Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.053950

ARTICLE

Blockchain-Based Certificateless Cross-Domain Authentication Scheme in the Industrial Internet of Things

Zhaobin Li*, Xiantao Liu*, Nan Zhang and Zhanzhen Wei

Department of Electronic and Communication Engineering, Beijing Electronic Science and Technology Institute, Beijing, 100070, China

*Corresponding Authors: Zhaobin Li. Email: lzb@besti.edu.cn; Xiantao Liu. Email: cnzjlxt@126.com

Received: 14 May 2024 Accepted: 26 July 2024 Published: 12 September 2024

ABSTRACT

The Industrial Internet of Things (IIoT) consists of massive devices in different management domains, and the lack of trust among cross-domain entities leads to risks of data security and privacy leakage during information exchange. To address the above challenges, a viable solution that combines Certificateless Public Key Cryptography (CL-PKC) with blockchain technology can be utilized. However, as many existing schemes rely on a single Key Generation Center (KGC), they are prone to problems such as single points of failure and high computational overhead. In this case, this paper proposes a novel blockchain-based certificateless cross-domain authentication scheme, that integrates the threshold secret sharing mechanism without a trusted center, meanwhile, adopts blockchain technology to enable cross-domain entities to authenticate with each other and to negotiate session keys securely. This scheme also supports the dynamic joining and removing of multiple KGCs, ensuring secure and efficient cross-domain authentication and key negotiation. Comparative analysis with other protocols demonstrates that the proposed cross-domain authentication protocol can achieve high security with relatively low computational overhead. Moreover, this paper evaluates the scheme based on Hyperledger Fabric blockchain environment and simulates the performance of the certificateless scheme under different threshold parameters, and the simulation results show that the scheme has high performance.

KEYWORDS

IIoT; blockchain; certificateless; cross-domain authentication

1 Introduction

The Industrial Internet of Things (IIoT) is an industrial ecosystem that utilizes the network interconnection of industrial resources to collect, exchange, and analyze information through the interoperability of systems and data. It aims to adapt to demand-driven environments, allocate resources flexibly, and optimize industrial processes [1]. Compared to the Internet of Things (IoT), IIoT focuses more on connecting originally isolated industrial resources so that it can provide more efficient production services [2].

Typically, IIoT systems consist of numerous devices across various management domains, each with unique security policies and varying access permissions. The cross-domain information

Copyright © 2024 The Authors. Published by Tech Science Press.
This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

interaction greatly increases the potential risks of data security and privacy leakage due to the generation and storage of a large amount of sensitive information [3]. Authentication is an effective measure to ensure the security of IIoT communications, but resource-constrained IIoT devices pose additional challenges to the computing and communication overhead of traditional authentication mechanisms [4]. Therefore, how to design lightweight and efficient cross-domain authentication methods has become an increasingly urgent requirement for IIoT.

Typical authentication is mainly based on Public Key Infrastructure (PKI), in which the Certificate Authority (CA) provides a trusted root for all PKI digital certificates [5]. However, certificate-based authentication schemes not only introduce high certificate management costs but also bring heavy communication and computational overheads [6].

To solve the certificate management problem, Shamir [7] proposed Identity-Based Public Key Cryptography (ID-PKC), in which the user key is generated entirely by the Private Key Generator (PKG). However, PKG has the ability to impersonate all user actions, meaning that ID-PKC has the key escrow problem. In 2003, Al-Riyami et al. [8] proposed Certificateless Public Key Cryptography (CL-PKC), in which the Key Generation Center (KGC) only generates the user's partial private key. The partial private key and the user's secret value combine to form the user's complete private key, thereby solving the key escrow problem.

Blockchain is essentially a distributed ledger with the features of decentralization, tamper resistance, non-repudiation, and traceability. In the area of improving blockchain performance for IIoT, many studies [9,10] have made important contributions to enable secure, efficient, and fair blockchain transaction data processing, while in the field of securing cryptosystems, as the ever-present problems of user revocation and potential risks of public key replacement in certificateless systems, combining certificateless cryptography with blockchain has been regarded as an effective solution by many scholars [11]. However, it is worth noting that the blockchain design in many existing certificateless schemes is incomplete. In certificateless cross-domain authentication applications, certain schemes have significant overheads, making them unsuitable for IIoT devices with limited resources. Furthermore, the issue of KGC's single point of failure in certificateless IIoT applications remains unresolved.

In this paper, a certificateless cross-domain authentication scheme that combines threshold secret sharing and blockchain is designed, which mitigates the single point of failure problem with a flexible multi-KGC system, and provides a reliable guarantee for the cross-domain communication security of IIoT through blockchain.

1.1 Contribution

The main contributions of this paper are summarized as follows:

a) A threshold-based multi-KGC certificateless mechanism is proposed, which solves the single point of failure and trust centralization problems of the KGC, and supports the dynamic joining and removing of KGCs.
b) A blockchain-based certificateless cross-domain authentication scheme for IIoT is designed, which enhances the security of key negotiation and improves authentication efficiency. Through security analysis, this paper proves that the proposed scheme can resist various attacks in the IIoT environment.
c) This paper implements the proposed scheme in the Hyperledger Fabric blockchain environment and evaluates the designed blockchain's throughput and latency performance.

1.2 Organization

The remainder of this paper is summarized as follows. Section 2 introduces related research works, including user identity and key management in Section 2.1, cross-domain authentication in Section 2.2, and decentralized KGC in Section 2.3. Section 3 presents the preliminaries of threshold secret sharing mechanisms, including Lagrange interpolation polynomial in Section 3.1, blockchain in Section 3.2, and authentication protocol in Section 3.3. In Section 4, we first give the system model and detailed design scheme, then analyze its security, including system model in Section 4.1, blockchain-based certificateless threshold scheme in Section 4.2, and informal security analysis in Section 4.3. Section 5 compares and analyzes the proposed scheme through experiments, including scheme computational efficiency in Section 5.1, comparison of protocol performance in Section 5.2, performance of blockchain in Section 5.3. Finally, Section 6 concludes this paper.

2 Related Work

2.1 User Identity and Key Management

In recent years, many studies have focused on using blockchain to manage user identities or public keys for certificateless systems. In 2019, Ali et al. [12] proposed a blockchain-based certificateless signature scheme for the Internet of Vehicles, which provides conditional privacy-protection authentication for vehicle-to-infrastructure through batch signature verification and aggregate signature verification. It also uses blockchain to keep track of how users' pseudonyms are registered and revoked. In 2020, Li et al. [13] proposed a certificateless signature scheme supporting user revocation for Internet of Vehicles (IoV), which uses a blockchain to store the revocation list and broadcasts the temporal key of the unrevoked users based on the revocation list. Although the scheme improves the transparency of the KGC's identity revocation operations, it also brings additional communication overhead. In 2021, Xu et al. [14] proposed a blockchain-based certificateless encryption scheme that registers users' IDs and public keys on the blockchain, constructs a hash table to manage the identity and public key, and realizes the updating and revocation of the user's public key by updating the public key corresponding to the ID. In 2023, Xu et al. [15] proposed a certificateless signature scheme for edge computing, which uses blockchain as public key directories and uses edge computing servers as blockchain nodes, participating in the blockchain consensus process and storing a copy of the blockchain.

Although the above certificateless application schemes [12–15] utilize the blockchain to achieve public storage, and thus manage user identities and public keys, they only propose application scenarios for blockchain without a complete design of its structure and consensus mechanism. In addition, most of them use bilinear pairing operations, which will increase the computational burden of certificateless systems, and thus are not suitable for resource-constrained IIoT environments.

