind signatures. In addition, it provides a pseudonym update or revocation function by counting bloom filters or revoked pseudonym bloom filters, and supports the widely used low-overhead revocation. Liu et al.[23] proposed an efficient and lightweight authentication protocol based on a certificate-less signature, which ensures that a service provider has no privilege to disclose the user's identity. Kumar and Chand[24] proposed an identity-based anonymous authentication and key agreement protocol to achieve mutual authentication and user anonymity, and leveraged cloud technology to increase storage capacity. Jegadeesan et al.[25] proposed a secure and effective privacy-preserving two-way anonymous authentication scheme wherein users are provided with data security and privacy protection at reduced computational and communication costs, and user

misbehavior in the system is tracked through various mechanisms. Jia et al.[26] proposed an identity-based anonymous-authentication key agreement protocol that is suitable for mobile edge computing environments wherein user identity information is protected, and users can access multiple mobile edge computing servers with a single registration.

Blockchain system has limited data content on the chain and requires substantial local storage space, which is unsuitable for storing large-scale data. Therefore, many scholars aim to improve the storage infrastructure.

Zarour et al.[27] proposed a storage architecture combining blockchain and IPFS to solve the problem of centralized data storage in healthcare. Chai et al.[28] proposed CyberChain, an authentication blockchain architecture based on Internet CyberTwin. In addition to protecting the private information of vehicles and improving the efficiency of network communication, CyberChain reduces the storage cost and addresses the security and privacy related concerns of traditional authentication that relies on centralized servers. However, the optimal development and migration strategy of the CyberChain network framework require further elaboration. Jayabalan and Jeyanthi[29] proposed a blockchain-based offline IPFS framework to realize a fail-safe and tamper-proof medical distributed ledger in healthcare, achieving desired robustness.

The aforementioned articles used blockchain technology to realize IA and considered storage efficiency, which solve the workload problem to a certain extent. However, external attacks during IA process, such as those on the private key of the data requester, are ignored to varying degrees.

In summary, herein, we use blockchain technology to ensure the unlinkability and tamper-proofing of the IA process and simultaneously ensure the traceability of identity identification. The difference between the adoption of single and multiattribute authorization centers is that the latter ensures that the user's private key is jointly authenticated by at least t of n authorization centers. In addition, the scheme can resist the collusion attacks from $t-1$ authorization centers (at most) owing to its high security. The relevant data are stored in the distributed IPFS to improve storage efficiency. Herein, we mainly focus on the dynamic management of identity information and setting a delay time for the public key of a data requester. Upon expiration, the public and corresponding private keys of a data requester become invalid, which can help counter theft and other malicious behavior to a certain extent. Our work focuses on ensuring the security of user identity information while improving authentication efficiency.

3 Preliminary

3.1 Blockchain

Blockchain is a chain structure that links blocks in chronological order and ensures that the content of the blocks cannot be forged in a cryptographic manner[30]. Blockchain has become the primary tool for many researchers to store and protect personally identifiable information due to its distributed, tamper-proof, and traceable nature[31]. Security and privacy provided by blockchain technology reduce the risk of user data leakage and large-scale data loss due to attacks on centralized devices.

3.2 IPFS

IPFS is a point-to-point distributed file system that can overcome the weaknesses of existing protocols in centralized systems for retrieving and sharing data. IPFS uniquely identifies each file by providing content addressing. Specifically, IPFS calculates a unique hash value based on the stored content and returns it to the user as the address of their stored data. The combination of blockchain and IPFS has presented solutions to the problem of complex data storage.

3.3 Bilinear pairing

Let G and G_T be two multiplicative cyclic groups with a large prime number p, and g is the generator of group G. The bilinear pair $e: G \times G \to G_T$ is a map that satisfies the following properties:

- **Bilinear.** For any $a, b \in Z_p$, there is $e(g^a, g^b) = e(g, g)^{ab}$, where Z_p is a multplicative group of nonzero integers modulo p, g^a and g^b represent the a-th and b-th power values of g, respectively.
- **Non-degenerate.** Given $g^c, g^d \in G$, therefore $e(g^c, g^d) \neq 1_{G_T}$, where 1_{G_T} represents the identity element of G_T.
- **Computability.** A valid algorithm exists for any $a, b \in Z_p$, which can calculate $e(g^a, g^b)$.

3.4 Distributed key generation

Distributed key generation technology is central to the

threshold cryptosystem, which calculates the shared public and private key set through multiparty participation. Distributed key generation does not need to rely on any trusted third party, and its core idea is (N, T) threshold secret sharing[32] among a group of participants, where N represents the total number of participants, and T represents the minimum number of participants needed for cooperation. Given that each participant has some information about the secret, threshold secret sharing allows N participants to jointly generate keys. The secret value can be reconstructed when there are more than T participants, but less than T participants cannot obtain secret-related information.

3.5 Elliptic Curve Cryptography (ECC)

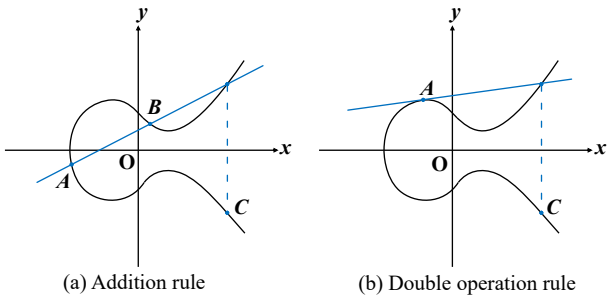
The ECC algorithm is an asymmetric encryption algorithm based on the mathematical theory of the elliptic curve. In general, this study discusses an elliptic curve with a binary cubic equation, which has many forms. In the elliptic curve cryptosystem, the most commonly used is the Weierstrass general formula, as stated in the following:

$$E = \left\{(x,y) \in \mathbf{R}^2 | y^2 = x^3 + ax + b,\ 4a^3 + 27b^2 \neq 0\right\} \quad (1)$$

An elliptic curve E defined on the domain F is a set of points $(x, y) \in F \times F$ that satisfies Eq. (1)

Addition rule: Draw a straight line through two points, A and B, on the curve to find their intersection points. The intersection point that is symmetric about the x-axis is defined as C, that is, $A + B = C, C \in E$, as shown in Fig. 2a.

Double operation rule: When the two points coincide, the tangent line of the elliptic curve at point A and the intersection of the elliptic curve are at one point. The intersection point that is symmetric about the x-axis is defined as $A + A$; that is, $A + A = C$, as shown in Fig. 2b.



(a) Addition rule (b) Double operation rule

Fig. 2 Addition rule and double operation rule on an elliptic curve.

3.6 Elliptic Curve Discrete Logarithm Problem (ECDLP)

The ECDLP is defined as follows: Given a prime number p and an elliptic curve E satisfying Eq. (1), P and Q are points on E. For $Q = KP$, find a positive integer K smaller than p when P and Q are known. Calculating Q is easier and faster if K and P are known, but calculating K is more challenging if Q and P are known. No effective method is available to solve this problem, which is the principle of the ECC algorithm.

The elliptic curve algorithm is a strong competitor to replace RSA, which was proposed by Ronald L. Rivest, Adi Shamir, and Leonard Adleman in 1978[33]. RSA is composed of the initial letters of their surnames. With the increase in security level, the key length of current encryption methods is expected to increase exponentially, while the ECC key length only increases linearly. For example, the 128-bit security encryption requires a 3072-bit RSA key but only a 256-bit ECC key. The small computation, fast processing, and low bandwidth requirements enable ECC to have a wide range of application prospects.

