Domain-Level Anonymous Cross-Domain Authentication Scheme for IIoT Based on Blockchain

*LIANG Yufeng**, *SUN Lu*

College of Computer Science and Technology / College of Software, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, P. R. China

(Received 10 June 2025; revised 11 September 2025; accepted 5 October 2025)

Abstract: The rapid development of the industrial internet of things (IIoT) has brought huge benefits to factories equipped with IIoT technology, each of which represents an IIoT domain. More and more domains are choosing to cooperate with each other to produce better products for greater profits. Therefore, in order to protect the security and privacy of IIoT devices in cross-domain communication, lots of cross-domain authentication schemes have been proposed. However, most schemes expose the domain to which the IIoT device belongs, or introduce a single point of failure in multi-domain cooperation, thus introducing unpredictable risks to each domain. We propose a more secure and efficient domain-level anonymous cross-domain authentication (DLCA) scheme based on alliance blockchain. The proposed scheme uses group signatures with decentralized tracing technology to provide domain-level anonymity to each IIoT device and allow the public to trace the real identity of the malicious pseudonym. In addition, DLCA takes into account the limited resource characteristics of IIoT devices to design an efficient cross-domain authentication protocol. Security analysis and performance evaluation show that the proposed scheme can be effectively used in the cross-domain authentication scenario of industrial internet of things.

Key words: industrial internet of things (IIoT); domain-level anonymity; cross-domain authentication; group signature

CLC number: TN918.4 **Document code:** A **Article ID:** 1005-1120(2025)S-0180-15

0 Introduction

Today, an increasing number of countries are encouraging factories to integrate industrial internet of things (IIoT) into various stages of production processes, thereby enhancing manufacturing efficiency and energy utilization, and elevating traditional industries to a new level of intelligence[1]. IIoT, as an important part of the internet of things, has entered a stage of rapid development. Each individual factory equipped with IIoT technology represents an IIoT domain. As manufacturing becomes more complex, multiple domains must collaborate to produce higher quality products and reap greater benefits. Therefore, a cross-domain authentication scheme that supports collaboration between multiple domains and establishes secure data exchange and information sharing has become a critical requirement for the development of IIoT. Although existing IIoT network infrastructures can easily connect IIoT devices across different domains, direct communication and cooperation between devices from different domains may lead to production data leakage or even endanger the production process due to each domain prioritizing its own interests and mutual distrust[2-3]. Therefore, it is imperative to propose a scheme that enables mutual authentication and session key establishment between IIoT devices from different domains while guaranteeing device privacy and data security.

In order to establish a trusted platform among domains and reduce the communication overhead as-

*Corresponding author, E-mail address: liangyufeng@nuaa.edu.cn.

How to cite this article: LIANG Yufeng, SUN Lu. Domain-level anonymous cross-domain authentication scheme for IIoT based on blockchain[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2025,42(S):180-194.

http://dx.doi.org/10.16356/j.1005-1120.2025.S.016

sociated with credential passing, researchers have introduced consortium blockchain technology when designing cross-domain authentication schemes[4-8]. A consortium blockchain is a distributed ledger maintained by several entities, where entities must undergo validation before joining the blockchain. The entities within the blockchain do not completely trust each other but operate under specific constraints and collaborate[9]. Therefore, consortium blockchain technology is incorporated to achieve secure sharing of public information in our proposed scheme.

Existing research has taken into account the problem of privacy leakage caused by attackers linking a device's real identity through intercepted messages, and thus has incorporated pseudonym management mechanisms to address this issue[4,10-12]. Most pseudonym management mechanisms are based on representative entities, referred to as the TA, which generates pseudonyms for IIoT devices within their respective domains and is responsible for revealing the true identities behind malicious pseudonyms. However, due to competitive and distrustful relationships between domains, TA may deceive for the benefit of their own domain when disclosing the real identities behind malicious pseudonyms. For instance, a TA might publish an unrelated identity for the malicious device to protect the real identity associated with the malicious pseudonym. Furthermore, many existing schemes do not provide domain-level anonymity, where domain information of pseudonyms is public[4-6,13]. To enhance privacy protection, Gao et al.[14] proposed a domain-level anonymity scheme that offers stronger privacy protection compared to device-level anonymity schemes, ensuring that the public cannot determine the domain to which a pseudonym belongs. However, the scheme introduces centralized devices responsible for generating pseudonyms and tracing the real identities behind malicious pseudonyms for devices across domains, which poses a risk of single-point failure leading to privacy breaches for all devices. Tong et al.[15] utilized zero-knowledge proof technology to achieve domain-level anonymity for devices, but the TA exposed the domain information when distributing the token, causing the device to reveal which domain it belongs to in subsequent cross-domain communication. At present, it is crucial to propose a trusted and decentralized domain-level pseudonym management mechanism for cross-domain authentication scenarios. In addition, since most IIoT devices still have low computing and storage capabilities, it is also essential to design an efficient cross-domain authentication scheme for IIoT devices with limited resources to improve the universality of cross-domain authentication schemes. To address the various issues with existing cross-domain authentication schemes for IIoT, we propose a blockchain-based domain-level anonymous cross-domain authentication (DLCA) scheme that supports domain-level anonymity. The main contributions of our research are as follows:

(1) We combine group signatures with decentralized tracing technology with the cross-domain authentication architecture for IIoT, enabling all verification servers to collectively track the domain where malicious pseudonyms reside without introducing the risk of a single point of failure, thereby implementing a secure and reliable domain-level pseudonym management scheme.

(2) We design an efficient cross-domain authentication scheme based on elliptic curve cryptography (ECC) to for resource-constrained IIoT devices, which enables cross-domain authentication and session key negotiation for IIoT devices while ensuring data confidentiality.

(3) BAN logic proves the proposed scheme achieves the intended authentication goal, and the automated formal verification tool Scyther proves the scheme's security. In terms of efficiency, simulation results demonstrate that our scheme has lower computational and communication overhead during the cross-domain authentication and key negotiation phases.

1 Preliminary

In this section, in order to facilitate the understanding of our proposed DLCA scheme, we first

introduce ECC, along with related hard problems and group signatures. We then present the system model, threat model, and security requirements of the proposed scheme. Table 1 presents the meanings of the notations used in the paper.

Table 1 Notations

Notation	Description
p, q	Two large prime numbers
E	An elliptic curve
G , G_1, G_2	Cyclic additive group
P	A generator of the group G
G_T	Cyclic multiplication group
$\bar{e}$	Bilinear mapping
TA^A	The representative entity of domain A
(vsk_i, vpk_i)	Communication key pair of VS_i
(sdk_A, pdk_A)	Communication key pair of TA^A
SD_i^A	An IIoT device i of domain A
RID_i	Real identity of SD_i^A
PID_i	A pseudonym of SD_i^A
(sk_i, pk_i)	Key pair of PID_i
$\oplus$	Energy density
$\|$	Exclusive-OR operation
$H(\cdot)$, $h(\cdot)$	Hash function
$\mathrm{Enc}(\cdot)$, $\mathrm{Dec}(\cdot)$	AES encryption and decryption algorithms
VS^*	A representative node elected by blockchain
$\perp$	Error response

1.1 Elliptic curve cryptosystem and related hard problems

ECC is an asymmetric encryption algorithm based on the mathematical theory of elliptic curves, which offers the advantage of using shorter keys to achieve security comparable to or higher than RSA[16]. The basic knowledge of ECC and several related computational challenges are succinctly described as follows.