2.2 Cross-Domain Authentication for IIoT

Typically, in IIoT scenarios, devices from different domains have frequent demands for communication and data exchange to achieve better cooperation. However, it becomes a challenging task to establish secure communication between different trust domains. Therefore, many scholars have worked on designing cross-domain authentication and key negotiation schemes based on blockchain. In 2020, Shen et al. [5] proposed a cross-domain authentication scheme based on the consortium blockchain, which achieves identity authentication through identity-based signatures but still suffers from the key escrow problem of ID-PKC. In 2022, Wang et al. [16] proposed a cross-domain authentication and key negotiation scheme for IIoT based on certificateless signatures also in the consortium blockchain, which tries to solve the key escrow problem of Shen's scheme. However, both

their schemes are based on bilinear pairing operations, which will increase the computational overhead of resource-constrained IIoT devices. In addition, Li et al. [17] proposed a certificateless cross-domain authentication scheme based on master-slave blockchains and edge computing. Although this scheme's master-slave multi-blockchain structure reduces the storage burden on the blockchain, it also causes the domain managers to bear the concurrent traffic of cross-chain access, which will affect the efficiency of cross-domain authentication. In 2024, Dong et al. [18] proposed a certificateless cross-domain authentication scheme based on consortium blockchains. Unfortunately, our analysis indicates that Dong's authentication mechanism in the cross-domain authentication phase of their scheme is not secure against public key replacement attacks and is vulnerable to malicious authentication requests from attackers, which may lead to Distributed Denial of Service (DDoS) attacks.

2.3 Decentralized KGC

In blockchain-based certificateless scheme, while the blockchain itself provides a distributed solution, most IIoT devices do not have the computational and storage capabilities to act as blockchain nodes, and they typically interact with the blockchain through the unique manager of their trust domain (centralized KGC) [19]. This means that if the KGC of a domain fails, it will affect the registration and authentication processes of all IIoT devices within that domain. More seriously, a malicious KGC will not only leak private information about devices but also upload erroneous device information into the blockchain, thus affecting the results of cross-domain authentication. Therefore, the single point of failure of the KGC and the over-concentration of trust are key issues affecting the reliability of certificateless cross-domain authentication systems.

To solve problems brought by the centralized KGC in certificateless systems, some scholars have proposed decentralized certificateless schemes based on blockchain smart contracts. In 2022, Wang et al. [20] proposed a blockchain-based certificateless signature scheme for IIoT, that utilizes an Ethereum-based smart contract instead of a centralized KGC to achieve partial private key distribution. In 2024, Yang et al. [21] proved that Wang's certificateless signature scheme [20] is insecure and proposed an improved scheme. In addition, Shim et al. [22] also noticed the insecurity of Wang's scheme and proposed an improved scheme, but this scheme is based on bilinear pairings, which imposes additional burdens on certificateless signatures. All of the above schemes create partial private keys through smart contracts on the public blockchain, thus avoiding forgery attacks launched by attackers using public parameters. However, it should not be ignored that problems such as slow consensus speed, low throughput, fully open information storage, and a lack of effective privacy protection on public blockchains still limit their wider application in IIoT [23].

Solving the single KGC trust concentration and single point of failure problem through threshold secret sharing is another research direction. The secret sharing mechanism was first proposed by Shamir [24] in 1979, which allows multiple users to share secret messages based on the threshold. In 2021, Wang et al. [25] applied Shamir's secret sharing protocol without a trusted center to a multi-KGC certificateless signature scheme, which achieves synchronized updates of multi-KGC keys based on blockchain. However, unfortunately, this scheme has a vulnerability of leaking the master private key of the KGC system to the user in the user key generation phase. In addition, this scheme does not consider the dynamic joining and removing of KGC servers.

3 Preliminaries

3.1 Lagrange Interpolation Polynomial

Definition: Given t arbitrary points on a two-dimensional plane, with their coordinates denoted as $(x_0, y_0), (x_1, y_1), \ldots, (x_{t-1}, y_{t-1})$, there exists a unique polynomial of degree $t-1$ denoted as $p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{t-1} x^{t-1}$ such that $p(x_i) = y_i \, (0 \le i \le t-1)$. The polynomial $p(x)$ can be obtained through polynomial interpolation, with the calculation formula as follows:

$$p(x) = \sum_{i=0}^{t} p(x_i) \prod_{j=0, j \neq i}^{t} \frac{x - x_j}{x_i - x_j} \tag{1}$$

For the polynomial $p(x)$ of degree $t-1$, its constant term coefficient a_0 can be calculated using the following formula:

$$a_0 = p(0) = \sum_{i=0}^{t} p(x_i) \prod_{j=0, j \neq i}^{t} \frac{x_j}{x_j - x_i} \tag{2}$$

3.2 Blockchain

A blockchain is a tamper-resistant ledger for recording transactions, maintained by a distributed network of mutually untrusting peers. Each peer independently holds a copy of the ledger. Through a consensus protocol, peers validate transactions, package them into blocks, and link these blocks through a hash chain. This process organizes the transactions into the ledger, ensuring consistency and order [26], the structure of the blockchain is shown in Fig. 1.

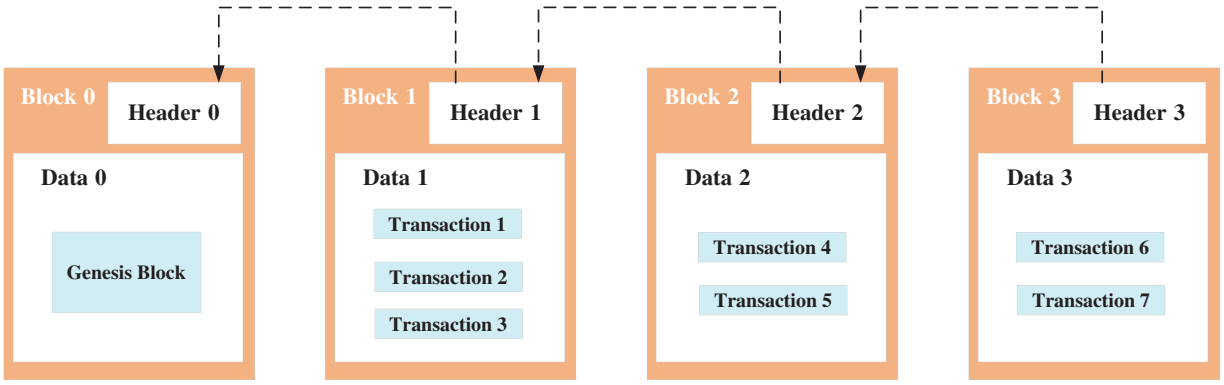

Figure 1: Structure of the blockchain

3.3 Authentication Protocol

The goal of an authentication protocol is to ensure that the communicating entities can reliably verify each other's true identities. During the key negotiation process of the protocol, there is an additional goal that the two entities share a common key known only to them. The shared key can subsequently be used to maintain confidentiality and data integrity for a certain duration [27]. This temporary cryptographic key that is generated for a specific session between communicating parties is commonly referred to as a session key.

4 Our Proposal

4.1 System Model

The system model of our proposed certificateless IIoT cross-domain authentication scheme based on blockchain is shown in Fig. 2. This system consists of IIoT devices, the multi-KGC network, and the blockchain. Since various IIoT devices belong to different industrial enterprises and industries, these devices are controlled by different trust domains and perform cross-domain authentication and communication between any two domains.

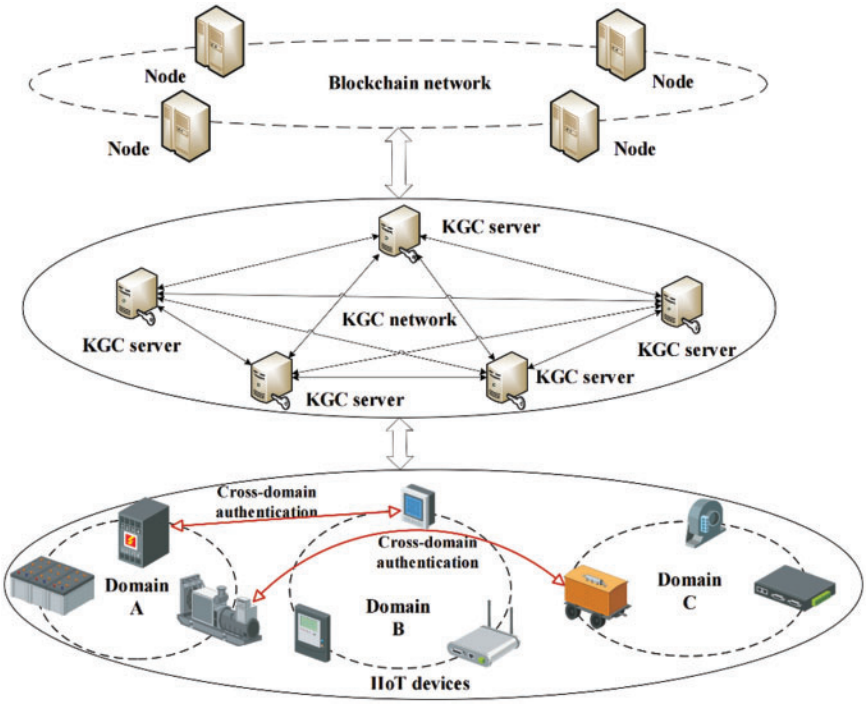

Figure 2: System model of the proposed scheme

During the system initialization phase, each domain identifies n KGC servers from the KGC network as domain certificateless service providers based on the threshold parameters (t, n). Then the relevant parameters for each domain are generated and uploaded to the blockchain. According to the threshold secret sharing protocol, only t KGC servers need to interact to generate valid partial private keys for devices in the domain. When sending partial private keys to users, KGC servers do not reveal their secrets or system master keys.