3.7 NIZKP

Zero-Knowledge Proof (ZKP) is the ability of the prover to convince the verifier that a certain assertion is correct without providing any valuable information. This system requires the prover and verifier to exchange information within a specified round and is called the interactive ZKP. Each interaction includes a promise, a challenge, and a response. However, several situations are unsuitable for such interaction. In this case, the NIZKP is generally adopted. The challenge and response are non-interactive, and the prover only needs to send a message to the verifier. ZKP needs to meet the following three characteristics:

- **Completeness:** If the prover and verifier are honest and follow each step of the proof process to perform correct calculations, then the proof must be successful, and the verifier must be able to accept the prover.
- **Soundness:** The prover cannot pass the verification step without secret knowledge.
- **Zero-knowledge:** After the proof is completed, the verifier can only know that the prover possesses secret knowledge and has no other information regarding the knowledge. That is, the prover does not reveal any information about the secret knowledge.

4 Proposed Model

4.1 System overview

Our distributed IA model is shown in Fig. 3. The two sharing parties are the data owner and requester. The former verifies the legitimacy of the latter's identity for data sharing. The head of each governmental department then acts as a representative for interdepartmental data sharing.

First, the data requester uses multiple identity attributes, such as name, employee number, and department number, as attribute set to generate attribute private keys. The data requester sends an attribute key generation request to at least t out of n authorization centers. Upon receiving the request, the authorization centers first verify the legitimacy of the identity attribute information of the data requester. Following verification, the attribute set submitted by the data requester is stored in IPFS, and the corresponding private key of attribute is generated and sent to the data requester. Second, after receiving a part of the private key of attribute from the authorization centers, the data requester generates its locally paired public-private key and uploads the public key to the blockchain. Finally, the data owner verifies the legitimacy of the data requester's identity through the IA scheme based on blockchain and NIZKP. After successful IA, the data requester and owner share data.

The entities and terms involved in the model are as follows:

- **Data requester:** The person in charge of a governmental department who requests access to the data of other departments. The data requester can query the appropriate data owner through a smart contract and prove the legitimacy of his/her identity and then request the corresponding access.
- **Data owner:** The person in charge of a governmental department who provides data to other departments. The data owner is responsible for verifying the legitimacy of the data requester's identity, which is assumed to be trustworthy in this study.
- **Attribute set:** The data requester composes an attribute set according to multiattribute factors such as name, employee number, and department number, which are used to generate attribute keys to address the issue of inflexible single-factor authentication.
- **Blockchain:** The data owner verifies the legitimacy of the data requester through the blockchain. Storing data in the blockchain incurs transaction costs and therefore is unsuitable for storing large amounts of data. Accordingly, we store hash values of data in the system for cost reduction.

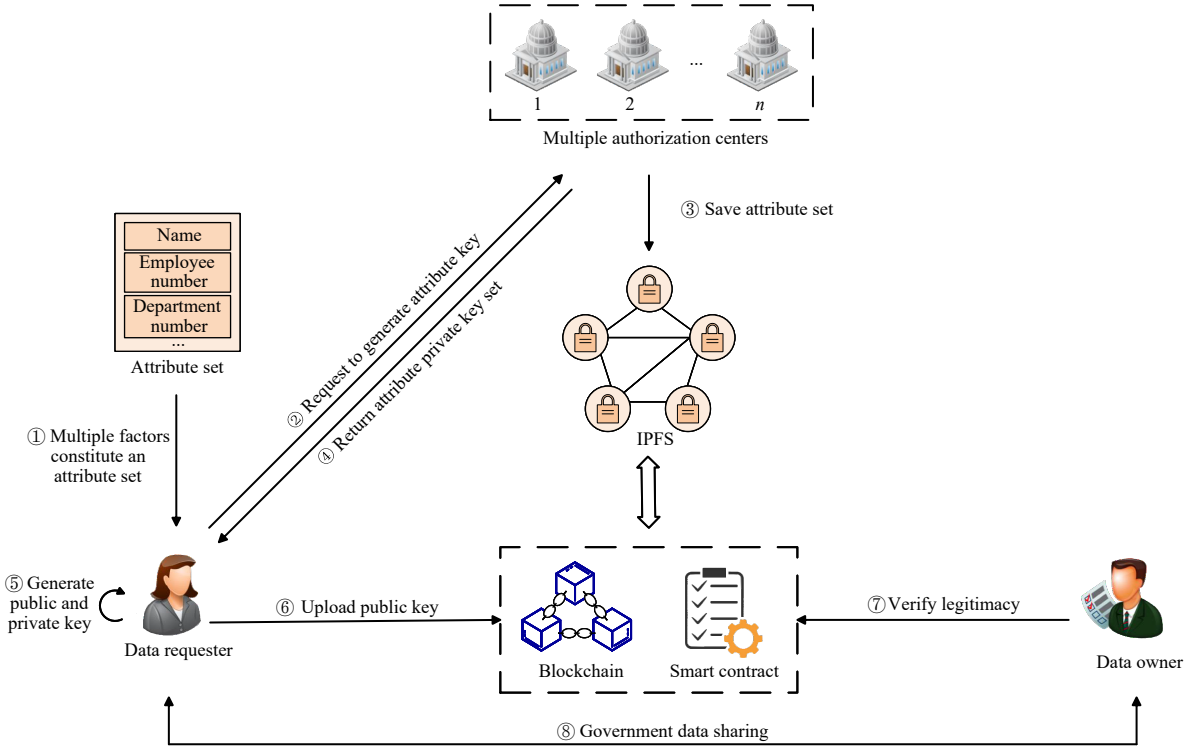


Fig. 3 Identity authentication model using multiattribute authorization centers.

• **Smart contract:** This program is combined with blockchain technology to manage the time validity of the data requester's keys.

• **IPFS:** It stores the attribute information of the data requester and uses the hash function to calculate a unique hash value for each attribute. IPFS is combined with blockchain to store identity information, ensure data security, and reduce storage costs.

• **Authorization center:** The proposed scheme employs multiple attribute authorization centers that are responsible for generating corresponding attribute keys for the data requester.

• **Government data sharing:** The data owner shares their information with the data requester, who uses the information to meet the department's needs.

4.2 Threat model

We consider the threat models in the authentication, which are divided into the following categories.

4.2.1 Identity data confidentiality threat

In practical application scenarios, attackers capture ciphertext information to collect sufficient private information for associating with real identities.

4.2.2 Identity data forgery threat

In the information transmission, the attacker may steal the private key of attribute information and then impersonate the legitimate data requester.

5 Identity Authentication Based on Blockchain

The scheme consists of three phases: initialization, key generation, and IA. The overall content framework is shown in Fig. 4.

The specific schemes of the three phases are defined below. Table 2 lists the main symbols used in this study.

5.1 Initialization phase

For the system to operate normally, initialization is required. The participants here include multiple attribute authorization centers.

The initialization phase is shown in Fig. 5. The details are as follows.

5.1.1 Generation of system public parameter

The authorization center generates prime order p according to safety parameter λ. Choose two multiplicative cyclic groups, G and G_T, with prime order p; g is the generator of the multiplicative cyclic group G, and the bilinear mapping is $e: G \times G \to G_T$.

