# Utilizing Certificateless Cryptography for IoT Device Identity Authentication Protocols in Web3

WU Zhihui[1,2], HONG Yuxuan[1], ZHOU Enyuan[3], LIU Lei[1], PEI Qingqi[1]

(1. Guangzhou Institute of Technology, Xidian University, Guangzhou 510700, China;
2. Guangzhou Lianrong Information Technology Co. Ltd., Guangzhou 510700, China;
3. The Hong Kong Polytechnic University, Hong Kong 999077, China)

DOI: 10.12142/ZTECOM.202402005

https://kns.cnki.net/kcms/detail/34.1294.TN.20240621.1417.004.html, published online June 21, 2024

Manuscript received: 2024-03-25

**Abstract:** Traditional methods of identity authentication often rely on centralized architectures, which poses risks of computational overload and single points of failure. We propose a protocol that offers a decentralized approach by distributing authentication services to edge authentication gateways and servers, facilitated by blockchain technology, thus aligning with the decentralized ethos of Web3 infrastructure. Additionally, we enhance device security against physical and cloning attacks by integrating physical unclonable functions with certificateless cryptography, bolstering the integrity of Internet of Thins (IoT) devices within the evolving landscape of the metaverse. To achieve dynamic anonymity and ensure privacy within Web3 environments, we employ fuzzy extractor technology, allowing for updates to pseudonymous identity identifiers while maintaining key consistency. The proposed protocol ensures continuous and secure identity authentication for IoT devices in practical applications, effectively addressing the pressing security concerns inherent in IoT network environments and contributing to the development of robust security infrastructure essential for the proliferation of IoT devices across diverse settings.

**Keywords:** blockchain; certificateless cryptography; identity authentication; IoT

**Citation** (Format 1): WU Z H, HONG Y X, ZHOU E Y, et al. Utilizing certificateless cryptography for IoT device identity authentication protocols in Web3 [J]. ZTE Communications, 2024, 22(2): 30 - 38. DOI: 10.12142/ZTECOM.202402005

**Citation** (Format 2): Z. H. Wu, Y. X. Hong, E. Y. Zhou, et al., "Utilizing certificateless cryptography for IoT device identity authentication protocols," *ZTE Communications*, vol. 22, no. 2, pp. 30 - 38, Jun. 2024. doi: 10.12142/ZTECOM.202402005.

## 1 Introduction

In the era of Web3 and the metaverse, the Internet of Things (IoT) plays a pivotal role in shaping the landscape of digital connectivity and immersive experiences[1-3]. As decentralized networks and blockchain-based technologies redefine the way we interact with digital platforms, the IoT acts as a fundamental enabler, bridging the physical and digital realms[4-5]. By seamlessly integrating a myriad of interconnected devices, sensors, and actuators into the digital fabric, the IoT facilitates real-time data exchange, automation, and smart decision-making within the metaverse environment. Moreover, in the context of Web3, where user sovereignty and data ownership are paramount, the IoT empowers individuals to leverage their connected devices to assert control over their digital identities and assets securely. Whether it is enhancing virtual experiences through augmented reality devices or enabling smart environments that adapt to users' preferences in the metaverse, the IoT emerges as a cornerstone technology, driving innovation and connectivity in the Web3 and metaverse era.

Among the various measures to ensure IoT security, identity authentication is crucial as it lays the foundation for the rapid and healthy development of IoT applications and is key to maintaining network security. Identity authentication in IoT applications primarily ensures the legitimacy of devices through effective verification methods, establishing trust relationships between devices and ensuring secure data