During the user registration phase, any IIoT device in the domain initiates a registration request to t KGC servers. These servers generate user-related parameters and upload them to the blockchain. At the same time, each KGC server generates a partial private key for the user and sends it to the user over a secure channel. The IIoT device aggregates the received partial private keys and combines them with its own secret value to generate a complete public and private key pair. It then completes the registration of its identity and public key on the blockchain.

During the cross-domain authentication phase, two entities involved in cross-domain communication obtain each other's domain parameters and user public keys from the blockchain. They verify each other's identity through signatures to realize cross-domain authentication. Based on this, the two

entities combine the authentication messages with their complete private keys to generate the same session key, thus completing the key negotiation process.

4.2 Blockchain-Based Certificateless Threshold Scheme

Fig. 3 is the sequence diagram of the initialization phase, the partial private key generation phase, and the user registration phase of the blockchain-based multi-KGC certificateless system in the scheme.

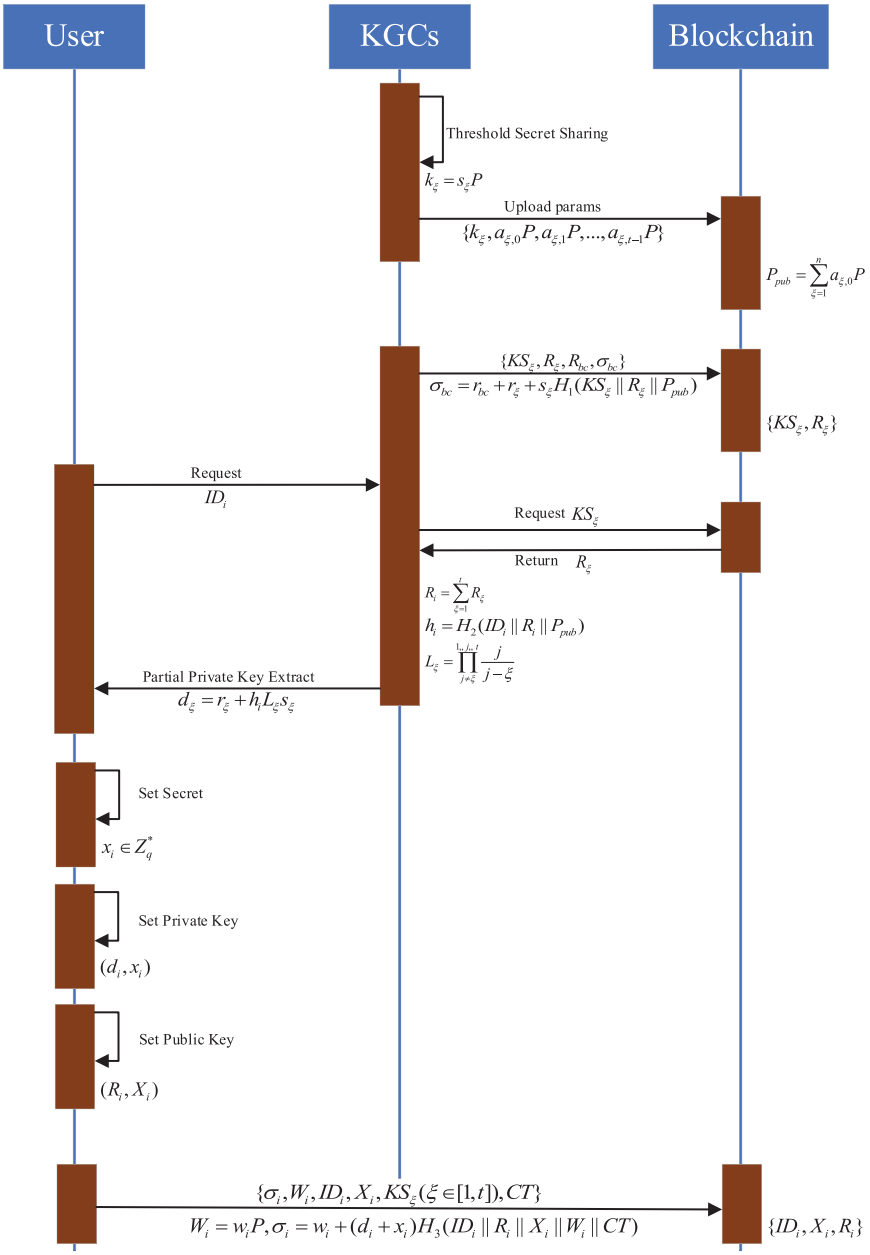

Figure 3: System initialization, partial private key generation, and user registration sequence diagram

4.2.1 Initialization Phase

Each domain selects n KGC servers from the multi-KGC network and determines the threshold value t. Afterwards, the system parameters of the respective domains are generated and uploaded to the blockchain. The specific steps are as follows:

Input the security parameter $k \in Z^+$, select an elliptic curve group G of order q, define P as the generator of G, and define four hash functions: H_1, H_2, H_3, H_4: $\{0,1\}^* \rightarrow Z_q^*$. Then, upload the system parameters $\{n, t, k, q, G, P, H_1, H_2, H_3, H_4\}$ to the blockchain ledger.

1) Each KGC server KS_ξ $(1 \leq \xi \leq n)$ generates a random $t-1$ degree polynomial: $g_\xi(x) = a_{\xi,0} + a_{\xi,1}x + \ldots + a_{\xi,t-1}x^{t-1}$, such that $a_{\xi,j} \in Z_q^*$ $(0 \leq j \leq t-1)$ and the coefficient $a_{\xi,0}$ is the secret of every KS_ξ. They then publicly announce $a_{\xi,j}P$ $(0 \leq j \leq t-1)$.
2) Each KS_ξ sends $g_\xi(j)$ through a secret channel to the other KS_j $(j \neq \xi)$. After receiving the message, KS_j verifies whether the equation $g_\xi(j)P = \sum_{k=0}^{t-1} j^k \left(a_{\xi,k}P\right)$ holds. If the verification fails, the device rejects the data; otherwise, KS_j obtains the $\{g_\xi(j) \mid 1 \leq \xi \leq n\}$.
3) Each KS_ξ can generate its private key $s_\xi = \sum_{j=1}^{n} g_j(\xi)$ and public key $k_\xi = s_\xi P$, and upload the public parameters $\{k_\xi, a_{\xi,0}P, a_{\xi,1}P, \ldots, a_{\xi,t-1}P\}$ to the blockchain. After receiving the above parameters, the blockchain smart contract verifies the validity of all the parameters uploaded by the KGC servers using $\sum_{i=1}^{n} k_i = \sum_{i=1}^{n}\sum_{j=1}^{n} g_j(i)P = \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{\xi=0}^{t-1} j^\xi \left(a_{i,\xi}P\right)$ and generates the domain's master public key $P_{pub} = \sum_{\xi=1}^{n} a_{\xi,0}P$, which will be recorded on the blockchain then. And at the same time, the domain's master key $s = \sum_{\xi=1}^{n} a_{\xi,0}$ is not held by any individual KGC server alone.