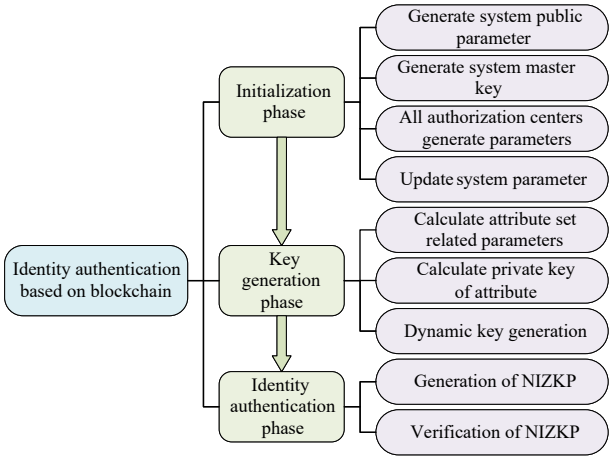


Fig. 4 Content framework of the three phases of identity authentication based on blockchain.

Table 2 List of main symbols.

Symbol	Description
a	Number of attribute elements
params	System public parameter
k	Attribute threshold
n	Number of authorization centers
t	Number of thresholds for user attribute key generation
P_i	Authorization center
s	System master key
Y	System master public key
d_u	Private key for each attribute
K_u	Private key of the data requester
G'	Generator
δ	Generator of the challenge
π	Response
$\text{pac}_{\text{nizkp}}$	NIZKP package
D_t	Delay time
Υ	Set of public keys of data requester and delay time

Define the full set of attributes as I, and $a-1$ attribute elements. Choose the hash function as $H: \{0,1\}^* \to G$. The system public parameter is params $= \{p, g, G, G_T, H, I, a, k, n, t\}$, and uploads it to the blockchain. Among them, the attribute threshold value is $k \in [1, a]$, n is the number of authorized centers, and t is the number of thresholds for user attribute key generation. This study assumes that authoritative authorization centers are trusted and do not mutually authenticate with other centers.

5.1.2 Generation of system master key

Step 1: Authorization center P_i selects the polynomial $f_i(x) = c_{i0} + c_{i1}x + \cdots + c_{i(t-1)}x^{t-1}$ of order $t-1$ and then P_i calculate $C_{ik} = g^{c_{ik}} \pmod p$, where $k = 0, 1, \ldots, t-1$. Then P_i broadcasts C_{ik} to other authorization centers.

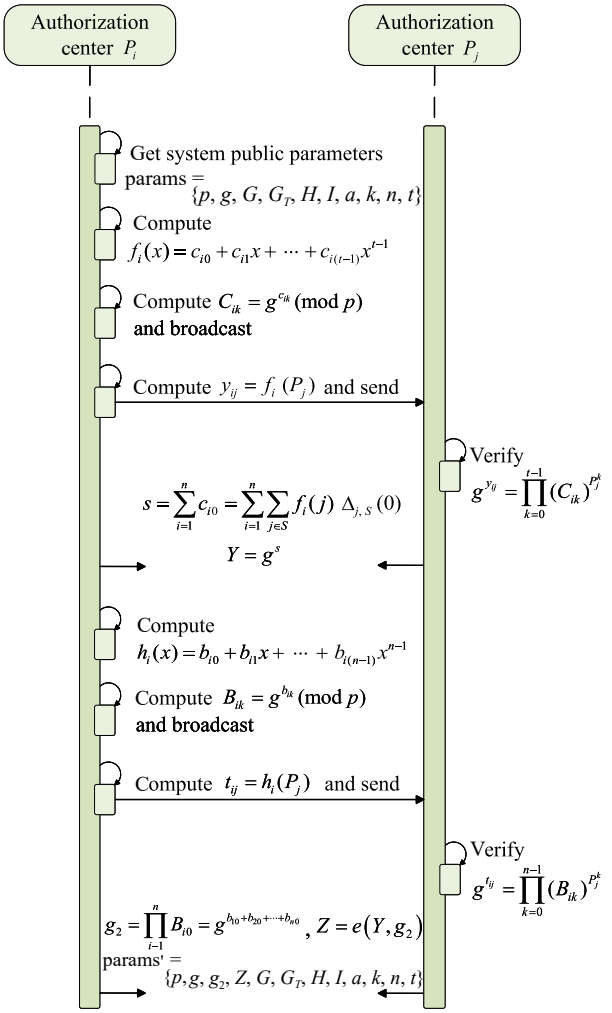


Fig. 5 Initialization phase.

P_i calculates the secret value $y_{ij} = f_i(P_j)$, where $j = 1, 2, \ldots, n$, and sends it to the authorization center P_j, where $j \neq i$.

Step 2: P_j verifies whether the following equation holds:

$$g^{y_{ij}} = \prod_{k=0}^{t-1}(C_{ik})^{P_j^k} \tag{2}$$

If true, authorization center P_j considers P_i to be the honest authorization center; otherwise, P_j requires P_i to rebroadcast y_{ij}.

Step 3: Therefore, the system master key can be obtained as

$$s = \sum_{i=1}^{n} c_{i0} = \sum_{i=1}^{n}\sum_{j\in S} f_i(j)\,\Delta_{j,S}(0) \tag{3}$$

where S represents t centers participating in the key generation. Thus, the system master public key $Y = g^s$ is obtained.

5.1.3 Calculation of parameter $g_2 \in G$ according to params by the authorization centers

Step 1: P_i selects the polynomial $h_i(x) = b_{i0} + b_{i1}x + \cdots + b_{i(n-1)}x^{n-1}$ of order $n-1$, and then calculates $B_{ik} = g^{b_{ik}} (\mathrm{mod}\, p)$, where $k = 0, 1, \ldots, n-1$. Then P_i broadcasts B_{ik} to other authorization centers and calculates the secret value $t_{ij} = h_i(P_j)$, where $j = 1, 2, \ldots, n$. Then, t_{ij} is sent to P_j, where $i \neq j$.

Step 2: P_j verifies whether the following equation holds:

$$g^{t_{ij}} = \prod_{k=0}^{n-1}(B_{ik})^{P_j^k} \tag{4}$$

If the above is true, then authorization center P_j considers P_i to be the honest authorization center; otherwise, P_j requests P_i to rebroadcast t_{ij}.

Step 3: After the above interaction, each authorization center can calculate the parameter,

$$g_2 = \prod_{i-1}^{n} B_{i0} = g^{b_{10} + b_{20} + \cdots + b_{n0}} \tag{5}$$

Then, the parameter $Z = e(Y, g_2)$ is generated.

5.1.4 System parameter update

The system parameter params′ = $\{p, g, g_2, Z, G, G_T, H, I, a, k, n, t\}$ is updated.

5.2 Key generation

The purpose is to generate the corresponding private key of attribute d_u for the attribute $u \in I$. Participants include multiple attribute authorization centers and data requesters. The specific steps are shown in Fig. 6, and the details are as follows.

5.2.1 Calculation of parameters related to attribute set

For each $u \in I$, P_i chooses a random value $r_{ij} \in Z_p$ and calculates

$$d_{u1}^{(i)} = g^{r_{iu}} \tag{6}$$

$$d_{u0}^{(i)} = g_2^{f_i(u)\,\Delta_{i,S}(u)} H(u)^{r_{iu}} \tag{7}$$

and then $d_{u0}^{(i)}$ and $d_{u1}^{(i)}$ are securely sent to the data requester.