Let F_p be a finite field, which is determined by a prime number p. Let a set of elliptic curve points E over F_p be defined by the equation: $y^2 = x^3 + ax + b \pmod p$, where $a, b \in F_p$ and $(4a^3 + 27b^2) \bmod p \neq 0$. Let the point at infinity be O, then O and other points on E make up an additive elliptic curve group G with the order q and generator P. The scalar multiplication of E is defined as $mP = P + P + \cdots + P$ (m times), where $P \in G$, $m \in Z_q^*$, $m > 0$. The following is a concise description of ECC and several related hard problems:

(1) Elliptic curve discrete logarithm problem (ECDLP): $x \in Z_q^*$, $P, Q \in G$ on curve E. Given $Q = xP$, it is computational hard for a probabilistic polynomial-time (PPT) adversary to calculate x.

(2) Elliptic curve decisional Diffie-Hellman problem (ECDDHP): $x, y \in Z_q^*$, and $X = xP$, $Y = yP$, where $X, Y, Z \in G$ on curve E. Given X and Y, it is difficult to determine whether Z is equal to xyP for a PPT adversary[17].

1.2 Group signatures with decentralized tracing

The traditional group signature[18], in which only the group administrator can obtain the signer identity through the signature alone, has a high degree of centralized trust. Consequently, it is unsuitable for achieving secure domain-level anonymity in IIoT cross-domain authentication systems, which require decentralized trust management. Lu et al. implemented a mechanism for balancing anonymity and accountability in group signatures by decentralizing the actions of tracking signers[19]. In this paper, we use group signatures with decentralized tracing proposed by Lu et al.[19] to assist domain-level anonymity of IIoT devices. The algorithms involved in this scheme and their functions are described as follows:

(1) Setup(1^λ, n, t): Initialization algorithm, run by publisher VS^*. The input parameters are security parameters 1^λ, the number of openers n (the openers are all VS on the blockchain), and the threshold t that can successfully trace the signature. The output parameter is the system public parameter group_param.

(2) IKGen(1^λ, Param): Key generation algorithm, run by publisher VS^*. The input parameters are security parameters 1^λ, The output parameter is VS^*'s public key ipk and VS^*'s private key isk, ipk is public.

(3) OKGen($\mathrm{Gen}_1(1^\lambda, \mathrm{Param}, n, t)$, $\cdots$, $\mathrm{Gen}_n(1^\lambda, \mathrm{Param}, n, t)$): The algorithm is an interactive execution among all VS, at the end of the execution, every opener VS holding a tracking key

osk_i, osk_i is secret. The tracking public key opk_i is calculated based on published information during execution.

(4) Join($J_{TA_i}(1^\lambda)$, $J_{VS^*}(1^\lambda, isk)$): By the publisher VS^* and want to be a group of members of TA_i interaction to execute the algorithm. At the end of the execution, TA_i get the member private key sec_{TA_i} and member certificate $cert_{TA_i}$, and sec_{TA_i} is secret.

(5) Sign(sec_{TA_i}, $cert_{TA_i}$, M_{TA_i}): Group signature algorithm, run by group member TA_i. The input parameters are member private key sec_{TA_i}, member certificate $cert_{TA_i}$ and message M_{TA_i}. The output parameter is signature σ_{TA_i}.

(6) Verify(σ_{TA_i}, M_{TA_i}): The verification group signature algorithm, which can be performed by any user, in this paper is run by all VS on the blockchain. The input parameters are the message M_{TA_i} and the signature σ_{TA_i}, and the output parameters is a bit value. 0 indicates that the authentication fails, and 1 indicates that the authentication succeeds.

(7) Oshare(osk_i, σ_{TA_i}, M_{TA_i}): The algorithm is executed by each opener VS. The input parameters are tracking key osk_i, message M_{TA_i} and signatureσ_{TA_i}. Output parameters are $share_i$ or $\perp$, and $share_i$ is a part of the result obtained from each VS.

(8) Open(M_{TA_i}, σ_{TA_i}, $\mathbb{S}$): Input parameters are message m, signature σ_{TA_i} and collection $\mathbb{S}=\{share_1, share_2, \cdots, share_n\}$. The output parameter is the signer identity TA_i or $\perp$, and $\perp$ indicates trace failure.

For the detailed implementation of these algorithms, readers are referred to the original work[19]. This paper will reference the algorithm names when introducing the DLCA scheme.

1.3 System model

This section outlines the entities involved in the proposed DLCA scheme, which comprises four types: Representative authority of each domain (TA), consortium blockchain (BC), verification server (VS) and IIoT device (SD).

VS: Multiple domains or each domain deploys an authentication server, known as a VS, as a node in the consortium blockchain. The VS is responsible for verifying group signatures, uploading information to the blockchain, and jointly tracing the real identities behind malicious pseudonyms. VS^* denotes a representative node elected by all VSs on the blockchain, which is responsible for system initialization.

TA: Each domain appoints a representative entity known as TA, which can be a key generation center, certification authority, or private key generator within the domain. TA is responsible for generating pseudonyms for devices and applying group signatures to these pseudonyms. With assistance from the VS, TA uploads relevant domain information, pseudonyms and group signatures to the blockchain. TA is limited to query operations on the blockchain.

SD: Each domain comprises numerous IIoT devices known as SD with sensing, processing, and executing capabilities. These devices are responsible for specific manufacturing tasks or collaborating with IIoT devices from other domains to manufacture products. With assistance from TA, SD achieves mutual authentication and key negotiation with devices from other domains.

BC: The consortium blockchain, which is composed of all VS nodes, collectively maintains the distributed ledger for the alliance. This ledger primarily records public parameter information and pseudonym information uploaded by all VSs.

As shown in Fig. 1, the DLCA framework is divided into two layers based on the entity types: The domain layer and the blockchain network layer.

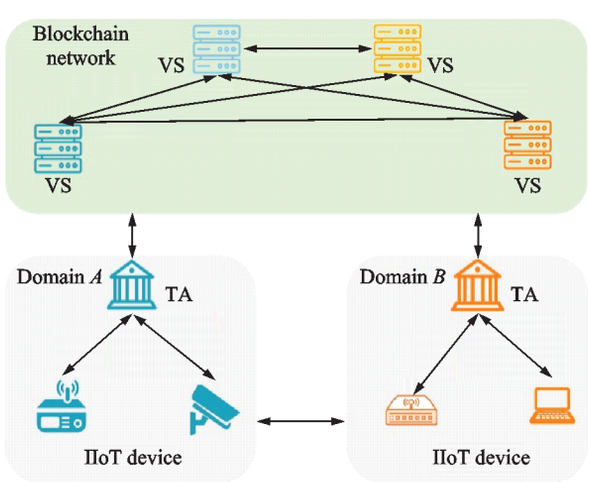

Fig.1 Framework of the DLCA scheme

1.4 Threat model

Considering the safety of the proposed scheme, we use the well-known Dolev-Yao adversary model[20]. Dolev and Yao have accurately depicted the attacker's behavior:

(1) Adversary can eavesdrop and intercept all messages passing through the IIoT network.

(2) Adversary can store intercepted or self-constructed messages associated with authentication in the IIoT network.

(3) Adversary can send intercepted or self-constructed messages associated with authentication in the IIoT network.

(4) Adversary can participate in the operation of the authentication protocol as a legitimate subject.

In addition, in order to reflect the security of the proposed scheme more comprehensively, on the basis of DY model, we introduce CK model to strengthen the attack capability of the adversary[21-22]. This allows an adversary to compromise a session key after the session completes, or to extract the long-term private key of a compromised IIoT device. In the proposed scheme, the default setting for TA is to maintain its original authentication mode with IIoT devices in the same domain, i.e., it utilizes the original encryption and authentication method within the domain for intra-domain communication. The proposed cross-domain authentication scheme must be capable of resisting common attacks under both the DY model and CK model.