4.2.2 Dynamic Joining and Revocation of KGC Servers

Any t auxiliary KGCs can help with the joining of a new KGC and the removing of an existing KGC, and update the sub-private keys of all KGCs.

The steps for the new KGC server KS_γ to join are as follows:

1) KS_γ randomly selects e_j $(1 \leq j \leq t)$ and sends them to the corresponding auxiliary KGCs KS_j $(1 \leq j \leq t)$, and calculates $e = \sum_{j=1}^{t} e_j$; KS_j splits e_j into t parts e_{jm} $(1 \leq m \leq t)$ according to $e_j = \sum_{m=1}^{t} e_{jm}$ and sends them to the other KS_m $(m \neq j)$. Consequently, each KS_m can obtain $e_m{}' = \sum_{j=1}^{t} e_{jm}$.
2) KS_m $(1 \leq m \leq t)$ randomly selects $b_{m,1}, b_{m,2}, \ldots, b_{m,t-1}$ in Z_q^* and calculates $b_{m,0} = e_m{}' - \sum_{j=1}^{t-1} b_{m,j}\gamma^j$, obtaining a polynomial of degree $t-1$: $\delta_m(x) = b_{m,0} + b_{m,1}x + \ldots + b_{m,t-1}x^{t-1}$, and sends $\delta_m(j)$ to KS_j $(j \neq m)$. Then KS_j generates a pseudo-share $F_j = s_j + \sum_{m=1}^{t} \delta_m(j)$ and sends it to KS_γ. After

receiving pseudo-shares from the t auxiliary KGCs, the new KGC server KS_γ can generate the sub-private key $s_\gamma = \sum_{\xi=1}^{t} F_\xi \prod_{j\neq\xi}^{1\leqslant j\leqslant t} \frac{\gamma - j}{j - \xi} - e$, with the sub-public key as $k_\gamma = s_\gamma P$.

3) An arbitrary auxiliary KGC server KS_p acts as an agent. The agent splits the sub-secret $a_{p,0}$ into two random values by $a_{p,0} = a'_{p,0} + a_{\gamma,0}$, and sends $a_{\gamma,0}$ and $g_p(\gamma)$ to KS_γ through a secret channel. Subsequently, KS_p updates the $t-1$ degree polynomial to $g'_p(x) = a'_{p,0} + a'_{p,1}x + \ldots + a'_{p,t-1}x^{t-1}$, updates the partial blockchain parameters to $\{a'_{p,0}P, a'_{p,1}P, \ldots, a'_{p,t-1}P\}$, and sends $g'_p(\xi)$ to all other KS_ξ $(\xi \neq p)$. KS_γ generates a random $t-1$ degree polynomial $g_\gamma(x) = a_{\gamma,0} + a_{\gamma,1}x + \ldots + a_{\gamma,t-1}x^{t-1}$, uploads the public parameters $\{k_\gamma, a_{\gamma,0}P, a_{\gamma,1}P, \ldots, a_{\gamma,t-1}P\}$, and sends $g_\gamma(\xi)$ to all other KS_ξ $(\xi \neq \gamma)$.
4) After the above steps, all KGC servers KS_ξ can update their private keys by $s'_\xi = s_\xi - g_p(\xi) + g'_p(\xi) + g_\gamma(\xi)$, update their public keys by $k'_\xi = s'_\xi P$, and upload them to the blockchain.

To revoke an existing KGC server KS_d, the steps are as follows:

1) An arbitrary auxiliary KGC server KS_p acts as an agent, and the other auxiliary KGC servers KS_j $(j \neq p)$ send their received shares from the revoking KGC server $g_d(j)$ to KS_p. Then KS_p can obtain the secret value $a_{d,0}$ of the revoked KGC by $a_{d,0} = g_d(0) = \sum_{\xi=1}^{t} g_d(\xi) \prod_{j\neq\xi}^{1\leqslant j\leqslant t} \frac{j}{j - \xi}$.
2) KS_p updates its secret value $a''_{p,0} = a_{p,0} + a_{d,0}$, generates a $t-1$ degree polynomial $g''_p(x) = a''_{p,0} + a''_{p,1}x + \ldots + a''_{p,t-1}x^{t-1}$, updates the partial blockchain parameters $\{a''_{p,0}P, a''_{p,1}P, \ldots, a''_{p,t-1}P\}$, and sends $g''_p(\xi)$ to all other KS_ξ $(\xi \neq p, \xi \neq d)$.
3) After the above steps, all KGC servers can update their private keys by $s''_\xi = s_\xi - g_d(\xi) - g_p(\xi) + g''_p(\xi)$, update their public keys by $k''_\xi = s''_\xi P$, and upload them to the blockchain.

4.2.3 Partial Private Key Generation Phase

A group of t KGC servers in the domain generate and submit random parameters to the blockchain. Subsequently, the KGC servers work together to generate a partial private key for a user U_i identified as $ID_U \in \{0,1\}^*$. The steps are as follows:

1) For any t KGC servers KS_ξ $(1 \leq \xi \leq t)$, they can generate Lagrange coefficients, represented as: $L_\xi = \prod_{j\neq\xi}^{1\leqslant j\leqslant t} \frac{j}{j - \xi}$.
2) Each KGC server KS_ξ $(1 \leq \xi \leq t)$ generates random values r_{bc} and r_ξ on Z_q^*, and computes $R_\xi = r_\xi P$, $R_{bc} = r_{bc}P$; then, KS_ξ generates the signature $\sigma_{bc} = r_{bc} + r_\xi + s_\xi H_1\left(KS_\xi \| R_\xi \| P_{pub}\right)$, and submits the set $\{KS_\xi, R_\xi, R_{bc}, \sigma_{bc}\}$ to the blockchain. The blockchain verifies the signature validity through $\sigma_{bc}P = R_{bc} + R_\xi + k_\xi H_1\left(KS_\xi \| R_\xi \| P_{pub}\right)$. If the equation holds, the pair $\{KS_\xi, R_\xi\}$ is recorded in the blockchain ledger.
3) When receiving the partial private key generation request from the user U_i, KS_ξ initiates a query request to the blockchain, inputs the parameter KS_ξ $(1 \leq \xi \leq t)$, and obtains $R_i = \sum_{\xi=1}^{t} R_\xi = \sum_{\xi=1}^{t} r_\xi P$. It then computes $h_i = H_2\left(ID_i \| R_i \| P_{pub}\right)$, and generates the sub-partial private key $d_\xi = r_\xi + h_i L_\xi s_\xi$. KS_ξ then sends d_ξ and the set $\{KS_\xi, R_\xi, R_{bc}, \sigma_{bc}\}$ to the user U_i through a secure channel. Then, KS_ξ can perform step 2 again to update the pair in the blockchain to $\left\{KS_\xi, R'_\xi\right\}$.

4) When receiving messages from t KS_ξ servers, the user U_i can compute $R_i = \sum_{\xi=1}^{t} R_\xi$ and $h_i = H_2\left(ID_i \| R_i \| P_{pub}\right)$, and then verifies the validity of each sub-partial private key through $d_\xi P = R_\xi + h_i L_\xi k_\xi$. When t valid sub-partial private keys are obtained, the partial private key $d_i = \sum_{\xi=1}^{t} d_\xi = \sum_{\xi=1}^{t} r_\xi + sh_i$ can be calculated, and the corresponding partial public key is $d_i P = R_i + h_i P_{pub}$.

4.2.4 User Registration Phase

This phase is launched by the user U_i, who generates the complete public-private key pair with the help of multiple KGCs and completes the verification and registration on the blockchain by uploading a signature. The steps are as follows:

1) The user U_i randomly selects $x_i \in Z_q^*$ as the secret value and computes $X_i = x_i P$.
2) The user U_i generates public keys denoted as (R_i, X_i) and private keys denoted as (d_i, x_i) .
3) The user U_i randomly selects $w_i \in Z_q^*$, computes $W_i = w_i P$; records the signature timestamp as CT, and computes $\sigma_i = w_i + (d_i + x_i)\, H_3\,(ID_i \| R_i \| X_i \| W_i \| CT)$, thereby generating the signature (σ_i, W_i).
4) The user U_i sends the parameters $\{\sigma_i, W_i, ID_i, X_i, KS_\xi\ (\xi \in [1, t]), CT\}$ to the blockchain through the KGC server. Subsequently, the blockchain smart contract verifies the signature validity through $\sigma_i P = W_i + \left(R_i + h_i P_{pub} + X_i\right) H_3\,(ID_i \| R_i \| X_i \| W_i \| CT)$. If the equation holds, the triplet $\{ID_i, X_i, R_i\}$ is recorded in the blockchain ledger.