5.2.2 Calculation of private key of attribute

After receiving the partial key from t authorization centers, the data requester calculates its private key of attribute as follows:

$$d_{u0} = \prod_{i=1}^{t} d_{j0}^{(i)} = g_2^{\sum_{i=1}^{t} f_i(j)\,\Delta_{i,S}(u)} H(j)^{\sum_{i=1}^{t} r_{iu}} \tag{8}$$

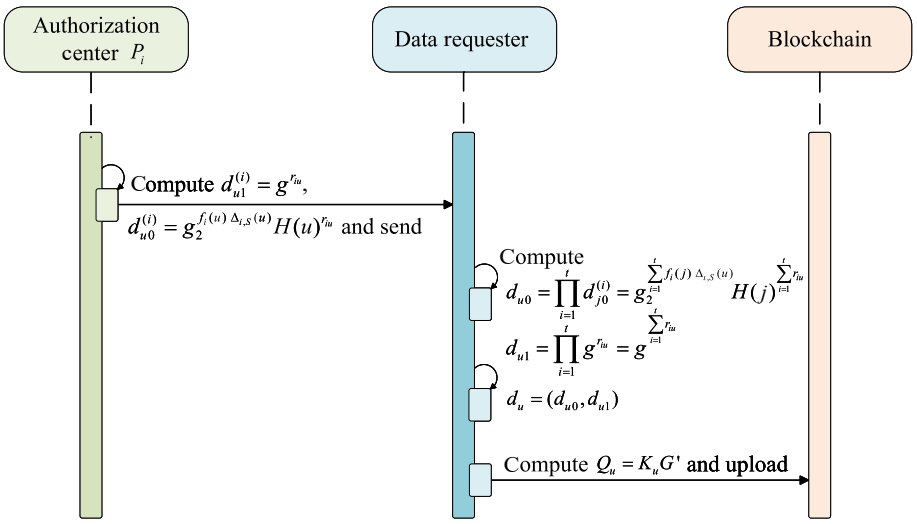


Fig. 6 Key generation.

$$d_{u1} = \prod_{i=1}^{t} g^{r_{iu}} = g^{\sum_{i=1}^{t} r_{iu}} \tag{9}$$

5.2.3 Dynamic key generation

Thus, the private key $d_u = (d_{u0}, d_{u1})$ of each attribute $j \in I'$ is obtained. After obtaining the private key for each attribute, the data requester generates a personal private key K_u. At the same time, the elliptic curve is used to calculate $Q_u = K_u G'$ to generate its public key Q_u, $Q_u \in E(F_p)$, where $E(F_p)$ represents the elliptic curve E defined over the finite field F_p, and G' is a generator of the eliptic curve E. The data requester uploads the public key information to the blockchain. To prevent the private key of the data requester from being stolen by the attacker, a dynamic key generation algorithm is designed, as shown in Algorithm 1.

Algorithm 1 Dynamic key generation algorithm

Input: Public key information of the data requester Q_u
Output: Public key status of the data requester R

```
Define a transaction that will perform delete operation after D_t seconds;
Add D_t to Q_u;
Add new element Υ = {Q_u, D_t} to blockchain;
Data owner queries Υ = {Q_u, D_t} from the blockchain;
if Υ exists then
    Public key information Q_u is valid;
    return 1;
else
    Public key information Q_u is invalid;
    return 0;
    Enter key generation phase;
end if
```

In the dynamic key generation algorithm, the concept of delay time is introduced, which is recorded as D_t, and in this algorithm, we define a smart contract. The delay time D_t is equivalent to the validity period of the public key. The smart contract records the public key Q_u and the delay time D_t of the data requester, which is recorded as $\Upsilon = \{Q_u, D_t\}$. $R = (0, 1)$ represents the result of IA, where 1 represents that the public key is valid, 0 otherwise.

When the set time comes, the smart contract automatically executes an operation to revoke the public key without human intervention, thereby making it invalid. Specifically, after D_t, the smart contract automatically deletes Υ, indicating that the public key of the data requester is invalid. Deleting a public key does not actually modify the data within the block in the blockchain, but rather indicates that Υ has been invalidated by initiating a new transaction. The underlying blockchain retains historical information, which is publicly accessible. Once the data owner queries the blockchain for Υ invalid transactions, the data requester cannot be authenticated, and needs to re-execute the key generation phase to prevent external attackers from stealing the key. Even if a malicious attacker succeeds in stealing the data requester's private key, the intercepted information is unusable after the key is dynamically changed. Thus, the loss of the data requester is minimal.

In this article, the blockchain maintains the chronergy of the data requester's public key. The entire process is performed autonomously by the smart contract, and the public key update is not artificially intervened. Therefore, the reliability of chronergy

management can be improved through the blockchain.

5.3 IA phase

The processes of this phase are divided into the generation and verification of NIZKP. The specific steps and details are shown in Fig. 7.

5.3.1 Generation of NIZKP

The purpose is to verify the data requester and perform data transmission. The participants are, therefore, the data requester and data owner, acting as prover and verifier, respectively. The input includes the elliptic curve $E(F_p)$, generator $G' \in E(F_p)$, and the public key Q_u of the data requester.

Step 1: The data requester randomly selects an integer $\upsilon \in F_p$ and calculates the point $M = \upsilon G'$.

Step 2: The data requester uses a cryptographic hash function H to calculate the challenge δ, such as $\delta = H(G' \| Q_u \| M)$.

Step 3: The data requester calculates the response π to the challenge δ, such as $\pi = \upsilon + \delta \cdot K_u (\mathrm{mod}\ p)$.

Step 4: The data requester generates a package $\mathrm{pac}_{\mathrm{nizkp}}$ including NIZKP. $\mathrm{pac}_{\mathrm{nizkp}}$ contains the following information: point M calculated in Step 1 and response π generated in Step 3.

Step 5: The data requester sends the package $\mathrm{pac}_{\mathrm{nizkp}}$ to the data owner.

5.3.2 Verification of NIZKP

This phase aims to receive NIZKP and verify the legitimacy of the data requester. The participants are the data requester and owner, acting as prover and verifier, respectively. The input includes the elliptic curve $E(F_p)$, generator $G' \in E(F_p)$, public key Q_u of the data requester, and the package $\mathrm{pac}_{\mathrm{nizkp}}$.

Step 1: The data owner receives the package $\mathrm{pac}_{\mathrm{nizkp}}$, including point M and response π.

Step 2: The data owner uses the blockchain to check whether Q_u is from a registered user. If so, then the data owner proceeds to Step 3; otherwise, the execution terminates.

Step 3: The data owner calculates the challenge δ using the public key Q_u, as calculated $\delta = H(G' \| Q_u \| M)$ by the data requester.