2 Proposed Scheme

2.1 System initialization phase

VS* takes a safe large prime n, an elliptic curve E defined by: $y^2 = x^3 + ax + b \bmod p$, where $a, b \in F_p$, and selects an additive group G generated by the generator P, with the order of the prime q. Then it generates a random number $vsk^* \in Z_q^*$ and computes $vpk^* = vsk^* \cdot P$. It secretly keeps communication secret key vsk^* in the local database, publishes communication public key vpk^* . And select hash functions $H: \{0,1\}^* \rightarrow Z_q^*$; $h: \{0,1\}^* \rightarrow \{0,1\}^n$. Finally, the VS* uploads the eccparam $= \{G, E, q, p, P, H, h\}$ to the consortium blockchain. After the upload is successful, each VS on the blockchain choose random $vsk_i \in Z_q^*$, computes communication public key $vpk_i = vsk_i \cdot P$, and uploads vpk_i into blockchain. Each TA chooses random $sdk_i \in Z_q^*$ based public parameters on the blockchain, and calculates communication public key $pdk_i = sdk_i \cdot P$, then sends the domain of information and communication public key pdk_i to the corresponding VS, and with the help of VS, the message is uploaded to the blockchain. Then VS* executes Setup(1^λ, n, t) algorithm and IKGen(1^λ, Param) algorithm, each opener VS on the blockchain runs OKGen($\mathrm{Gen}_1(1^\lambda, \mathrm{Param}, n, t), \cdots, \mathrm{Gen}_n(1^\lambda, \mathrm{Param}, n, t)$) interactive algorithm. The TA_i who wants to become a group member then applies to VS*, and the two perform Join($J_{\mathrm{TA}_i}(1^\lambda), J_{\mathrm{VS}^*}(1^\lambda, isk)$) algorithm.

2.2 Pseudonym management phase

The proposed pseudonym management mechanism is mainly improved in three aspects. First of all, in terms of privacy, not only SD's device-level anonymity is realized, but also the domain information of pseudonym is hidden by using group signature technology to realize SD's domain-level anonymity. Secondly, in terms of security, in order not to cause a single point of failure, we use group signatures with decentralized tracing technology to avoid third-party central nodes in the scheme. Only the number of VSs that exceed the threshold can collectively get which domain the malicious pseudonym was obtained from. After determining the domain where the malicious pseudonym resides, TA of the domain discloses the real identity of the malicious pseudonym, and the public can verify the validity and authenticity of the real identity. This effectively prevents the TA from spoofing for the benefit of its own domain while tracking the true identity of malicious devices. Finally, in terms of efficiency, we define the length of each pseudonym to be 32 bits. Based on the hash function and its one-way feature, TA can generate multiple pseudonyms for a device at one time and upload them to the block-

chain, reducing the number of interactions with the blockchain, and the adversary cannot judge which pseudonyms belong to the same device according to the information in this batch of pseudonyms. This section takes domain A as an example to introduce the proposed pseudonym management mechanism in detail.

TA^A selects random numbers $d_i \in Z_q^*$, for SD_i^A ($i=1, 2, \cdots, n$) which belongs to its domain, calculates $PIDi = H(sdk_A * H(d_i \| RID_i \| ts) \cdot P)$, where $PID_i = (PID_{i,1} \| PID_{i,2} \| \cdots \| PID_{i,8})$, namely using the SHA-256 hash function to one-time generated eight pseudonyms for SD_i^A, ts is the validity period of the pseudonyms. TA^A chooses random numbers $z, l \in Z_q^*$, computes $L = l \cdot P, z_1 = z, z_2 = H(z_1), \cdots, z_8 = H(z_7)$. Then TA^A calculates $Z_1 = z_1 \cdot P, Z_2 = z_2 \cdot P, \cdots, Z_8 = z_8 \cdot P$. where $(z_1, z_2, \cdots, z_8)$ and $(Z_1, Z_2, \cdots, Z_8)$ are private key sk_i and public key pk_i of $(PID_{i,1}, PID_{i,2}, \cdots, PID_{i,8})$, respectively. The SD_i^A's pseudonyms and secret information d_i are recorded locally by TA^A. TA^A executes $Sign(sec_{TA^A}, cert_{TA^A}, M_{TA^A})$, where M_{TA^A} denotes that all authenticated pseudonyms which are generated by TA^A and the corresponding information (public key and validity period). TA^A sends $\{\sigma_{TA^A}, M_{TA^A}\}$ to a VS, then the VS sends message to other nodes using the P2P network, and each VS that has received the message performs Verify$(\sigma_{TA^A}, M_{TA^A})$ algorithm. When the number of nodes whose output result is 1, exceeds fifty percent, $\{\sigma_{TA^A}, M_{TA^A}\}$ will be uploaded to the blockchain.

After receiving the upload success response, TA^A sends message $M_1 = \{PID_i, (z_1, z_2, \cdots, z_8), (Z_1, Z_2, \cdots, Z_8), ts\}$ to SD_i^A through the original intra-domain channel. M_1 will be recorded locally by SD_i^A.

2.3 Cross-domain authentication key negotiation phase

In this section, SD_i^A in domain A uses the pseudonym PID_i (a legitimate pseudonym selected from $PID_{i,1}, PID_{i,2}, \cdots, PID_{i,8}$) and SD_j^B in domain B uses the pseudonym PID_j (a legitimate pseudonym selected from $PID_{j,1}, PID_{j,2}, \cdots, PID_{j,8}$) for cross-domain authentication and key negotiation. Fig.2 shows the process of cross-domain authentication and key negotiation between SD_i^A and SD_j^B. It is