4.2.5 Cross-Domain Authentication Protocol

Assuming Alice and Bob are the two parties involved in the protocol for IIoT cross-domain communication, they belong to domains A and B, respectively. In our protocol, both domains have the same partial system parameters $\{k, q, G, P, H_1, H_2, H_3, H_4\}$ and their own different P_{pub-A} and P_{pub-B}. After the certificateless system initialization, partial private key generation, and user registration phases, Alice and Bob have generated their complete private keys (d_A, x_A) and (d_B, x_B), respectively. At the same time, the blockchain ledger also has recorded two sets of parameters for Alice, $\{ID_A, X_A, R_A\}$, and for Bob, $\{ID_B, X_B, R_B\}$. The specific process of the protocol is as follows:

1) Alice selects a random value $v_A \in Z_q^*$, computes $V_A = v_A P$, calculates $T_A = (x_A + d_A)\, V_A$, generates the timestamp t_1, and creates a signature $\sigma_A = w_A + (x_A + d_A)\, H_4\,(ID_A \| ID_B \| V_A \| T_A \| t_1)$. Then, Alice sends the authentication message $\{ID_A, \sigma_A, V_A, T_A, t_1\}$ to Bob.
2) After receiving the message, Bob first checks the validity of the timestamp t_1. Then, Bob inputs ID_A to the blockchain node and queries $\{P_{pub-A}, X_A, R_A\}$. If there are no relevant parameters for Alice, a user from Domain A, the process is terminated. Otherwise, Bob verifies $\sigma_A P = W_A + \left(R_A + h_A P_{pub-A} + X_A\right) H_4\,(ID_A \| ID_B \| V_A \| T_A \| t_1)$. If the equation holds, Bob confirms that the message is from the registered user Alice from Domain A.
3) Bob selects a random value $v_B \in Z_q^*$, computes $V_B = v_B P$, calculates $T_B = (x_B + d_B)\, V_B$, generates the timestamp t_2, and creates a signature $\sigma_B = w_B + (x_B + d_B)\, H_4\,(ID_A \| ID_B \| V_B \| T_B \| t_2)$. Then, Bob sends the authentication message $\{ID_B, \sigma_B, V_B, T_B, t_2\}$ to Alice.
4) After receiving the message, Alice first checks the validity of the timestamp t_2. Then, Alice inputs ID_B to the blockchain node and queries $\{P_{pub-B}, X_B, R_B\}$. If there are no relevant parameters for Bob, a user from Domain B, the process is terminated. Otherwise, Alice

verifies $\sigma_B P = W_B + \left(R_B + h_B P_{pub-B} + X_B\right) H_4 \left(ID_A \| ID_B \| V_B \| T_B \| t_2\right)$. If the equation holds, Alice confirms that the message is from the registered user Bob from Domain B.

5) After the above steps, Alice and Bob can compute the same session key: $SK = v_A \left(x_A + d_A\right) T_B = v_B \left(x_B + d_B\right) T_A = v_A v_B \left(x_A + d_A\right) \left(x_B + d_B\right) P$.

4.3 Informal Security Analysis

Here, the security of the proposed scheme is informally analyzed, which shows that the scheme is robust to well-known adversarial attacks.

4.3.1 KGC Collusion Attacks

In our multi-KGC certificateless system, the system master key is not held by any single KGC server alone. Therefore, in order to obtain the master key and control the system, the adversary must control a certain number of KGC servers to cooperate. However, according to the threshold secret sharing principle, for the threshold value t set for the domain, if there are fewer than t servers cooperating in the domain, the sub-secret $a_{\xi,0}$ of each server KS_ξ cannot be recovered, and the domain master key needs to be obtained by $s = \sum_{\xi=1}^{n} a_{\xi,0}$. Thus, the system can prevent collusion of up to $t-1$ servers from recovering the domain master key.

4.3.2 Public Key Replacement Attacks

In our multi-KGC certificateless system, the blockchain is managed by each peer-to-peer node in the form of a shared ledger and follows immutability. After the user registration phase, each registered IIoT device U_i has a corresponding triplet $\{ID_i, X_i, R_i\}$ in the blockchain ledger. In practice, the blockchain introduces a signature verification mechanism in the user update process of the blockchain ledger. If an adversary wishes to impersonate a user to replace the identity and public key information in the blockchain ledger, he can either become the administrator of the blockchain controlling the majority of the nodes to modify the ledger or the contract, or he can initiate a transaction to update the information about the ledger legally by verifying the signature. But without possessing the user's private key, adversaries cannot impersonate the user to generate a valid signature. Therefore, based on the trust in the blockchain consensus mechanism, an adversary can be prevented to some extent from launching a replacement attack on the public key of a registered user in the blockchain ledger.

4.3.3 Device Capture Attacks

In our multi-KGC certificateless system, even if an IIoT device is captured and an adversary discovers the user's public key (R_i, X_i) and private key (d_i, x_i) through power analysis technology, it is computationally infeasible for the adversary to find $\sum_{\xi=1}^{t} r_\xi$ from Z_q^* that satisfies $R_i = \sum_{\xi=1}^{t} r_\xi P$ based on the Elliptic Curve Discrete Logarithm Problem (ECDLP) assumption. Thereby, the adversary cannot recover the master key s from $d_i = \sum_{\xi=1}^{t} r_\xi + sh_i$ without knowing $\sum_{\xi=1}^{t} r_\xi$. That is, adversaries cannot utilize the captured device to obtain the domain master key, nor can they forge legitimate signatures.

4.3.4 Forward Secrecy

Forward secrecy implies that even if a user's long-term key is compromised, the confidentiality of the previous session key will not be affected. The proposed protocol is based on the hardness

assumption of ECDLP, and generates a session key by combining the private key with a new nonce in each execution. So even in the case of a long-term key (private key) compromise, it is still hard for an attacker to derive previous session keys, as they would have to know the specific nonce used in each session. Therefore, the protocol guarantees forward secrecy for cross-domain authentication.

4.3.5 Replay Attacks

Replay attacks imply that adversaries intercept information transmitted during communication and replay it in subsequent authentication processes. In our protocol, the message sender performs the signature algorithm by signing the current timestamp as part of the message, which ensures that the message is fresh and cannot be tampered with. On the other hand, after receiving the message, the message receiver can also examine the timestamp to prevent the message from being replayed. Therefore, this protocol is resistant to replay attacks.

4.3.6 Eavesdropping Attacks

Eavesdropping attacks imply that adversaries intercept the authentication messages of both communicating parties during key negotiation and forge the session key. In our protocol, if the adversary intercepts both Alice's authentication message $T_A = (x_A + d_A) V_A$ and Bob's authentication message $T_B = (x_B + d_B) V_B$, the probability of the adversary obtaining the session key $SK = v_A v_B (x_A + d_A) (x_B + d_B) P$ is negligible based on the difficulty assumption of the Elliptic Curve Computational Diffie-Hellman (ECCDH) problem. Therefore, this protocol is protected against eavesdropping attacks.

4.3.7 Key Leakage Attacks

Key leakage attacks imply that adversaries obtain the user's key by intercepting information about the negotiation authentication during the key negotiation process. In our protocol, even if the adversary intercepts Alice's authentication message $T_A = (x_A + d_A) V_A$ and $V_A = v_A P$, based on the hardness assumption of the ECDLP, the probability that the adversary finds the legitimate key $(x_A + d_A)$ from Z_q^* is negligible. Thus, adversaries cannot obtain Alice's key information and cannot impersonate Alice to communicate with Bob. Therefore, this protocol can resist key leakage attacks.