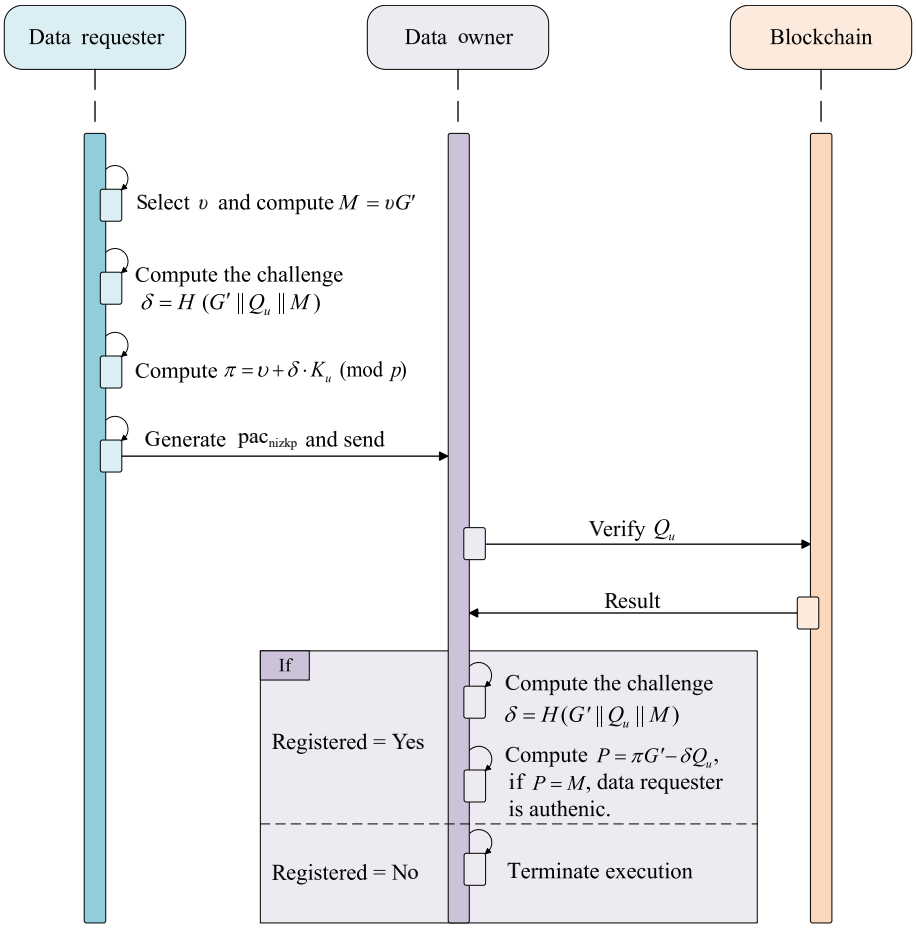


Fig. 7 Authentication process.

Step 4: The data owner calculates a point $P = \pi G' - \delta Q_u$ and checks if $P = M$ holds. If $P = M$ is true, then the data requester is a legitimate user and has been authenticated. Otherwise, IA fails, and execution terminates.

Step 5: After the IA is successful, the data requester and the data owner can transfer corresponding data to achieve the purpose of data sharing.

6 Experimental Result

6.1 Security analysis

6.1.1 Correctness analysis

When authenticating the data requester, the data owner obtains the relevant parameters from the packet $\text{pac}_{\text{nizkp}}$ sent by the data requester and proves whether the equation $P = M$ is valid, thereby verifying the identity legitimacy,

$$\begin{aligned} P &= \pi G' - \delta Q_u = \\ &[\upsilon + \delta \cdot K_u \,(\bmod p)] G' - \delta Q_u = \\ &\upsilon G' + \delta \cdot K_u \,(\bmod p)\, G' - \delta Q_u = \\ &M + \delta Q_u - \delta Q_u = M \end{aligned}$$

From the above equation, $P = M$ is true.

6.1.2 Resistance to single point of failure

The overall design of the scheme is based on blockchain and IPFS technology, and the decentralized and distributed characteristics of the scheme ensure that the scheme remains unaffected even if a node fails. In addition, multiattribute authorization centers are established to realize attribute management. The security decentralization of attribute management authority not only effectively counters illegal operations arising from the centralized authority, but also reduces the potential message leakages arising from the failure of a single authorization center. Therefore, the proposed scheme can effectively avoid the problem of a single point of failure.

6.1.3 Resistance to external attacks

The multiattribute authorization center differs from the single counterpart in that the data requester needs to request the private key of attribute from t authorization centers and locally generate the personal private key. This scheme can resist collusion attacks from $t-1$ authorization centers (at most) and is highly secure. In addition, herein, the following two attack cases are presupposed for security analysis of the authentication model based on multiattribute authorization centers.

(1) An adversary obtains a data requester's private key of attribute by attacking an authorization center.

Assuming that the adversary successfully infiltrates an attribute authorization center and obtains the private key of attribute, it cannot synthesize the personal private key of the data requester and, therefore, cannot pass the authentication. Given that t attribute private keys generate the personal private key of the data requester, the attack difficulty is t times higher than that of the system that provides a public-private key pair through only a single authorization center. Thus, even if the private key of attribute of an attribute authorization center is leaked, the proposed scheme can ensure reliable IA of the data requester.

(2) An adversary obtains the personal private key by attacking the data requester.

Assuming that this is achieved through illegal means, the adversary can obtain relevant information to pass the authentication. To address this issue, this study proposes a dynamic key generation algorithm. When the set time D_t is reached, the smart contract automatically generates new transactions to invalidate Υ and uploads it to the blockchain without human intervention. Thus, the adversary cannot pass the authentication when the data owner queries the transaction. The cost of obtaining the personal private key by the requester of the attack data is high. After time D_t, the personal private key stolen by the adversary will become invalid. Alternatively, the adversary may try to obtain the private key by cracking the public key of the data requester. The public-private key pair is generated based on ECC, and the security of the key is based on the difficulty of solving ECDLP, which requires parsing to compromise the authentication scheme. However, this is computationally difficult, and no algorithm can solve ECDLP in polynomial time. Therefore, the proposed scheme can ensure reliable data requester authentication.

6.2 Performance analysis of the IA scheme

In this section, we present the performance analysis results of the proposed IA scheme and other similar advanced schemes, such as that developed by Liu et al.[23], Kumar and Chand[24], Jegadeesan et al.[25], and Jia et al.[26], to compare computational and communication costs.

The experimental environment was established on a Lenovo T430 with Intel (R) Core (TM) i7-7500U 2.90 GHz CPU, 8 GB RAM, and Windows 10 64-bit

operating system on a laptop. Cygwin software's Pairing-Based Cryptography library (PBC)[34] was used to simulate ryptographic operations.

6.2.1 Computational cost

The computational cost is the time required to complete the encryption operation performed for data requester authentication. Herein, 100 experiments were randomly conducted, and the time of all calculations was averaged to obtain the computational cost. The symbols used for the computational comparison and their running times are shown in Table 3.

Table 4 presents the computational time relationships of the analyzed authentication schemes. These relationships can be used to determine the authentication time cost of the aforementioned schemes for comparison with that of the proposed scheme. m represents the number of data requesters in the authentication.

In addition, Fig. 8 provides the computational costs of performing authentication when the aforementioned schemes are employed. The authentication experiment involved 20–100 simultaneous data requesters. As shown in Fig. 8, the computational cost of authentication increases as the number of data requesters increases. Thus, the time to perform authentication is proportional to the number of data requesters involved in the authentication process. Considering 100 data requesters, the values of authentication time obtained are as follows: Liu et al.'s scheme[23]: $m\,T_{bp}+(m+1)\,T_h+(2\,m+1)\,T_e \approx 1065.08$ ms; Kumar et al.'s scheme[24]: $(m+1)\,T_{bp}+(m+1)\,T_h+(2m+1)\,T_{sm} \approx 741.57$ ms; Jegadeesan and Chand's scheme[25]: $(m+1)\,T_{bp}+(m+1)\,T_h+(m+1)\,T_{sm} \approx 518.97$ ms; Jia et al.'s scheme[26]: $m\,T_e+5\,m\,T_h+4\,m\,T_{sm} \approx 1276.55$ ms. The authentication time of the proposed IA scheme is $(m+1)\,T_{bp}+m\,T_h+2\,m\,T_{sm}+m\,T_{pa} \approx 739.44$ ms. Thus, the proposed scheme shows better performance in computational cost compared with analyzed authentication schemes.

Table 3 Cryptographic operations and their running time.

Cryptographic operation	Running time (ms)
Bilinear pairing (T_{bp})	2.9100
Exponential operation (T_e)	3.8500
Hash operation (T_h)	0.0023
Scalar multiplication (T_{sm})	2.2260
Point addition operation (T_{pa})	0.0010

Table 4 Computational time relationship of the analyzed authentication schemes.