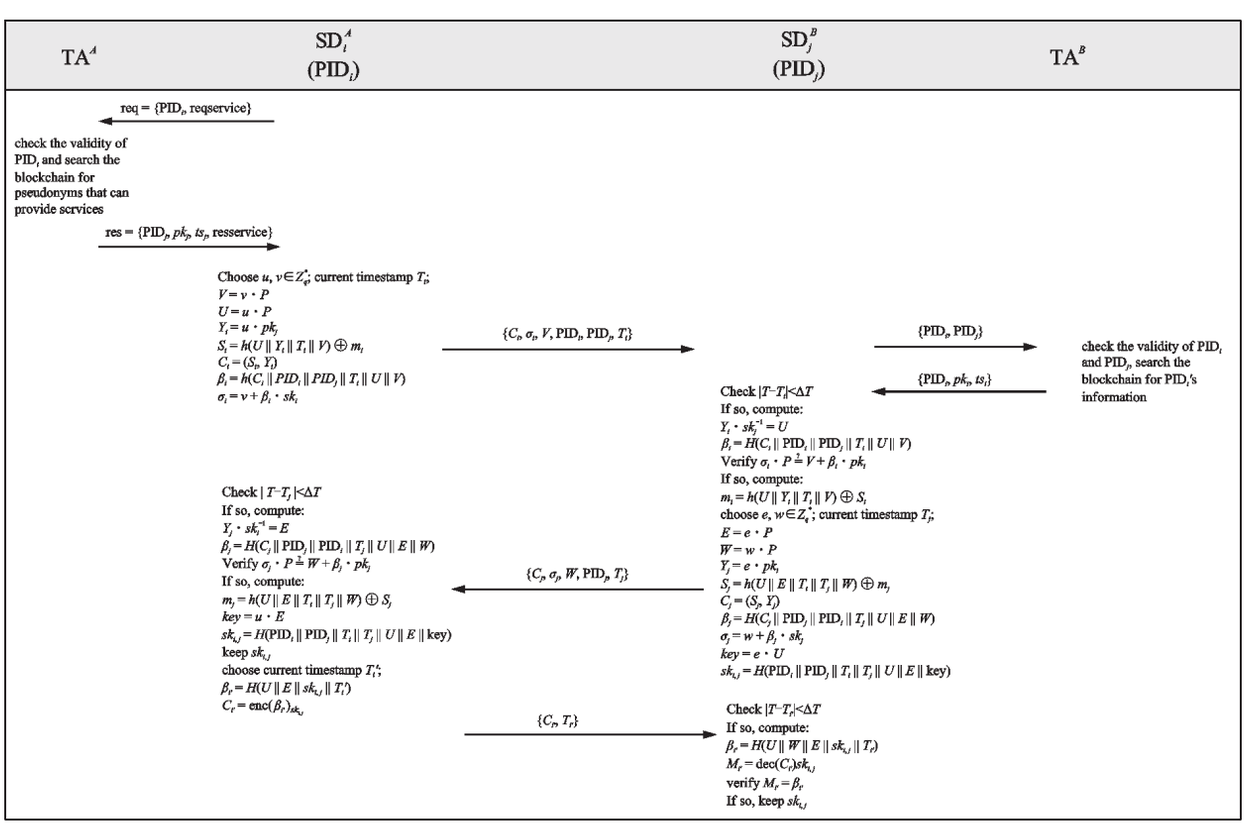

Fig.2 Process of cross-domain authentication and key negotiation between SD_i^A and SD_j^B

worth noting that at this stage, the both devices do not know the real identity of the other and which domain the other belongs to.

First, SD_i^A sends a cross-domain request req = {PID_i, reqservice} to TA^A. PID_i indicates the pseudonym under which SD_i^A will initiate cross-domain authentication, reqservice represents the service SD_i^A needs.

After receiving the req, TA^A verifies whether PID_i is a legitimate pseudonym, and queries the blockchain according to reqservice, and sends pseudonym PID_j that can provide the service to SD_i^A in the form of res = {PID_j, pk_j, ts_j, resservice}, where resservice represents the service that PID_j can provide, and ts_j is the validity period of the PID_j. After receiving the res, SD_i^A selects random numbers $u, v \in Z_q^*$, and current timestamp T_i. The variable m_i is description of the permissions that SD_i^A requests from SD_j^B. According to the selected parameters above, SD_i^A performs the calculation for Eqs.(1—6).

$$V = v \cdot P \tag{1}$$

$$U = u \cdot P \tag{2}$$

$$Y_i = u \cdot pk_j \tag{3}$$

$$S_i = H(U \| Y_i \| T_i \| V) \oplus m_i \tag{4}$$

$$\beta_i = H(C_i \| \mathrm{PID}_i \| \mathrm{PID}_j \| T_i \| U \| V) \tag{5}$$

$$\sigma_i = v + \beta_i \cdot sk_i \tag{6}$$

Denote (S_i, Y_i) as C_i. Finally, SD_i^A sends $M_{i,1} = \{C_i, \sigma_i, V, \mathrm{PID}_i, \mathrm{PID}_j, T_i\}$ to SD_j^B.

SD_j^B after receiving $M_{i,1}$ verifies the freshness of timestamp T_i by check whether $|T - T_i| < \Delta T$ is true, where T is the timestamp of receiving the message, and then sends {PID_i, PID_j} to TA^B. TA^B queries the relevant information of PID_i on the blockchain, determines that PID_i and PID_j are legitimate pseudonyms, and sends {PID_i, pk_i, ts_i} to SD_j^B, where ts_i is the validity period of the PID_i. Finally, SD_j^B computes

$$Y_i \cdot sk_j^{-1} = U \tag{7}$$

$$\beta_i = H(C_i \| \mathrm{PID}_i \| \mathrm{PID}_j \| T_i \| U \| V) \tag{8}$$

verifing whether Eq.(9) is established.

$$\sigma_i \cdot P \stackrel{?}{=} V + \beta_i \cdot pk_i \tag{9}$$

After verification, SD_j^B selects random numbers $e, w \in Z_q^*$, current timestamp T_j, m_j is description of the permissions that SD_j^B can give SD_i^A. Using the selected parameters, SD_j^B computes Eq.(10) to Eq.(18).

$$m_i = H(U \| Y_i \| T_i \| V) \oplus S_i \tag{10}$$

$$E = e \cdot P \tag{11}$$

$$W = w \cdot P \tag{12}$$

$$Y_j = e \cdot pk_i \tag{13}$$

$$S_j = H(U \| E \| T_i \| T_j \| W) \oplus m_j \tag{14}$$

$$\beta_j = H(C_j \| \mathrm{PID}_j \| \mathrm{PID}_i \| T_j \| U \| E \| W) \tag{15}$$

$$\sigma_j = w + \beta_j \cdot sk_j \tag{16}$$

$$\mathrm{key} = e \cdot U \tag{17}$$

$$sk_{i,j} = H(\mathrm{PID}_i \| \mathrm{PID}_j \| T_i \| T_j \| U \| E \| key) \tag{18}$$

where $sk_{i,j}$ is the negotiation key of SD_i^A and SD_j^B, and denote (S_j, Y_j) as C_j. Finally, SD_j^B sends $M_{j,1} = \{C_j, \sigma_j, W, \mathrm{PID}_j, T_j\}$ to SD_i^A.

After receiving the message $M_{j,1}$, SD_i^A verifies the freshness of timestamp T_j by checking whether $|T - T_j| < \Delta T$ is true, where T is the timestamp of receiving the message. SD_i^A derives the following equations.

$$Y_j \cdot sk_i^{-1} = E \tag{19}$$

$$\beta_j = H(C_j \| \mathrm{PID}_j \| \mathrm{PID}_i \| T_j \| U \| E \| W) \tag{20}$$

And SD_i^A verifies whether Eq.(21) is established.

$$\sigma_j \cdot P \stackrel{?}{=} W + \beta_j \cdot pk_j \tag{21}$$

After verification, SD_i^A calculates

$$m_j = H(U \| E \| T_i \| T_j \| W) \oplus S_j \tag{22}$$

$$\mathrm{key} = u \cdot E \tag{23}$$

$$sk_{i,j} = H(\mathrm{PID}_i \| \mathrm{PID}_j \| T_i \| T_j \| U \| E \| \mathrm{key}) \tag{24}$$

SD_i^A takes the $sk_{i,j}$ as the session key between PID_i and PID_j, and saves it locally. Then SD_i^A selects current timestamp T_i' and computes

$$\beta_{i'} = H(U \| E \| sk_{i,j} \| T_i') \tag{25}$$

$$C_{i'} = \mathrm{enc}(\beta_{i'})_{sk_{i,j}} \tag{26}$$

Finally, SD_i^A sends $M_{i,2} = \{C_{i'}, T_i'\}$ to SD_j^B.

After receiving the message $M_{i,2}$, SD_j^B verifies the freshness of timestamp T_i' by checking whether $|T - T_i'| < \Delta T$ is true, where T is the timestamp of receiving the message. SD_j^B derives the following equations.

$$\beta_{i'} = H(U \| E \| sk_{i,j} \| T_i') \tag{27}$$

$$M_{i'} = \mathrm{dec}(C_{i'})_{sk_{i,j}} \tag{28}$$

Then SD_j^B verifies whether Eq. (29) is established.

$$M_{i'} \stackrel{?}{=} \beta_{i'} \tag{29}$$

After verification, SD_j^B takes the $sk_{i,j}$ as the session key between PID_i and PID_j, and saves it locally.