5 Performance Analysis

In this section, the computational efficiency of the proposed scheme is comparatively analyzed under different threshold parameters. Also, our scheme is compared with other existing solutions regarding cross-domain authentication and key negotiation. Furthermore, the performance of blockchain operations in the proposed scheme is analyzed in terms of system throughput and average latency. The simulation platform runs on an Intel Core i5-3740 @ 3.20 GHz CPU with 4 GB of RAM, under the Ubuntu 20.04 operating system.

5.1 Scheme Computational Efficiency

This subsection uses the Golang programming language, based on the Crypto standard library, to evaluate the computational efficiency of the proposed scheme. The elliptic curve parameters used for the simulation tests are Secp256r1.

In our multi-KGC certificateless scheme, since the computational efficiency is mainly affected by the total number of KGC servers n and the threshold parameter t during system initialization, KGC

joining and revocation, and user partial private key generation and verification, this paper analyzes and compares the computational efficiency of these three phases at first.

Fig. 4 shows the computational cost of the system initialization phase, KGC joining and revocation phase, and user partial private key generation phase when the total number of KGC servers n is fixed ($n = 35$) and the threshold parameter t varies.

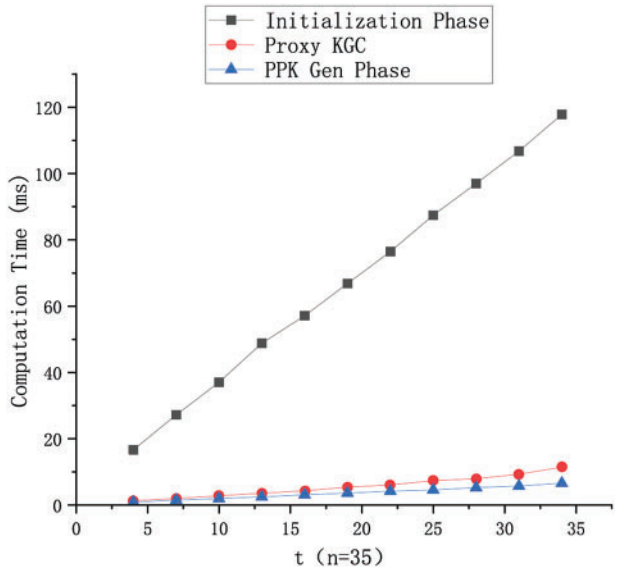

Figure 4: When $n = 35$, computational cost for different values of threshold parameter

Fig. 5 shows the computational cost of the system initialization phase, KGC joining and revocation phase, and user partial private key generation phase when the threshold parameter t is fixed ($t = 20$) and the total number of KGC servers n varies.

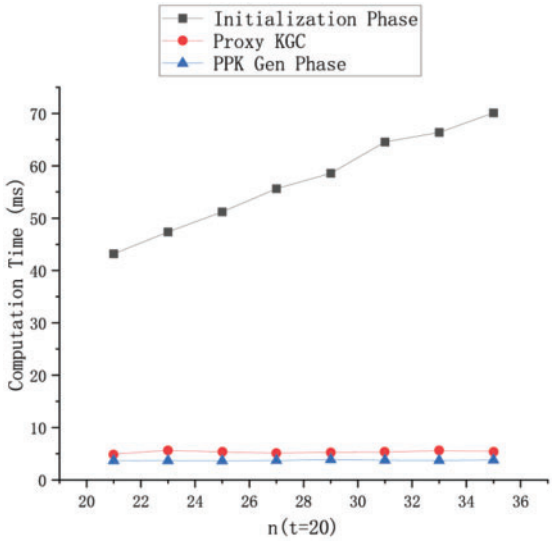

Figure 5: When $t = 20$, computational cost for different number of KGC servers

From the Figs. 4 and 5, although the locations of the sample points show slight deviations from an ideal straight line, these two figures still visualize the change of computational overhead as the independent variables t and n vary:

1) The trend for the initialization phase indicates that its computation overhead is almost linearly related to both t and n.
2) The trend for the proxy KGC during KGCs' dynamic changing indicates that its computation overhead is almost linearly related to t, and independent of n.
3) The trend for the partial private key generation phase indicates that its computation overhead is almost linearly related to t, and independent of n.

In conjunction with the scheme proposed in this paper, the correlation and non-correlation reflected in these two figures can be explained as follows:

1) In the execution of the proposed scheme, scalar multiplications on elliptic curves bear the main cryptographic operations of the scheme, while point addition on elliptic curves and number multiplication in finite fields are almost negligible. So the computational overhead is mainly related to the number of scalar multiplications performed.
2) In the initialization phase, each KGC in a (t, n)-threshold multi-KGC system needs to undergo t scalar multiplications for the polynomial parameter generation process, $t(n-1)$ scalar multiplications for the verification process, and 1 scalar multiplication for the public key generation process, which sum up to $tn+1$ times.
3) In the dynamic joining and revocation phase, the proxy KGC needs to undergo t scalar multiplications to update its blockchain parameters.
4) In the partial private key generation phase, the user needs to undergo $2t$ scalar multiplications to realize the checking of the sub-partial private keys for t auxiliary KGCs.

After a specific analysis of the variations in the threshold parameters, this paper will next analyze their impact on the security of multi-KGC certificateless systems. In the design of the (t, n) threshold system based on Shamir's secret sharing, collusive attacks with no more than $t-1$ participants are prevented. So in the practical application of the proposed scheme, too small a threshold value t can cause a security threat to the whole multi-KGC system, in other words, once a few of them are controlled, the adversary has the ability to execute all the behaviors of the domain. However, at the same time, if the threshold value is set too large, although the security is improved, the system resources consumed will again increase substantially, which will affect the efficiency and real-time performance of the cryptographic service. In other words, the threshold parameter needs to be flexibly adjusted according to the requirements of IIoT scenarios under the premise of ensuring a certain degree of security.

5.2 Comparison of Protocol Performance

The performance of cross-domain authentication and key negotiation protocols depends largely on the cryptographic operations involved, such as scalar multiplication and bilinear pairing. In order to better compare with other certificateless cross-domain authentication schemes, this section uses the Miracl library to evaluate the computational overhead of various main cryptographic operations based on the Secp160r1 parameters. The measured time for the main cryptographic operation is as follows: (1) bilinear pairing operation: $BP = 2.96227$ ms, (2) scalar multiplication operation on group G: $SM = 0.331978$ ms, (3) modular exponentiation operation on group G_T: $ET = 0.303923$ ms. Other

cryptographic operations, such as point addition, hashing, and modular inverse operations, are not considered here due to their relatively lower computational overhead.

Based on the measured computational overhead of the main cryptographic operations, this paper investigates the existing certificateless cross-domain authentication schemes and compares their computational overhead during the cross-domain authentication and key negotiation phases. Fig. 6 shows the performance comparison of various schemes.

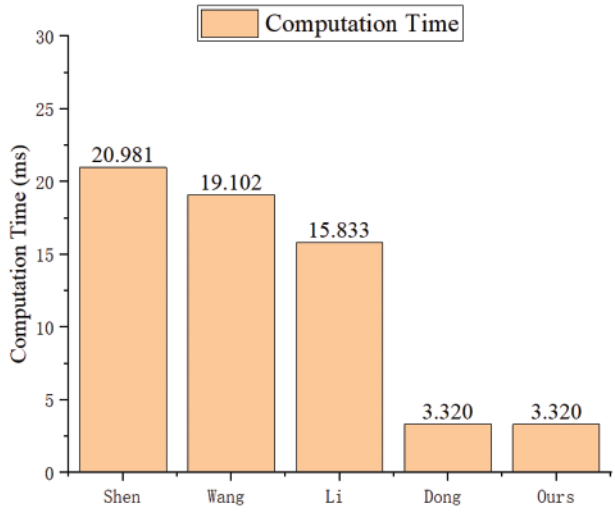

Figure 6: Performance comparison of various schemes [5,16,28,18]

In the comparison of various schemes, this paper excludes all cryptographic operations involved in their public and private keys generation process. Instead, this paper focuses on the cross-domain authentication and key negotiation process based on the existing public and private keys. In the above process, both communicating entities implement mutual cross-domain authentication, and each entity generates a session key. The total time spent by both parties in performing the main cryptographic operations is summarized and recorded based on the test results of the local cryptographic operations.