**This work was supported by the National Key Research and Development Program of China under Grant No. 2021YFB2700600, the National Natural Science Foundation of China under Grant No. 62132013, the Key Research and Development Programs of Shaanxi under Grant Nos. S2024-YF-YBGY-1540 and 2021ZDLGY06-03, the Basic Strengthening Plan Program under Grant No. 2023-JCJQ-JJ-0772, the Key-Area Research and Development Program of Guangdong Province under Grant No. 2021B0101400003, Hong Kong RGC Research Impact Fund under Grant Nos. R5060-19 and R5034-18, Areas of Excellence Scheme under Grant No. AoE/E-601/22-R, and General Research Fund under Grant Nos. 152203/20E, 152244/21E, 152169/22E and152228/23E.**

communication. Moreover, it effectively prevents malicious devices from accessing IoT systems, averting potential security incidents and ensuring the safe and reliable operation of IoT systems. Therefore, strengthening research on IoT identity authentication technology is of significant practical importance for safeguarding IoT security and promoting its healthy development.

To ensure communication security, certificateless cryptography technology has been widely applied in the design of identity authentication schemes. This cryptographic system, with its notable advantages of not requiring key management and simplifying certificate management processes, has garnered significant attention and active research from academia and industry worldwide. Furthermore, the combination of blockchain technology with the IoT is considered a crucial development trend. The distributed nature of blockchain is highly suitable for meeting the network access requirements of IoT devices in dynamic environments. Additionally, blockchain's traceability provides potential avenues for privacy protection and accountability of IoT devices. These characteristics of blockchain technology, particularly its data storage and distributed architecture, provide a technical foundation for achieving efficient, secure identity authentication, and trusted access for IoT devices.

However, existing identity authentication schemes based on certificateless cryptography and blockchain technology still have shortcomings that prevent them from meeting authentication requirements in IoT scenarios. These include significant computational and communication overheads, making them unsuitable for resource-constrained IoT scenarios, inadequate defense against common malicious attacks such as physical/cloning attacks, and insufficient support for key security features such as dynamic anonymity.

This paper addresses the security and efficiency issues facing current IoT device identity authentication. For single-device authentication scenarios, a novel trusted IoT device identity authentication protocol is proposed, combining certificateless cryptography for secure and efficient identity authentication between IoT devices and introducing blockchain technology for trustworthy data storage and traceability within the authentication system. Initially, authentication services are shifted from centralized trusted authorities to edge devices, with edge authentication gateways and servers assuming identity authentication responsibilities, thereby decentralizing the authentication architecture. Furthermore, the protocol integrates physical unclonable functions with certificateless cryptography to safeguard device secret values against malicious attacks, ensuring the integrity of device signatures. Finally, dynamic anonymity in the authentication process is achieved through fuzzy extractor technology, enabling the continuous change of users' pseudonymous identities while maintaining the consistency and security of digital signatures, thus enhancing user anonymity.

## 2 Related Work

### 2.1 Certificate Based Identity Authentication Schemes

Identity authentication schemes can be categorized into centralized Public Key Infrastructure (PKI)-based schemes and decentralized schemes based on Distributed Public Key Infrastructure (DPKI). Traditional PKI relies on centralized authorization and authentication, often leading to single points of failure, particularly in IoT environments where numerous certificate issuance requests might overwhelm central CA servers, impacting service availability. DPKI-based schemes address these shortcomings by leveraging distributed infrastructures like blockchain, granting users complete control over their digital identities while ensuring privacy protection. For instance, SAMIR et al.[6] proposed Decentralized Trustworthy-Self-Sovereign Identity Management (DT-SSIM), a framework utilizing Shamir's secret sharing scheme, blockchain, and smart contracts for identity sharing management, integrity checks, and user identity verification. Similarly, YIN et al.[7] presented a distributed IoT identity scheme integrating IoT devices as lightweight blockchain nodes and incorporating a dual-certificate model based on commitments and Bullet-Proofs for privacy protection. BAO et al.[8] proposed a blockchain-based identity management scheme for industrial IoT, ensuring identity authenticity, blindness, unlinkability, traceability, revocability, and public verifiability. Moreover, VERMA et al.[9] introduced an efficient aggregation signature scheme for industrial IoT, improving performance, especially in computational overhead and execution time, crucial for resource-constrained IoT devices. Despite DPKI's advancements in enhancing user key security and reducing private key transmission needs, certificate authentication and management remain challenges due to resource-intensive operations like revocation, storage, distribution, and authentication. Further research and optimization are required to enhance DPKI's applicability, especially in resource-constrained IoT scenarios such as smart homes and vehicular networks.