2.4 Illegal pseudonym tracing and identity revocation phase

If PID_i is illegal, $VS_i (i = 1, 2, \cdots, n)$ will look for group signatures that contain PID_i on the blockchain. Then all VS working together will execute $\text{Oshare}(osk_i, \sigma_{TA^A}, M_{TA^A})$ algorithm and $\text{Open}(M_{TA^A}, \sigma_{TA^A}, \mathbb{S})$ algorithm to trace the TA^A who generated the illegal pseudonym PID_i. Notably, the parameters M_{TA^A} and σ_{TA^A} are the plaintext on the blockchain containing the malicious pseudonym PID_i and the group signature of that plaintext respectively. Then TA^A publishes PID_i's real identity RID_i and corresponding secret data d_i, as well as all pseudonyms generated for RID_i at the same time. VS calculates $PID_i = H(H(d_i \| RID_i \| ts) \cdot pdk_A)$ based on the information published by TA^A to verify that a 32-bit continuous data segment in the output result is the malicious pseudonym, and then queries the remaining seven pseudonyms in the blockchain based the output result. If the result of the query is consistent with the information published by TA^A, VS^* will upload the information of the illegal device to the blacklist on the blockchain. If too many malicious pseudonyms generated by TA^A are recorded in the blacklist, the reputation of domain A is affected.

3 Security Analyses

3.1 Formal security analysis with BAN logic

BAN logic is a pioneering work in the formal analysis of security protocols and is widely used because of its simplicity and intuition[23-27]. In this paper, we use BAN logic to formally analyze the proposed cross-domain authentication protocol. The notations in BAN logic are described in Table 2, and the logical rules are shown in Table 3.

Table 2 Ban logic notations

Notation	Description
$A \mid\equiv X$	A believes X
$A \mid\sim X$	A once said X or A had sent message X
$A \lhd X$	A sees X
$A \mid\Rightarrow X$	A has jurisdiction over X
$\#(X)$	X is fresh
$\overset{K}{\mapsto} A$	K is a public key of A
$A \overset{K}{\leftrightarrow} B$	K is the key shared between A and B
$A \overset{X}{\Leftrightarrow} B$	X is the secret shared between A and B
$\{X\}_K$	X is encrypted with K
$\{X\}_{K^{-1}}$	X is signed with the private key K^{-1}

Table 3 Ban logic rules

R	Rule	Notation	Description
$R1$	Message-meaning rule	$\dfrac{A \mid\equiv \overset{K}{\mapsto} B, A \lhd \{X\}_{K^{-1}}}{A \mid\equiv B \mid\sim X}$	If A believes that the K is public key of B and sees a message X encrypted under K^{-1}, then A believes that B once said X
$R2$	Nonce-verification rule	$\dfrac{A \mid\equiv \#(X), A \mid\equiv B \mid\sim X}{A \mid\equiv B \mid\equiv X}$	If A believes that X is fresh and B once said X, then A believes that B believes X
$R3$	Jurisdiction rule	$\dfrac{A \mid\equiv B \mid\Rightarrow X, A \mid\equiv B \mid\equiv X}{A \mid\equiv X}$	If A believes that B has jurisdiction over X and B believes X, then A believes X
$R4$	Fresh rule	$\dfrac{A \mid\equiv \#(X)}{A \mid\equiv \#(X, T)}$	If A believes that X is fresh, then A believes that formulae (X, T) is fresh
$R5$	Belief rule	$\dfrac{A \mid\equiv X, A \mid\equiv T}{A \mid\equiv (X, T)}$	If A believes in X and T individually, then A believes that collective formula (X, T)
$R6$	Receiving rule	$\dfrac{A \lhd \{X\}_K, A \mid\equiv \overset{K}{\mapsto} A}{A \lhd X}$	If A receives encrypted X and believes that the K is public key of A, then A receives X

(1) Expected goals declaration

To prove that our scheme is secure, we need to prove that the following beliefs hold

$$G1{:}\mathrm{SD}_i|\equiv \mathrm{SD}_j|\equiv \mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j$$

$$G2{:}\mathrm{SD}_i\left|\equiv \mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j\right.$$

$$G3{:}\mathrm{SD}_i\left|\equiv \#(\mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j)\right.$$

$$G4{:}\mathrm{SD}_j|\equiv \mathrm{SD}_i|\equiv \mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j$$

$$G5{:}\mathrm{SD}_j\left|\equiv \mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j\right.$$

$$G6{:}\mathrm{SD}_j\left|\equiv \#(\mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j)\right.$$

(2) Message formalization

Message formalization is to specify the exchanged messages. In the proposed scheme, U and E are secret values shared between SD_i and SD_j. the formalized message is as follows

$$M1{:}\mathrm{SD}_j \lhd \{\{U\}_{pk_j}, T_i, \{T_i, U\}_{pk_i^{-1}}\}$$

$$M2{:}\mathrm{SD}_i \lhd \{\{E\}_{pk_i}, T_j, \{T_i, T_j, U, E\}_{pk_j^{-1}}\}$$

$$M3{:}\mathrm{SD}_j \lhd \{T_i', \{T_i', U, E, sk_{i,j}\}\, sk_{i,j}\}$$

(3) Initial assumptions declaration

As described in our protocol, we have the following assumptions

$$A1{:}\mathrm{SD}_i\left|\equiv \overset{pk_i}{\rightarrow} \mathrm{SD}_j\right.$$

$$A2{:}\mathrm{SD}_i\left|\equiv \overset{pk_i}{\rightarrow} \mathrm{SD}_i\right.$$

$$A3{:}\mathrm{SD}_j\left|\equiv \overset{pk_j}{\rightarrow} \mathrm{SD}_j\right.$$

$$A4{:}\mathrm{SD}_j\left|\equiv \overset{pk_j}{\rightarrow} \mathrm{SD}_i\right.$$

$$A5{:}\mathrm{SD}_i|\equiv \#(T_i), \#(U), \#(T_j), \#(T_i')$$

$$A6{:}\mathrm{SD}_j|\equiv \#(T_j), \#(E), \#(T_i), \#(T_i')$$

$$A7{:}\mathrm{SD}_i|\equiv \mathrm{SD}_j|\Rightarrow E$$

$$A8{:}\mathrm{SD}_j|\equiv \mathrm{SD}_i|\Rightarrow U$$

$$A9{:}\ \mathrm{SD}_i|\equiv T_i, U, T_i'$$

$$A10{:}\mathrm{SD}_j|\equiv T_j, E$$

(4) Logic verification

Finally, we use the BAN logic to prove that the proposed scheme achieves the beliefs, and the detail process is described below:

From $M1$, $A4$ and $R6$, we deduce

$S1{:}\ \mathrm{SD}_j \lhd \{U, T_i\}$

From $M1$, $A3$ and $R1$, we deduce

$S2{:}\ \mathrm{SD}_j|\equiv \mathrm{SD}_i|\sim \{U, T_i\}$

From $A6$ and $R4$, we deduce

$S3{:}\ \mathrm{SD}_j|\equiv \#(U, T_i)$, Because $sk_{i,j}=H(\mathrm{PID}_i\|\mathrm{PID}_j\|T_i\|T_j\|U\|E\|e\cdot U)$, so get

$$G6{:}\mathrm{SD}_j\left|\equiv \#(\mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j)\right.$$