Due to the use of bilinear pairing operations in references [5,16,28], the computational overhead is relatively heavy, which leads to a larger overall cost, especially for resource-constrained IIoT devices, greatly reducing their cross-domain authentication performance. In addition, the heavy computational tasks of bilinear pairing operations may also introduce higher latency, which is unacceptable for IIoT devices that require real-time response. Although reference [18] eliminated bilinear pairing operations, we find that its cross-domain authentication process is not secure against public key replacement attacks, and is vulnerable to malicious authentication request attacks from attackers, posing a significant security risk. The cross-domain authentication process in our scheme is based on generating and verifying digital signatures and guaranteeing that they cannot be forged based on the ECDLP, while adding little extra overhead compared to the reference [18]. Therefore, compared to other literature schemes, the proposed certificateless cross-domain authentication and key agreement protocol can achieve safer cross-domain authentication and key negotiation at a relatively lower computational cost.

5.3 Performance of Blockchain

The prototype blockchain system is designed based on the enterprise-level blockchain platform Hyperledger Fabric, and its performance is evaluated. The version of the blockchain platform used in the experiment is Fabric 2.2.

Specifically, the experiment tests and analyzes the smart contracts of the registration (user registration and KGC registration) and query (public system parameters, user public keys, etc.) processes in the proposed certificateless scheme. The main performance metrics include transaction throughput, system throughput, and average transaction latency.

Fig. 7 shows the performance of the blockchain during the registration process under different transaction send rates.

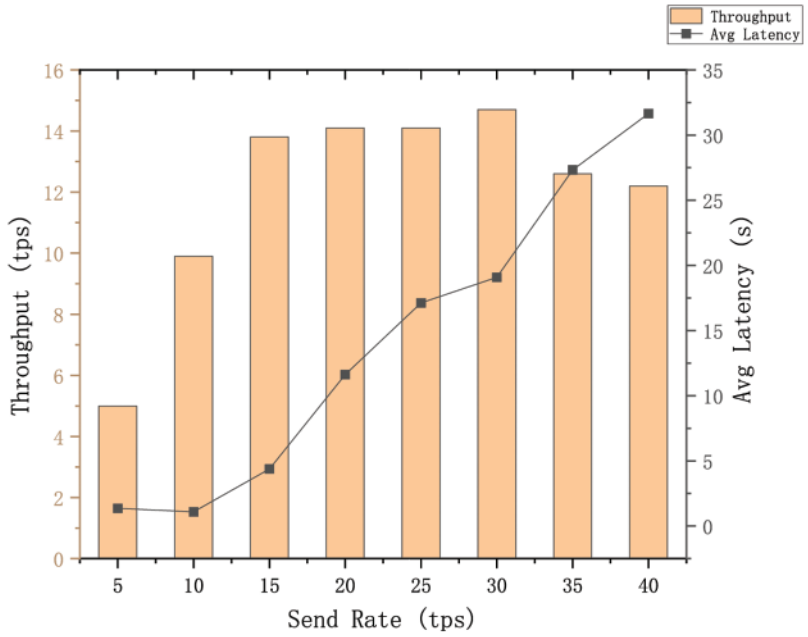

Figure 7: Performance of the blockchain during the registration process

As shown in Fig. 7, when the transaction send rate is less than 15 tps, the transaction sending rate is low, and almost every received transaction can be processed in real-time, so the latency of the blockchain system at this time mainly depends on the length of time that the blockchain generates blocks, and the average transaction latency is relatively low at this time, roughly within 3 s. However, as the transaction sending rate gradually increases, especially when the transaction send rate exceeds 15 tps, the growth of system throughput is not obvious, and the average transaction latency tends to increase further. This is due to network congestion occurring, which means that transactions are forced to wait and cannot be processed in time, which in turn leads to a rise in transaction latency, and worse, more and more transactions are lost during this period as well.

Fig. 8 shows the performance of the blockchain during the query process under different transaction send rates.

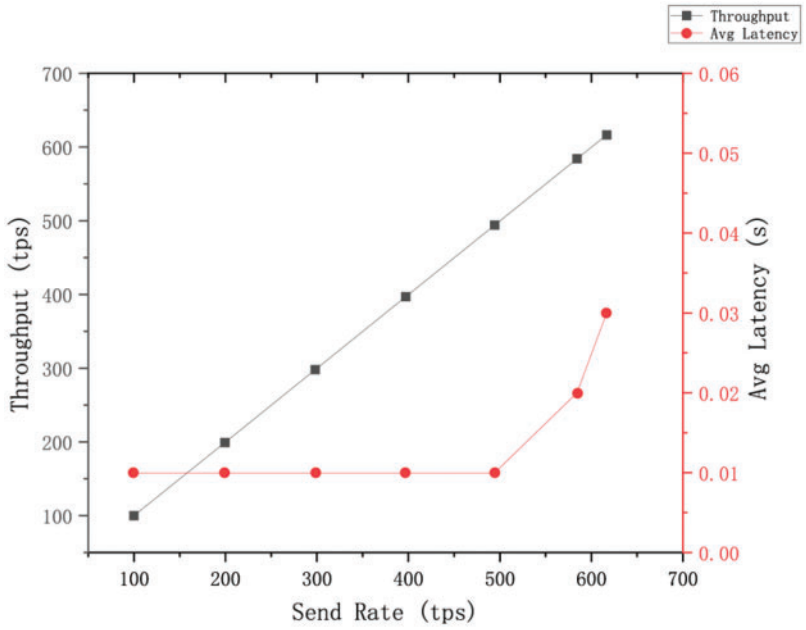

Figure 8: Performance of the blockchain during the query process

As shown in Fig. 8, the system throughput increases with the transaction send rate and reaches its maximum when the transaction send rate reaches about 600 tps due to the transaction capacity reaching the upper limit. In addition, compared to the entity registration process, the average transaction latency of the entity public parameter query process is relatively low, as this query process does not need to wait for consensus and can read data directly from the state database. As the transaction sending rate increases, especially when the transaction send rate exceeds 500 tps, there is a certain degree of attenuation in the efficiency of the blockchain nodes in querying the ledger data, and the average transaction latency rises more significantly, but since the latency of the process is in a smaller order of magnitude compared to the registration process, the latency is still within the acceptable range.

This paper focuses on the realization of a certificateless cryptographic scheme based on blockchain to solve the security problem in IIoT, so it adopts the more mature Hyperledger Fabric to realize the prototype system. While the references [9,10] focused on the improvement of the performance of blockchain in the IIoT application, they provide effective schemes to improve transaction data processing. Also, in terms of the next step, the improvement of blockchain is an important way to further optimize the performance of the scheme in this paper.

6 Conclusion

This paper has proposed a certificateless multi-KGC cross-domain authentication scheme based on blockchain for IIoT, which incorporates a threshold secret-sharing mechanism without a trusted center. The scheme can support the dynamic joining and removal of KGCs and can efficiently achieve cross-domain identity authentication and key negotiation. The results showed that the proposed scheme could achieve a multi-KGC certificateless system with low computational overhead and can implement cross-domain authentication and key negotiation more securely and efficiently. In addition, the blockchain prototype system in the proposed scheme can be well implemented based

on the Hyperledger Fabric blockchain platform. Although this paper has simulated and tested the performance of the proposed scheme based on a prototype system, the deployment of the blockchain-based multi-KGC system and the communication process of resource-constrained IIoT devices are not fully discussed, and the optimal threshold parameter intervals for the practical application of the scheme should also be further explored. In future work, this paper considers exploring a more suitable range of threshold parameters to balance the security and efficiency of the multi-KGC certificateless system, taking into account the needs of different IIoT application scenarios.

Acknowledgement: We would like to extend our sincere thanks to the following individuals for their contributions to this work: Hong Zhao, for his instrumental role in the scheme improvement and for providing critical insights that greatly enhanced the quality of our study. The team at Cryptographic Evaluation and Secure Communications Laboratory, for their technical assistance and for facilitating access to essential equipment and resources. All participants in our study, who generously volunteered their time and shared their experiences, making this work possible.