### 2.2 Identity Authentication Schemes Based on Certificateless Cryptography

AL-RIYAMI and PATERSON introduced Certificateless Public Key Cryptography (CL-PKC) to address key management issues in identity-based cryptographic systems[10]. CL-PKC eliminates the need for certificate authentication while resolving key management problems by involving a Key Generation Center (KGC) that generates partial private keys for users. DING et al. designed an anonymous identity authentication scheme based on an elliptic curve and certificateless signature technology, suitable for resource-constrained IoT devices, effectively resisting impersonation attacks[11]. WANG et al. developed a reliable and efficient certificateless signature scheme using blockchain technology and smart contracts to address potential issues such as man-in-the-middle attacks

and KGC compromise[12]. LI et al. proposed a lightweight authentication scheme for Information-Physical Energy Systems (IPES) combining elliptic curve cryptography and certificateless cryptography[13]. In vehicular ad hoc networks (VANETs), WANG et al. introduced a certificateless anonymous revocable authentication protocol for vehicle-to-vehicle communication[14]. ZHOU et al. presented a privacy-preserving identity authentication protocol based on certificateless aggregate signature schemes, while ALI et al. proposed an Enhanced Lightweight and Secure Certificateless Authentication Scheme (ELWSCAS)[15-16]. IQBAL et al. proposed a Certificateless Aggregate Signature (CLAS) scheme based on super elliptic curve cryptography[17]. These schemes balance authentication efficiency and security, offering alternatives suitable for scenarios where key management burdens are intolerable.

# 3 Methodology

## 3.1 System Design

### 3.1.1 System Model

As is shown in Fig. 1, the identity authentication system proposed in this paper mainly consists of the following three types of entities: Trusted Authority (TA), Edge Authentication Gateway (EAG), and IoT Terminal Device (TD). Each entity is described as follows:

TA, as an authoritative entity with abundant computational and storage resources, is the highest authority node in the system, responsible for generating and publishing the system's public parameters. TA also manages the identities of TD, including generating pseudonym identity markers, enabling the traceability of real identities, and functionalities such as identity revocation.

EAS, as the honest node, has stronger computing and storage capabilities than EAG and is mainly responsible for verifying the legitimacy of the aggregated signatures forwarded by EAG.

TD is generally considered as an untrusted node with relatively limited computational and storage resources. TD requires identity authentication when accessing networks or data.