From $S3$, $S2$ and $R2$, we deduce

$S4{:}\ \mathrm{SD}_j|\equiv \mathrm{SD}_i|\equiv \{U, T_i\}$

From $S4$, $A8$ and $R3$, we deduce

$S5{:}\ \mathrm{SD}_j|\equiv \{U, T_i\}$

From $S5$, $A10$, $R5$, we deduce

$$G5{:}\mathrm{SD}_j\left|\equiv \mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j\right.$$

From $M3$, $G5$, $R1$, we deduce

$$S6{:}\ \mathrm{SD}_j\left|\equiv \mathrm{SD}_i|\sim \mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j\right.$$

From $S6$, $G6$, $R2$, we deduce

$$G4{:}\ \mathrm{SD}_j|\equiv \mathrm{SD}_i|\equiv \mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j$$

From $M2$, $A2$ and $R6$, we deduce

$S7{:}\ \mathrm{SD}_i \lhd \{E, T_j\}$

From $M2$, $A1$ and $R1$, we deduce

$S8{:}\ \mathrm{SD}_i|\equiv \mathrm{SD}_j|\sim \{T_i, T_j, U, E\}$

From $A6$ and $R4$, we deduce

$S9{:}\ \mathrm{SD}_i|\equiv \#(T_i, T_j, U, E)$

because $sk_{i,j}=H(\mathrm{PID}_i\|\mathrm{PID}_j\|T_i\|T_j\|U\|E\|u\cdot E)$, so get

$$G3{:}\mathrm{SD}_i\left|\equiv \#(\mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j)\right.$$

From $S8$, $S9$ and $R2$, we deduce

$S10{:}\ \mathrm{SD}_i|\equiv \mathrm{SD}_j|\equiv \{T_i, T_j, U, E\}$

Because SD_i believe SD_j believe all constitute $sk_{i,j}$ secret value U and E, so get

$$G1{:}\ \mathrm{SD}_i|\equiv \mathrm{SD}_j|\equiv \mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j$$

From $S10$, $R5$, we deduce

$S11{:}\mathrm{SD}_i|\equiv \mathrm{SD}_j|\equiv \{T_j, E\}$

From $S11$, $A7$, $R3$, we deduce

$S12{:}\mathrm{SD}_i|\equiv \{T_j, E\}$

From $S12$, $A9$ and $R5$, we deduce

$$G2{:}\ \mathrm{SD}_i\left|\equiv \mathrm{SD}_i \overset{sk_{i,j}}{\leftrightarrow} \mathrm{SD}_j\right.$$

3.2 Informal security analysis

The following sections explain the attack resil-

ience of the proposed DLCA scheme against different attacks.

(1) Mutual authentication. Our proposed DLCA scheme supports two-way identity authentication between IIoT devices to ensure the integrity of authentication. In the cross-domain authentication key negotiation phase, SD_i^A and SD_j^B send the selected random tokens U and E to each other through the encryption authentication channel. After receiving the ciphertext, the receiver uses its own private key to decrypt the token and uses the sender's public key to verify the signature to determine whether the token is generated by a legitimate pseudonym. An adversary cannot forge a legitimate token without obtaining a legitimate pseudonym's private key.

(2) Resistance to replay attack. In the cross-domain authentication key negotiation phase, SD_i^A and SD_j^B put the timestamp T_i, T_j, and random tokens U, E in the request and reply, respectively. If an attacker inserts a new timestamp into the message, the signature will be invalidated, so the intercepted message is not conducive to the attacker's replay attack.

(3) Resistance to impersonation attack. In order to launch a successful impersonation attack, the adversary needs to obtain a legitimate pseudonym's private key. Without the private key, forging a verifiable signature is computationally infeasible due to the hardness of the ECDLP and the one-way property of the hash function. Consequently, an adversary cannot successfully impersonate a legitimate IIoT device.

(4) Adherence to anonymity. In our proposed DLCA scheme, devices always authenticate and authorize each other under pseudonyms, and since the real identity of the device is hidden by the random number d_i generated by the TA of the domain to which it belongs, only the TA in the cooperative domain can restore the real identity of the device after obtaining d_i. Due to the one-way nature of the hash function and the difficulty of ECDLP, the adversary cannot recover the d_i to obtain the real identity of the device. Therefore, the scheme meets the requirement of anonymity.

(5) Resistance to eavesdropping attacks. If an adversary can eavesdrop on cross-domain authentication information from an unsecured channel, he could abuse the content to impersonate any legitimate device by modifying or replaying the messages. For instance, an adversary sniffs the channel between SD_i^A and SD_j^B in the cross-domain authentication key negotiation phase. i,e. the adversary accesses the communicate message $\{C_i, \sigma_i, V, PID_i, T_i\}$, but the adversary cannot be able to get any critical information. This is due to the communicate message is encrypted by utilizing receiver's public key, and timestamps T_i and sender's signatures have been added to prevent replay attacks and impersonation attacks. As a result, DLCA successfully resists eavesdropping attacks.

(6) Supports perfect forward secrecy. The proposed scheme is in line with perfect forward secrecy, and even when any private key is exposed, the session key negotiated by the two devices cannot be obtained. This is because, in addition to leaking the long-term key, short-term nonces (u, e generated and saved locally by the device during the previous authentication process) are required to decrypt the content of the previous session. As a result, our DLCA scheme supports perfect forward security.

(7) Resistance to ephemeral secret leakage attack. As mentioned above, the adversary must obtain short-term nonces u and e to calculate the session key, and if the adversary wants to successfully obtain short-term nonces, he needs to be able to access the long term secrets (sk_i, sk_j) to decrypt to get U or E, then the ECDLP needs to be solved to obtain short-term nonces u and e.

(8) Resistance to known session key threat. The proposed scheme provides mutual authentication, and in each round of authentication, the session key is negotiated between devices based on long-term secrets and short-term nonces. Therefore, even if one session key is compromised, it cannot affect the security of past session keys.

3.3 Formal security verification using scyther

In this section, the formal analysis of DLCA is carried out using Scyther. Scyther is a tool for the

automatic verification of security protocols[28]. Scyther has been successfully used for the analysis and design of protocols, and has also been used for theoretical research and teaching. The cross-domain authentication key negotiation phase of DLCA scheme is written in Scyther using security protocol description language (SPDL), validating various authentication properties. As exhibited in Fig. 3, through testing the secret of plain message m_i, or m_j and tokens U or E generated by SD_i^A or SD_j^B, the secret of key negotiated by both devices, and the Alive, Weakagree, Nisynch, and Niagree of entities, respectively. The verification results in the fifth column indicate that the verification passed, and the sixth column further explains that the verification result does not find an attack path within the bounded query, i.e., the proposed DLCA scheme is secure.

Fig.3 Verified results of DLCA with Scyther during the cross-domain authentication key negotiation phase

4 Performance Evaluation

In this section, we analyze the computational and communication costs of the proposed scheme. To further demonstrate the advantage of DLCA in consideration of performance, we compare with four related schemes[4,7,12,14]. For the convenience of expression, we call the scheme BASA in Ref.[4], the scheme BCFA in Ref.[7], the scheme CPMA in Ref.[12], and the scheme PACA in Ref.[14], respectively. The experimental platform is composed of an Intel i5-12500H CPU with 100 MHz frequency, 16 GB RAM, and the operation system is Ubuntu 22.04 in WSL. In order to compare the computational cost of cross-domain authentication schemes, evaluation of single cryptographic operation are simulated with the Miracl Core. For bilinear pairings based cross domain authentication schemes for IIoT, we use the 256-bit BarretoNaehrig curve. For ECC based cross domain authentication schemes for IIoT, we use SM2 elliptic curve public key cryptography algorithm recommends curve parameter. For convenience, we define some notations about execution time as follows, and the operations and their overhead are listed in Table 4.