Authentication scheme	Authentication time (ms)
Liu et al.[23]	$m\,T_{bp}+(m+1)\,T_h+(2m+1)\,T_e$
Kumar and Chand[24]	$(m+1)\,T_{bp}+(m+1)\,T_h+(2m+1)\,T_{sm}$
Jegadeesan et al.[25]	$(m+1)\,T_{bp}+(m+1)\,T_h+(m+1)\,T_{sm}$
Jia et al.[26]	$m\,T_e+5\,m\,T_h+4\,m\,T_{sm}$
Our scheme	$(m+1)\,T_{bp}+m\,T_h+2\,m\,T_{sm}+m\,T_{pa}$

6.2.2 Communication cost

The communication cost is the number of bits exchanged between the data requester and the owner when interacting during authentication. The results of the communication cost comparison between the analyzed authentication schemes are shown in Fig. 9.

The hash function of the proposed scheme adopts SHA-256. In the generation of NIZKP for the IA

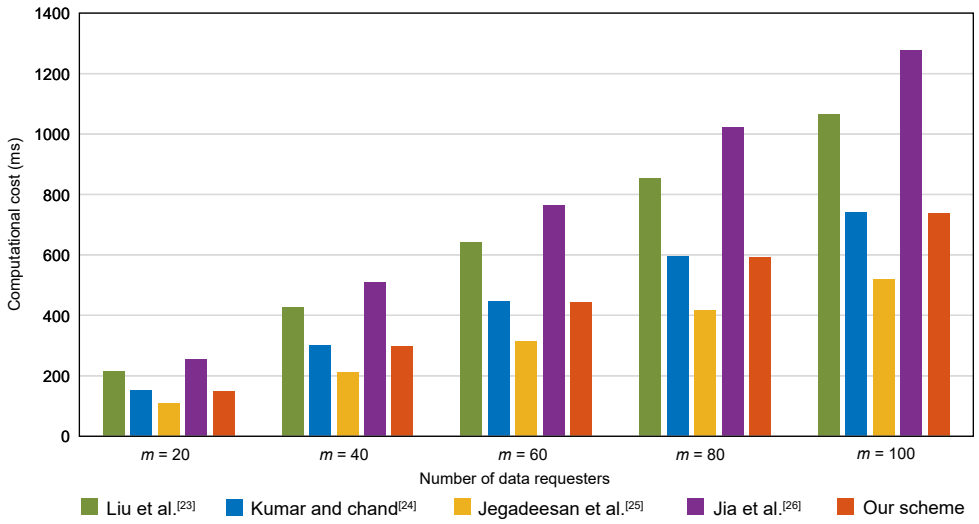


Fig. 8 Comparison of computational costs of various authentication schemes.

phase, the communication cost of operations, such as input, calculation of point M, and the computation of challenge δ of the data requester, is calculated in 256 bits. The same calculation is used in the verification of NIZKP for the IA phase, the communication cost of operations, such as input, verifying the public key pub_{dr} of the data requester, computing challenges δ, and verifing equation.

The communication cost in the authentication phase is calculated and transmitted in 3584 bits. The authentication cost of other schemes presented in Fig. 9 are as follows: Liu et al.'s scheme[23]: 3840 bits; Kumar and Chand's scheme[24]: 5440 bits; Jegadeesan et al.'s scheme[25]: 6048 bits; Jia et al.'s scheme[26]: 4736 bits. Accordingly, the proposed authentication scheme significantly reduces communication costs and presents certain advantages.

6.3 Performance analysis of key generation phase

The effectiveness of the proposed scheme is verified through simulation experiments in MATLAB 2018b with two perspectives. First, the time when the attribute authorization center generates the private key of attribute is compared; the result is shown in Fig. 10. Second, the time when the data requester obtains the private key is compared; the result is shown in Fig. 11. The interaction time between the data requester and the authorization center is not considered.

Figure 10 shows the time spent by a single authorization center and multiattribute authorization centers to synthesize a private key as the number of attributes of the data requester increases. Regardless of the number of authorization centers, the time to generate the private key gradually increases with the increasing number of attributes of the data requester. However, the total time required by multiattribute

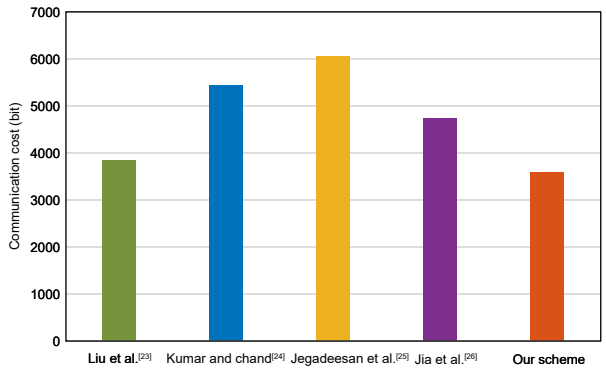


Fig. 9 Comparison of communication costs of various authentication schemes.

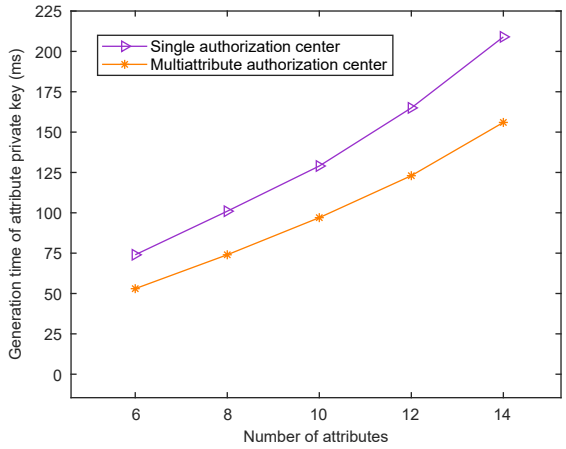


Fig. 10 Relationship between the number of attributes and generation time of the private key of attribute.

authorization centers is shorter than that of a single authorization center. This is because in the case of a single authorization center, all attributes of the data requester are sent as a whole, and the private key of attribute returned to the data requester by the authorization center is the personal private key. However, in the case of multiattribute authorization centers, multiple attributes of the data requester are sent to different authorization centers, each of which is responsible for an attribute that does not intersect with another. Multiple authorization centers generate the corresponding private keys of attribute of the data requester in parallel, thereby saving time.

Figure 11 shows the variation in the time when the data requester generates the personal private key with an increasing number of attributes in the case of a single authorization center and multiattribute authorization centers. As the number of attributes of the data requester increases, the time for the data requester to generate the personal private key by a single authorization center is always 0, while that in the case of multiattribute authorization centers is constantly increasing. We believe that in a single authorization center, the private key of attribute is the personal private key of the data requester, who therefore does not need to spend extra time to generate the personal private key. In the case of the multiattribute authorization centers, in addition to obtaining the private key of attribute, the data requester also needs to use multiple attribute private keys to generate a personal private key, which takes a certain amount of time. Therefore, the time of the proposed scheme is slightly higher than that in the single

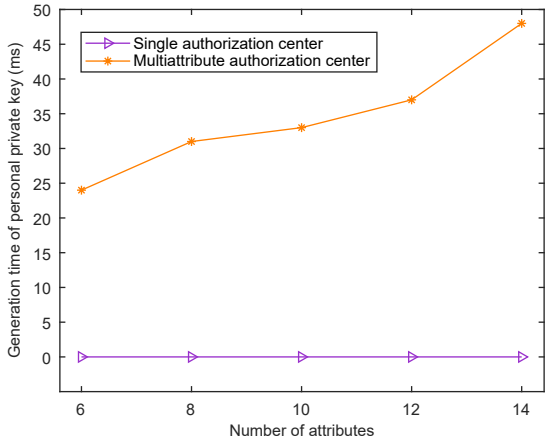


Fig. 11 Relationship between the number of attributes and the time when the data requester obtains the personal private key.

authorization center.