Funding Statement: This work was supported in part by the Fundamental Research Funds for the Central Universities (Nos. 3282024052, 3282024058), and the "Advanced and Sophisticated" Discipline Construction Project of Universities in Beijing (No. 20210013Z0401).

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Zhaobin Li, Xiantao Liu; data collection: Nan Zhang; analysis and interpretation of results: Zhaobin Li, Xiantao Liu, Nan Zhang; draft manuscript preparation: Xiantao Liu, Zhanzhen Wei. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the corresponding author, Xiantao Liu, upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman and D. O. Wu, "Edge computing in industrial internet of things: Architecture, advances and challenges," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 4, pp. 2462–2488, 2020. doi: 10.1109/COMST.2020.3009103.

[2] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial internet of things: Challenges, opportunities, and directions," *IEEE Trans. Ind. Inform.*, vol. 14, no. 11, pp. 4724–4734, 2018. doi: 10.1109/TII.2018.2852491.

[3] F. Li *et al.*, "BLMA: Editable blockchain-based lightweight massive IIoT device authentication protocol," *IEEE Internet Things J.*, vol. 10, no. 24, pp. 21633–21646, 2023. doi: 10.1109/JIOT.2023.3308725.

[4] K. Tange, M. De Donno, X. Fafoutis, and N. Dragoni, "A systematic survey of industrial internet of things security: Requirements and fog computing opportunities," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 4, pp. 2489–2520, 2020. doi: 10.1109/COMST.2020.3011208.

[5] M. Shen *et al.*, "Blockchain-assisted secure device authentication for cross-domain industrial IoT," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 942–954, 2020. doi: 10.1109/JSAC.2020.2980916.

[6] T. Li, H. Wang, D. He, and J. Yu, "Designated-verifier aggregate signature scheme with sensitive data privacy protection for permissioned blockchain-assisted IIoT," *IEEE Trans. Inf. Forensics Secur.*, vol. 18, pp. 4640–4651, 2023. doi: 10.1109/TIFS.2023.3297327.
[7] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. CRYPTO '84*, Santa Barbara, CA, USA, 1984, pp. 47–53. doi: 10.1007/3-540-39568-7_5.
[8] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Proc. ASIACRYPT*, Taipei, Taiwan, 2003, pp. 452–473. doi: 10.1007/978-3-540-40061-5_29.
[9] S. Jiang, J. Cao, H. Wu, and Y. Yang, "Fairness-based packing of industrial IoT data in permissioned blockchains," *IEEE Trans. Ind. Inform.*, vol. 17, no. 11, pp. 7639–7649, 2021. doi: 10.1109/TII.2020.3046129.
[10] S. Jiang, J. Cao, C. L. Tung, Y. Wang, and S. Wang, "SHARON: Secure and efficient cross-shard transaction processing via shard rotation," in *IEEE INFOCOM 2024*, 2024.
[11] H. -N. Dai, Z. Zheng, and Y. Zhang, "Blockchain for internet of things: A survey," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8076–8094, 2019. doi: 10.1109/JIOT.2019.2920987.
[12] I. Ali, M. Gervais, E. Ahene, and F. Li, "A blockchain-based certificateless public key signature scheme for vehicle-to-infrastructure communication in VANETs," *J. Syst. Archit.*, vol. 99, no. 1, 2019, Art. no. 101636. doi: 10.1016/j.sysarc.2019.101636.
[13] K. Li, W. F. Lau, M. H. Au, I. W. -H. Ho, and Y. Wang, "Efficient message authentication with revocation transparency using blockchain for vehicular networks," *Comput. Elect. Eng.*, vol. 86, no. 7, 2020, Art. no. 106721. doi: 10.1016/j.compeleceng.2020.106721.
[14] G. Xu, J. Dong, and C. Ma, "A certificateless encryption scheme based on blockchain," *Peer-to-Peer Netw. Appl.*, vol. 14, no. 5, pp. 2952–2960, Sep. 2021. doi: 10.1007/s12083-021-01147-w.
[15] G. Xu, J. Dong, C. Ma, J. Liu, and U. G. Omar Cliff, "A certificateless signcryption mechanism based on blockchain for edge computing," *IEEE Internet Things J.*, vol. 10, no. 14, pp. 11960–11974, 2023. doi: 10.1109/JIOT.2022.3151359.
[16] X. Wang, C. Gu, F. Wei, S. Lu, and Z. Li, "A certificateless-based authentication and key agreement scheme for IIoT cross-domain," *Secur. Commun. Netw.*, vol. 2022, Oct. 2022, Art. no. 3693748. doi: 10.1155/2022/3693748.
[17] W. Li, S. Zhang, Z. Chen, and L. Sen, "Cross-domain authentication scheme for IoT devices based on blockchain," presented at the Proc. 2022 IEEE 13th Int. Con. Soft. Eng. Serv. Sci. (ICSESS), Oct. 21–23, 2022, pp. 67–73. doi: 10.1109/ICSESS54813.2022.9930157.
[18] J. Dong, G. Xu, C. Ma, J. Liu, and U. G. O. Cliff, "Blockchain-based certificate-free cross-domain authentication mechanism for Industrial Internet," *IEEE Internet Things J.*, vol. 11, no. 2, pp. 3316–3330, 2024. doi: 10.1109/JIOT.2023.3296506.
[19] W. Mao, P. Jiang, and L. Zhu, "BTAA: Blockchain and TEE-assisted authentication for IoT systems," *IEEE Internet Things J.*, vol. 10, no. 14, pp. 12603–12615, 2023. doi: 10.1109/JIOT.2023.3252565.
[20] W. Wang, H. Xu, M. Alazab, T. R. Gadekallu, Z. Han and C. Su, "Blockchain-based reliable and efficient certificateless signature for IIoT devices," *IEEE Trans. Ind. Inform.*, vol. 18, no. 10, pp. 7059–7067, 2022. doi: 10.1109/TII.2021.3084753.
[21] X. Yang, W. Wang, T. Tian, and C. Wang, "Cryptanalysis and improvement of a blockchain-based certificateless signature for IIoT devices," *IEEE Trans. Ind. Inform.*, vol. 20, no. 2, pp. 1884–1894, 2024. doi: 10.1109/TII.2023.3282317.
[22] K. -A. Shim, "A secure certificateless signature scheme for cloud-assisted Industrial IoT," *IEEE Trans. Ind. Inform.*, vol. 20, no. 4, pp. 6834–6843, 2024. doi: 10.1109/TII.2023.3343437.
[23] X. Guo, Q. Guo, M. Liu, Y. Wang, Y. Ma and B. Yang, "A certificateless consortium blockchain for IoTs," presented at the Proc. 2020 IEEE 40th Int. Con. Distri. Comp. Syst. (ICDCS), Nov. 29–Oct. 1, 2020, pp. 496–506. doi: 10.1109/ICDCS47774.2020.00054.
[24] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979. doi: 10.1145/359168.359176.

[25] L. P. Wang, J. B. Gao, Q. S. Li, and Z. Chen, "Blockchain-based multi-recipient multi-message signcryption scheme," (in Chinese), *J. Softw.*, vol. 32, no. 11, pp. 3606–3627, Nov. 2021. doi: 10.13328/j.cnki.jos.006034.
[26] E. Androulaki *et al.*, "Hyperledger fabric: A distributed operating system for permissioned blockchains," presented at the Proc. Thirteenth EuroSys Conf., Porto, Portugal, 2018. doi: 10.1145/3190508.3190538.
[27] W. Diffie, P. C. Van Oorschot, and M. J. Wiener, "Authentication and authenticated key exchanges," *Des., Codes Crypt.*, vol. 2, no. 2, pp. 107–125, 1992. doi: 10.1007/BF00124891.
[28] Y. Li, W. Chen, Z. Cai, and Y. Fang, "CAKA: A novel certificateless-based cross-domain authenticated key agreement protocol for wireless mesh networks," *Wirel. Netw.*, vol. 22, no. 8, pp. 2523–2535, Nov. 2016. doi: 10.1007/s11276-015-1109-7.