Table 4 Computational cost of cryptographic operations

Cryptographic operation	Execution time/ms
T_{bp}	1.252 3
$T_{sm1\text{-}bp}$	0.243 2
$T_{sm2\text{-}bp}$	0.649 4
$T_{pa1\text{-}bp}$	0.001 7
$T_{pa2\text{-}bp}$	0.004 5
$T_{exp\text{-}bp}$	0.979 1
T_{mtp}	1.702 7
$T_{sm\text{-}ecc}$	0.528 1
$T_{pa\text{-}ecc}$	0.002 0
T_h	0.001 0
T_{aes}	0.001 2

T_{bp}: The execution time of a bilinear pairing operation $\bar{e}(S, T)$, where $S \in G_1$ and $T \in G_2$.

$T_{\text{sm1-bp}}$: The execution time of a scale multiplication operation $x \cdot P_1$ related to the bilinear pairing, where $x \in Z_N^*$ and $P_1 \in G_1$.

$T_{\text{sm2-bp}}$: The execution time of a scale multiplication operation $y \cdot P_2$ related to the bilinear pairing, where $y \in Z_N^*$ and $P_2 \in G_2$.

$T_{\text{pa1-bp}}$: The execution time of a point addition operation $S_1 + S_2$ related to the bilinear pairing, where $S_1, S_2 \in G_1$.

$T_{\text{pa2-bp}}$: The execution time of a point addition operation $T_1 + T_2$ related to the bilinear pairing, where $T_1, T_2 \in G_2$.

$T_{\text{exp-bp}}$: The execution time of an exponential operation Q^z related to the bilinear pairing, where $Q \in G_T$ and $z \in Z_N^*$.

T_{mtp}: The execution time of a hash-to-point operation related to the bilinear pairing where the hash function maps a string to a point of G_2.

$T_{\text{sm-ecc}}$: The execution time of a scale multiplication operation $z \cdot P$ related to the ECC, where $z \in Z_q^*$ and $P \in G$.

$T_{\text{pa-ecc}}$: The execution time of a point addition operation $\overline{S} + \overline{T}$ related to the ECC, where $\overline{S}, \overline{T} \in G$.

T_{h}: The execution time of a SHA256 hash function operation.

T_{aes}: The execution time of AES-ECB encryption and decryption.

The computational cost and the communication cost for all schemes were measured and presented in Fig.4, and the calculation cost of a cross-domain authentication represents the sum of the calculation cost of each entity in the process of an authentication. Table 5 presents statistics of the cryptographic

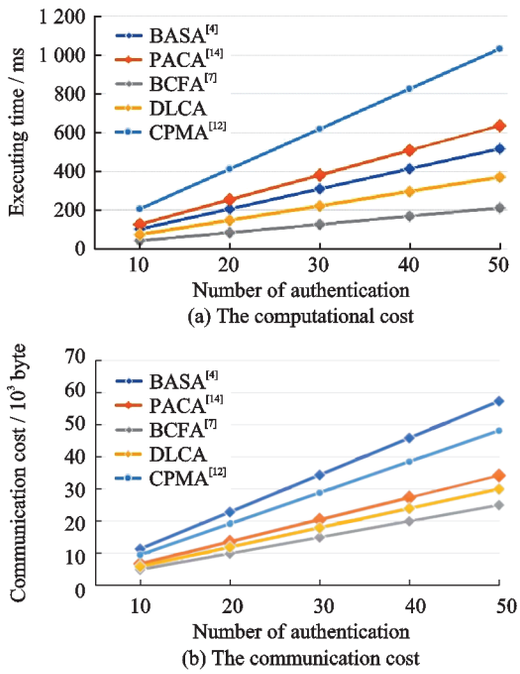

Fig.4 The computational and the communication cost comparison

operations consumed by each scheme during the cross-domain authentication key negotiation phase. The reason we do not consider the pseudonym management phase is that it can be executed at any time when the device is idle and does not require real-time. In the cross-domain authentication key negotiation phase, BASA and BCFA are considered to be executed by two devices in different domains, and the calculation of PACA scheme involves four entities. CPMA is mainly performed by a device in a domain and gateways in t domains, we define $t = 5$. Our proposed DLCA consumes less computational time than BASA, PACA and CPMA, with BCFA consuming the least computational time. However, BCFA, BASA and PACA do not provide data confidentiality until the session key is established, and DLCA always maintains data confidentiality.

Table 5 Comparison of computation cost

Scheme	Computation cost/ms
BASA	$2(T_{\text{bp}} + T_{\text{sm1-bp}} + T_{\text{sm2-bp}} + T_{\text{pa2-bp}} + 2T_{\text{exp-bp}} + 2T_{\text{sm-ecc}} + T_{\text{h}}) \approx 10.331\,2$ ms
CPMA	$5T_{\text{bp}} + (t+1)T_{\text{mtp}} + (2t+7)T_{\text{sm1-bp}},\ t=5 \approx 20.617\,1$ ms
BCFA	$2(4T_{\text{sm-ecc}} + T_{\text{pa-ecc}} + 2T_{\text{h}} + T_{\text{aes}}) \approx 4.235\,2$ ms
PACA	$2(11T_{\text{sm-ecc}} + 2T_{\text{pa-ecc}} + 7T_{\text{h}}) \approx 11.640\,2$ ms
DLCA	$2(7T_{\text{sm-ecc}} + T_{\text{pa-ecc}} + 6T_{\text{h}}) + T_{\text{aes}} \approx 7.412\,6$ ms

We assume that the length of the timestamp involved in all relevant schemes is 4 bytes, and the real identity of each entity is 4 bytes. Since the sizes of p and q are 32 bytes, then the element in G is 64 bytes and the length of element in ring Z_q^* is 32 bytes. Since the length of the element in ring Z_N^* is 32 bytes, and the size of $\bar{p}$ is 32 bytes, then the element in group G_1 is 64 bytes. The element in group G_2 is 128 bytes. The communication cost of related cross domain authentication schemes for IIoT is presented in Table 6. DLCA is less costly in communication than BASA, CPMA and CPMA. BCFA is the least expensive to communicate, but it does not provide device anonymity.

Table 6 Comparison of communication cost

Scheme	Communication cost/byte
BASA	1 152
CPMA	968
BCFA	504
PACA	688
DLCA	604

5 Conclusions

In this paper, a domain-level anonymous cross-domain authentication scheme based on blockchain for industrial internet of things was proposed, which we called the scheme DLCA. The consortium blockchain provided a trusted platform for various IIoT domains to share domain information and parameters. DLCA not only realized domain-level anonymity of SD, but also considered that in order to avoid a single point of failure, the public can trace the real identity of the malicious pseudonym. DLCA achieved the desired authentication goal in BAN logical proof and verifies the security of the scheme using Scyther. Because the pseudonym management phase can be executed in advance, it does not need to occupy network resources during cross-domain authentication. In the cross-domain authentication key negotiation phase, compared with other schemes in terms of communication cost and calculation cost, the proposed DLCA scheme has advantages.

References

[1] MING Y, YANG P, MAHDIKHANI H, et al. A secure one-to-many authentication and key agreement scheme for industrial IoT[J]. IEEE Systems Journal, 2023, 17(2): 2225-2236.

[2] HUO R, ZENG S Q, WANG Z H, et al. A comprehensive survey on blockchain in industrial internet of things: Motivations, research progresses, and future challenges[J]. IEEE Communications Surveys & Tutorials, 2022, 24(1): 88-122.