Figure 12 shows the key generation phase's total time, including the generation of the private key of attribute by the authorization center and the personal private key by the data requester. When the number of attributes of the data requester is less than eight, the time spent in the key generation phase of the proposed scheme of the multiattribute authorization center is slightly more than that of the single authorization center. However, as the number of attributes increases, the time spent in the key generation phase of the proposed scheme outperforms that of the single authorization center. Although the time for the data requester to generate the personal private key increases, this is a self-generation. No authorization center or third party can obtain the private key, which has high security. Thus, we consider that the time a data requester takes to generate a personal private key is worthwhile and acceptable.

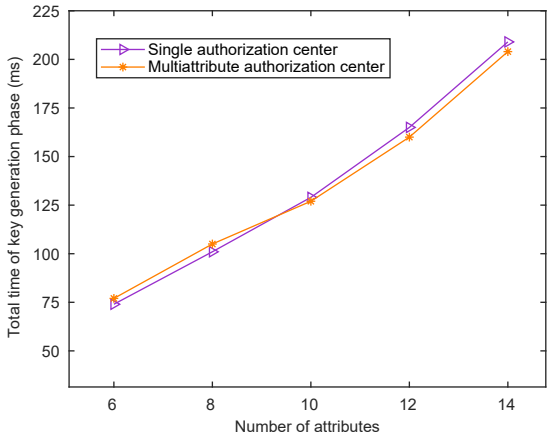


Fig. 12 Total time of the key generation phase.

6.4 Performance analysis of the dynamic key generation algorithm

Enterprise Operation System (EOS) is used to build a blockchain simulation environment for chronergy management and implement a mechanism called delayed transaction, which can suspend a specified time before execution without human intervention. At the same time, compared with other blockchain projects, EOS can support millions of users and has strong parallel execution capabilities. The parameters used to build a blockchain simulation environment based on EOS, such as the content size and version number of the environment configuration[35], are shown in Table 5.

A total of 1000 data requesters are simulated, and the results are shown in Fig. 13. where the abscissa represents the time of the experiment, and the ordinate represents the probability of losing the personal private key of the data requester. Over time, the risk of losing the personal private key of the data requester increases over a six-month period. A dynamic key is added to the proposed scheme, and the public key of the data

Table 5 Blockchain environment configuration.

Hardware environment	Software environment
Elastic compute service	Ubuntu 16.04.6
CPU Intel (R) Core (TM) i7-7500U	Eosio 1.7.0
RAM 2 GB	Eosio.cdt 1.6.1
SATA 40 GB	Eosio.contracts 1.5.2
Bandwidth 1 Mbps	JAVA 1.7
–	JPBC 2.0.0

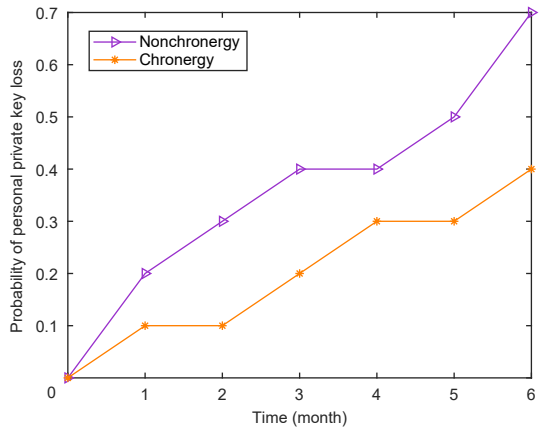


Fig. 13 Performance comparison of public key chronergy management.

requester is updated periodically. Therefore, the private key corresponding to the public key is more secure than in the scheme without a dynamic update.

The public key information of the data requester is stored in the blockchain, and the security of the public key information is guaranteed based on its immutability. In addition, given that the chronergy of the data requester's public key information is maintained by the blockchain, no additional computing resources are necessary.

7 Conclusion

In this study, the identity legitimacy of government data requesters was considered to ensure the security of data sharing among governmental departments. We propose an efficient IA scheme based on multiattribute authorization centers. In this scheme, although the data requester needs to spend a certain amount of time to generate a personal private key, the key escrow problem can be solved, and security can be ensured. Further, a dynamic key generation algorithm is proposed wherein the public key of the data requester is configured to be dynamically updated, and the authentication mechanism is deployed through NIZKP. Simulation experiments demonstrated the effectiveness of the proposed scheme. Compared with similar authentication schemes, the proposed scheme performs better in terms of computational and communication costs. However, due to length limitation, there are other issues worth delving into, such as the existence of malicious nodes in blockchain may lead to consensus security problems and how to share data in the next step. Therefore, our future research will focus on designing a reasonable reward and punishment mechanism for nodes to improve consensus efficiency and formulate an efficient and secure data-sharing scheme.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Nos. 61771289 and 61832012), the Natural Science Foundation of Shandong Province (Nos. ZR2021QF050 and ZR2021MF075), the Shandong Natural Science Foundation Major Basic Research (No. ZR2019ZD10), the Shandong Key Research and Development Program (No. 2019GGX1050), and the Shandong Major Agricultural Application Technology Innovation Project (No. SD2019NJ007).

References

[1] L. Qi, W. Lin, X. Zhang, W. Dou, X. Xu, and J. Chen, A correlation graph based approach for personalized and compatible web APIs recommendation in mobile APP development, *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 6, pp. 5444–5457, 2023.

[2] X. Zhou, W. Liang, W. Li, K. Yan, S. Shimizu, and K. I. K. Wang, Hierarchical adversarial attacks against graph-neural-network-based IoT network intrusion detection system, *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9310–9319, 2022.

[3] P. Zhang, M. Zhou, Q. Zhao, A. Abusorrah, and O. O. Bamasag, A performance-optimized consensus mechanism for consortium blockchains consisting of trust-varying nodes, *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2147–2159, 2021.

[4] Z. Rahman, I. Khalil, X. Yi, and M. Atiquzzaman, Blockchain-based security framework for a critical industry 4. 0 cyber-physical system, *IEEE Commun. Mag.*, vol. 59, no. 5, pp. 128–134, 2021.

[5] W. Diffie and M. Hellman, New directions in cryptography, *IEEE Trans. Inf. Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[6] W. Wang, H. Huang, L. Zhang, Z. Han, C. Qiu, and C. Su, BlockSLAP: Blockchain-based secure and lightweight authentication protocol for smart grid, in *Proc. IEEE 19th Int. Conf. on Trust, Security and Privacy in Computing and Communications* (*TrustCom*), Guangzhou, China, 2020, pp. 1332–1338.

[7] G. Li, X. Ren, J. Wu, W. Ji, H. Yu, J. Cao, and R. Wang, Blockchain-based mobile edge computing system, *Inf. Sci.*, vol. 561, pp. 70–80, 2021.

[8] M. Di Mauro, G. Galatro, M. Longo, F. Postiglione, and M. Tambasco, HASFC: A MANO-compliant framework for availability management of service chains, *IEEE Commun. Mag.*, vol. 59, no. 6, pp. 52–58, 2021.