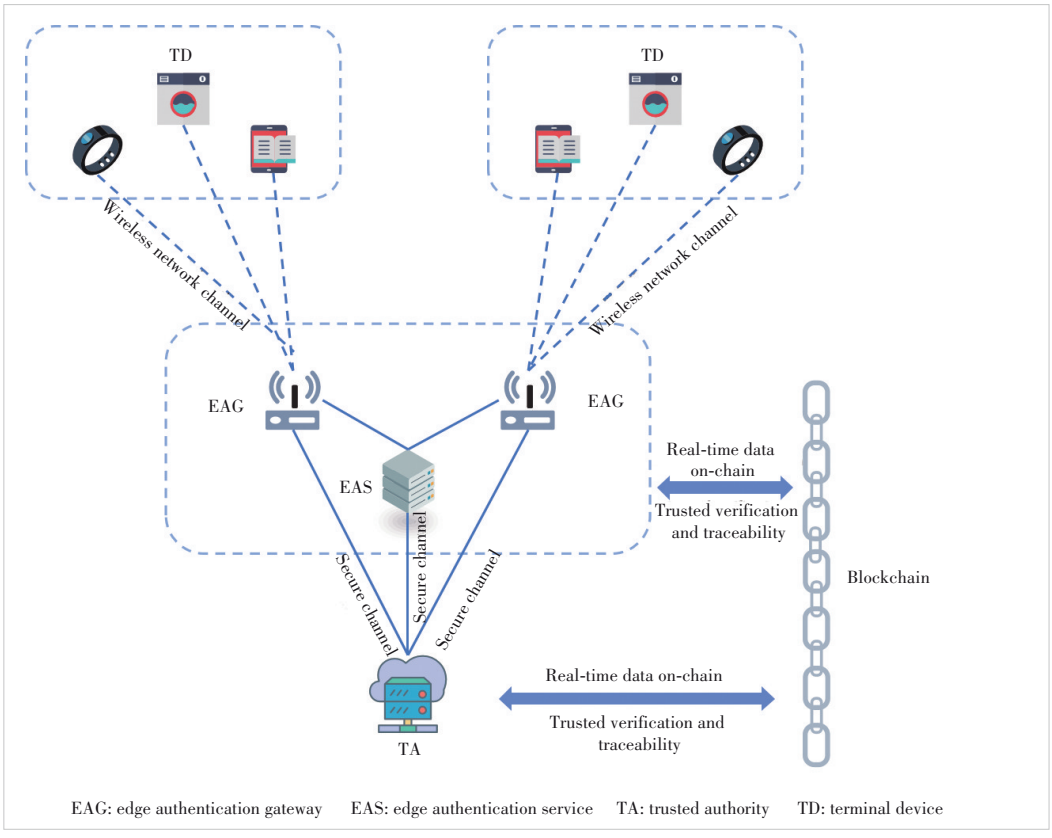
### 3.1.2 System Assumption

Based on reasonable assumptions, this paper proposes an authentication scheme for IoT devices based on a distributed architecture. The assumptions are as follows:

1) Trusted authority organizations are legitimate and absolutely trustworthy.

2) System initialization and key distribution phases are conducted in a secure communication environment, preventing malicious attackers from stealing relevant communication data during these phases. However, during the authentication phase, malicious attackers might still be able to eavesdrop, forge, or tamper with the transmitted messages.

3) All IoT devices are embedded with Physical Uncloneable Function (PUF) chips, and there is no need to employ error correction mechanisms to ensure the unclonability and tamper-proof nature of PUFs.




▲Figure 1. System architecture

## 3.2 Scheme Description

The identity authentication protocol proposed in this paper mainly includes ten parts: system initialization, secret value generation, pseudonym identity generation, partial private key generation, signature generation, signature verification, batch signature

generation, batch signature verification, pseudonym identity update, and identity revocation. The following section details the specific algorithmic implementations of these components.

#### 3.2.1 System Initialization Phase

The system initialization phase is carried out by TA, mainly used for generating public parameters and a master key for the system. When inputting the security parameters, TA randomly selects a large prime number $q$ that satisfies $q > 2^{\lambda}$ and an elliptic additive cyclic group $G$ of order $q$ over the finite field $F_q$. $P$ is the generator of the group $G$. TA chooses six secure hash functions $H_i:\{0,1\}^* \rightarrow Z_q (i = 1, 2, \cdots, 6)$ and randomly chooses $msk \in Z_q$ as the system master secret key and calculates the corresponding public key $MPK = msk \cdot P$. Then, TA chooses a fuzzy extractor $F_{EXT} = (Gen, Rep)$, where $Gen$ and $Rep$ respectively represent the key generation algorithm and the key reproduction algorithm of the fuzzy extractor. Finally, TA obtains the system public parameters $Params = (q, G, P, MPK_{TA}, H_1, H_2, H_3, H_4, H_5, H_6, F_{EXT})$ and broadcasts them within the system.