[3] SERROR M, HACK S, HENZE M, et al. Challenges and opportunities in securing the industrial internet of things[J]. IEEE Transactions on Industrial Informatics, 2021, 17(5): 2985-2996.

[4] SHEN M, LIU H S, ZHU L H, et al. Blockchain-assisted secure device authentication for cross-domain industrial IoT[J]. IEEE Journal on Selected Areas in Communications, 2020, 38(5): 942-954.

[5] TONG F, CHEN X, WANG K M, et al. CCAP: A complete cross-domain authentication based on blockchain for internet of things[J]. IEEE Transactions on Information Forensics and Security, 2022, 17: 3789-3800.

[6] WANG F Q, CUI J, ZHANG Q Y, et al. Blockchain-based light weight message authentication for edge-assisted cross-domain industrial internet of things[J]. IEEE Transactions on Dependable and Secure Computing, 2024, 21(4): 1587-1604.

[7] DONG J N, XU G X, MA C, et al. Block chain-based certificate-free cross-domain authentication mechanism for industrial internet[J]. IEEE Internet of Things Journal, 2024, 11(2): 3316-3330.

[8] ZHANG Y J, LUO Y H, CHEN X, et al. A lightweight authentication scheme based on consortium blockchain for cross-domain IoT[J]. Security and Communication Networks, 2022, 9686049:15.

[9] DIB O, BROUSMICHE K L, DURAND A, et al. Consortium blockchains: Overview, applications and challenges[J]. International Journal on Advances in Telecommunications, 2018, 11(1): 51-64.

[10] CHEN J, ZHAN Z Y, HE K, et al. XAuth: Efficient privacy-preserving cross-domain authentication[J]. IEEE Transactions on Dependable and Secure Computing, 2022, 19(5): 3301-3311.

[11] CUI J, LIU N, ZHANG Q Y, et al. Efficient and anonymous cross-domain authentication for IIoT based on blockchain[J]. IEEE Transactions on Network Science and Engineering, 2023, 10(2): 899-910.

[12] ZHONG H, GU C D, ZHANG Q Y, et al. Conditional privacy-preserving message authentication scheme for cross-domain industrial internet of things[J]. Ad Hoc Networks, 2023, 144: 103137.

[13] CUI J, ZHU Y H, ZHONG H, et al. Efficient blockchain-based mutual authentication and session key agreement for cross-domain IIoT[J]. IEEE Internet of Things Journal, 2024, 11(9): 16325-16338.

[14] GAO B Y, YAN H R, TIAN R. A privacy-aware cross-domain device authentication scheme for IIoT based on blockchain[C]//Proceedings of 2021 IEEE 23rd International Conference on High Performance Computing & Communications; the 7th International Conference on Data Science & Systems; the 19th International Conference on Smart City; the 7th International Conference on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys). [S. l.]: IEEE, 2021: 561-570.

[15] TONG F, CHEN X, HUANG C, et al. Blockchain-assisted secure intra/inter-domain authorization and authentication for internet of things[J]. IEEE Internet of Things Journal, 2023, 10(9): 7761-7773.

[16] LENSTRA A K, KLEINJUNG T, THOMÉ E. Universal security: From bits and mips to pools, lakes and beyond[C]//Proceedings of Number Theory and Cryptography: Papers in Honor of Johannes Buchmann on the Occasion of His 60th Birthday. Berlin, Heidel berg: Springer, 2013: 121-124.

[17] CANETTI R. Decisional Diffie-Hellman assumption[M]//Encyclopedia of Cryptography and Security. [S.l.]: Springer, 2005: 140-142.

[18] CHAUM D, VAN HEYST E. Group signatures[C]//Proceedings of Advances in Cryptology—Eurocrypt'91: Workshop on the Theory and Application of Cryptographic Techniques. Brighton, UK: Springer, 1991: 257-265.

[19] LU T T, LI J T, ZHANG L, et al. Group signatures with decentralized tracing[C]//Proceedings of the Information Security and Cryptology: the 15th International Conference, Inscrypt 2019. Nanjing, China: Springer, 2020: 435-442.

[20] DOLEV D, YAO A. On the security of public key protocols[J]. IEEE Transactions on Information Theory, 1983, 29(2): 198-208.

[21] CANETTI R, KRAWCZYK H. Analysis of key-exchange protocols and their use for building secure channels[C]//Proceedings of International conference on the theory and applications of cryptographic techniques. [S.l.]: Springer, 2001: 453-474.

[22] CANETTI R, KRAWCZYK H. Universally composable notions of key exchange and secure channels [C]//Proceedings of International Conference on Advances in cryptology-EUROCRYPT 2002. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002: 337-351.

[23] BURROWS M, ABADI M, NEEDHAM R. A logic of authentication[J]. ACM Transactions on Computer Systems (TOCS), 1990, 8(1): 18-36.

[24] ZHANG Q Y, WU J, ZHONG H, et al. Efficient anonymous authentication based on physically unclonable function in industrial internet of things[J]. IEEE Transactions on Information Forensics and Security, 2023, 18: 233-247.

[25] XU H, HSU C, HARN L, et al. Three-factor anonymous authentication and key agreement based on fuzzy biological extraction for industrial internet of things[J]. IEEE Transactions on Services Computing, 2023, 16(4): 3000-3013.

[26] WANG M M, RUI L L, YANG Y, et al. A blockchain-based multi-CA cross-domain authentication scheme in decentralized autonomous network[J]. IEEE Transactions on Network and Service Management, 2022, 19(3): 2664-2676.

[27] WANG C Y, WANG D, DUAN Y H, et al. Secure and lightweight user authentication scheme for cloud-assisted internet of things[J]. IEEE Transactions on Information Forensics and Security, 2023, 18: 2961-2976.

[28] CAO J, MA M D, FU Y L, et al. CPPHA: Capability-based privacy-protection handover authentication mechanism for SDN-based 5G HetNets[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 18(3): 1182-1195.

Author

The first/corresponding author Ms. **LIANG Yufeng** is currently Ph.D. candidate at Nanjing University of Aeronautics and Astronautics. Her research interests focus on the information security and applied cryptography.

Author contributions Ms. **LIANG Yufeng** carried out the literature review, designed the study, performed the technique analysis and drafted the initial manuscript. Dr. **SUN Lu** contributed to methodology development and analyzed of critical vevison of the manuscript. Both authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing interests.

(Production Editor: LIU Yandong)

基于区块链的工业物联网域级匿名跨域认证方案

梁玉凤，孙 璐

(南京航空航天大学计算机科学与技术学院/软件学院，南京 211106，中国)

摘要：工业物联网(Industrial internet of things, IIoT)的快速发展为配备工业物联网技术的工厂带来了巨大的利益，每个工厂都代表一个工业物联网域。越来越多的域选择相互合作以生产更好的产品，获得更大的利润。因此，为了保护工业物联网设备在跨域通信中的安全性和隐私性，研究者们提出了许多跨域认证方案。然而，大多数方案暴露了工业物联网设备所属的域，或者在跨域合作中引入单点故障，从而给每个域带来不可预测的风险。本文提出了一种基于联盟区块链的更安全高效的跨域认证方案。该方案采用具有分布式追踪的群签名技术，为每个工业物联网设备提供域级匿名性，并允许公众跟踪恶意假名的真实身份。同时还考虑到IIoT设备资源有限的特点，设计了高效的跨域认证协议。安全性分析和性能评估表明，该方案可有效地应用于工业物联网跨域认证场景。

关键词：工业物联网；域级匿名；跨域认证；组签名