[9] Y. M. Tseng, J. L. Chen, and S. S. Huang, A lightweight leakage-resilient identity-based mutual authentication and key exchange protocol for resource-limited devices, *Comput. Networks*, vol. 196, p. 108246, 2021.

[10] H. Boche, R. F. Schaefer, S. Baur, and H. V. Poor, On the algorithmic computability of the secret key and authentication capacity under channel, storage, and privacy leakage constraints, *IEEE Trans. Signal Process.*, vol. 67, no. 17, pp. 4636–4648, 2019.

[11] M. A. Khan, I. U. Din, T. Majali, and B. S Kim, A survey of authentication in internet of things-enabled healthcare systems, *Sensors*, vol. 22, no. 23, p. 9089, 2022.

[12] Y. Yu, Y. Li, J. Tian, and J. Liu, Blockchain-based solutions to security and privacy issues in the internet of things, *IEEE Wireless Commun.*, vol. 25, no. 6, pp. 12–18, 2018.

[13] M. Asghar, R. R. M. Doss, and L. Pan, A scalable and efficient PKI based authentication protocol for VANETs, in *Proc. 28th Int. Telecommunication Networks and Applications Conf.* (*ITNAC*), Sydney, Australia, 2018, pp. 1–3.

[14] F. Marino, C. Moiso, and M. Petracca, PKIoT: A public key infrastructure for the internet of things, *Trans. Emerg. Telecommun. Technol.*, vol. 30, no. 10, p. e3681, 2019.
[15] H. Qiu, M. Qiu, and R. Lu, Secure V2X communication network based on intelligent PKI and edge computing, *IEEE Network*, vol. 34, no. 2, pp. 172–178, 2020.
[16] J. Arm, P. Fiedler, and O. Bastan, Offline access to a vehicle via PKI-based authentication, in *Proc. Int. Conf. on Computer Safety, Reliability, and Security*, York, UK, 2021, pp. 76–88.
[17] D. D. F. Maesa and P. Mori, Blockchain 3. 0 applications survey, *J. Parallel Distrib. Comput.*, vol. 138, pp. 99–114, 2020.
[18] C. Feng, K. Yu, A. K. Bashir, Y. D. Al-Otaibi, Y. Lu, S. Chen, and D. Zhang, Efficient and secure data sharing for 5G flying drones: A blockchain-enabled approach, *IEEE Network*, vol. 35, no. 1, pp. 130–137, 2021.
[19] S. Guo, X. Hu, S. Guo, X. Qiu, and F. Qi, Blockchain meets edge computing: A distributed and trusted authentication system, *IEEE Trans. Ind. Inf.*, vol. 16, no. 3, pp. 1972–1983, 2020.
[20] A. Barnawi, S. Aggarwal, N. Kumar, D. M. Alghazzawi, B. Alzahrani, and M. Boulares, Path planning for energy management of smart maritime electric vehicles: A blockchain-based solution, *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 2282–2295, 2023.
[21] S. Garg, K. Kaur, G. Kaddoum, J. J. P. C. Rodrigues, and M. Guizani, Secure and lightweight authentication scheme for smart metering infrastructure in smart grid, *IEEE Trans. Ind. Inf.*, vol. 16, no. 5, pp. 3548–3557, 2020.
[22] J. S. Shin, S. Lee, S. Choi, M. Jo, and S. H. Lee, A new distributed, decentralized privacy-preserving ID registration system, *IEEE Commun. Mag.*, vol. 59, no. 6, pp. 138–144, 2021.
[23] J. Liu, Z. Zhang, X. Chen, and K. S. Kwak, Certificateless remote anonymous authentication schemes for wirelessBody area networks, *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 2, pp. 332–342, 2014.
[24] M. Kumar and S. Chand, A lightweight cloud-assisted identity-based anonymous authentication and key agreement protocol for secure wireless body area network, *IEEE Syst. J.*, vol. 15, no. 2, pp. 2779–2786, 2021.
[25] S. Jegadeesan, M. Azees, N. R. Babu, U. Subramaniam, and J. D. Almakhles, EPAW: Efficient privacy preserving anonymous mutual authentication scheme for wireless body area networks (WBANs), *IEEE Access*, vol. 8, pp. 48576–48586, 2020.
[26] X. Jia, D. He, N. Kumar, and K. K. R. Choo, A provably secure and efficient identity-based anonymous authentication scheme for mobile edge computing, *IEEE Syst. J.*, vol. 14, no. 1, pp. 560–571, 2020.
[27] K. Zarour, O. A. Bounab, Y. Marir, and I. Boumezbeur, Blockchain-based architecture centred patient for decentralised storage and secure sharing health data, *Int. J. Electron. Healthcare.*, vol. 12, no. 2, pp. 170–190, 2022.
[28] H. Chai, S. Leng, J. He, K. Zhang, and B. Cheng, CyberChain: Cybertwin empowered blockchain for lightweight and privacy-preserving authentication in internet of vehicles, *IEEE Trans. Veh. Technol.*, vol. 71, no. 5, pp. 4620–4631, 2022.
[29] J. Jayabalan and N. Jeyanthi, Scalable blockchain model using off-chain IPFS storage for healthcare data security and privacy, *J. Parallel Distrib. Comput.*, vol. 164, pp. 152–167, 2022.
[30] S. Nakamoto, *Bitcoin*: *A Peer-to-Peer Electronic Cash System*, http://bitcoin.org/bitcoin.pdf, 2008.
[31] Q. Feng, D. He, S. Zeadally, M. K. Khan, and N. Kumar, A survey on privacy protection in blockchain system, *J. Network Comput. Appl.*, vol. 126, pp. 45–58, 2019.
[32] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, Secure distributed key generation for discrete-log based cryptosystems, in *Proc. Int. Conf. on the Theory and Applications of Cryptographic Techniques*, Prague, Czech Republic, 1999, pp. 295–310.
[33] R. L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, *Commun. ACM*, vol. 21, no. 2, pp. 120–126, 1978.
[34] Cygwin: Linux environment emulator for windows, http://www.cygwin.com/, 2022.
[35] J. A. Fernandez-Carrasco, T. Egues-Arregui, F. Zola, and R. Orduna-Urrutia, ChronoEOS: Configuration control system based on EOSIO blockchain for on-running forensic analysis, in *Proc. Int. Congress on Blockchain and Applications*, L'Aquila, Italy, 2022, pp. 37–47.

Junhua Wu received the PhD degree from Harbin Engineering University, Harbin, China in 2009. She is currently an associate professor at School of Computer Science, Qufu Normal University, Rizhao, China. She has published more than 40 research articles. Her research interests include IoT, artificial intelligence, cloud computing, and blockchain.

Meiquan Wang received the BEng degree in computer science and technology from Qufu Normal University, Rizhao, China in 2020, where she is currently a master student. Her current research interests include blockchain and privacy protection.

Tongdui Zhang received the MEng degree from China University of Petroleum, China in 2008. He is currently an associate researcher at Science and Technology Innovation Service Institution of Rizhao, Rizhao, China. His research interests include computer technology and technical innovation.

Junhao Wu is currently an undergraduate student in network engineering at Qufu Normal University, Rizhao, China. His research interests include blockchain and privacy protection.

Guangshun Li received the PhD degree from Harbin Engineering University, China in 2008. He is currently a professor at School of Computer Science, Qufu Normal University. He was a visiting scholar at the Hong Kong Polytechnic University in the second half of 2019. His research interests include IoT, artificial intelligence, edge computing, and blockchain.