#### 3.2.2 Secret Value Generation Phase

In the secret value generation phase, TD generates a secret incentive value and its public key based on the secret incentive value and the PUF function. TD determines the secret challenge value $c_{TD}$ and generates the response value $r_{TD} = PUF(c_{TD})$ based on PUF. Then, TD computes $s_{TD} = H_1(r_{TD})$ as its secret key and calculates the corresponding public key $S_{TD} = s_{TD} \cdot P$.

Different from existing solutions, TD does not directly store the private key $s_{TD}$, but instead store a secret challenge value $c_{TD}$ determined by themselves. Even if an attacker successfully steals this secret value, they cannot generate an identical private key based on this secret value due to the unclonable and tamper-proof characteristics of PUF. Therefore, this mechanism can effectively resist physical/cloning attacks by attackers on IoT terminal devices.

#### 3.2.3 Pseudonym Identity Generation Phase

During the pseudonym identity generation phase, TD collaborates with TA to generate a pseudonym identity, which is used for subsequent anonymous communication of TD. Assuming TD's real identity marker is $RID_{TD}$, TD sends its public key $S_{TD}$ along with $RID_{TD}$ to TA to apply for the generation of a pseudonym identity. Upon receiving the pseudonym identity generation request message from TD, TA first verifies the legitimacy of the request from TD: TA checks if it exists in the malicious device list $List_{\text{malicious}}$ maintained by TA. If it is on the list, the request is denied. Otherwise, TA proceeds to calculate the pseudonym identity $PID_{TD} = H_2(S_{TD}, msk_{TA}) \oplus RID_{TD}$ and sends it to TD.

#### 3.2.4 Partial Private Key Generation Phase

The partial private key generation phase mainly completes the generation of partial public/private keys of TD and fuzzy extractor key.

1) Extract Partial Private Key

Upon receiving $(S_{TD}, PID_{TD})$ from TD, TA chooses $v_{TD} \in Z_q$ and calculates the corresponding public key $V_{TD} = v_{TD} \cdot P$. TA computes the partial private key $d_{TD} = v_{TD} + msk_{TA} \cdot h_1$, where $h_1 = H_3(MPK_{TA}, V_{TD}, PID_{TD})$.

2) Extract Fuzzy Extractor Key

TA executes the following formula to obtain the fuzzy extractor key $\delta_{TD}$ and helper string $\eta_{TD}$ of TD: $< \delta_{TD}, \eta_{TD} >= Gen(PID_{TD})$, where $Gen(\cdot)$ is fuzzy extractor's key generation algorithm. TA calculates the TD's on-chain index $idx_{TD} = H_4(\delta_{TD})$ and uses it as an input parameter to trigger the smart contract, which uploads TD's public key pair $(S_{TD}, V_{TD})$ and pseudonym identity $PID_{TD}$ to the blockchain for certification, and set a certain validity period for $PID_{TD}$. TA secretly keeps TD's original pseudonym identity $PID_{\text{origin}} = PID_{TD}$ and helper string $\eta_{TD}$. At the same time, to ensure that TD's partial private key and helper string are not leaked to other nodes in the network, TA sends them to TD through a secure channel, and similarly sends $\eta_{TD}$ to EAG within the domain through a secure channel. After receiving them, TD can verify the legitimacy of the partial private key through the following equation. Ultimately, TD completes the identity registration process in the system, obtaining the secret value pair $(c_{TD}, d_{TD}, \delta_{TD})$ and the public key pair $(S_{TD}, V_{TD})$.

#### 3.2.5 Signature Generation Phase

1) Offline Signature Generation

TD chooses $e_{TD} \in Z_q$ and computes $E_{TD} = e_{TD} \cdot P$. TD computes $\vartheta_{\text{offline}} = e + d_{TD} \cdot h_2$ and obtains offline signature $\sigma_{\text{offline}} = (E_{TD}, \vartheta_{\text{offline}})$, where $h_2 = H_5(E_{TD}, S_{TD}, V_{TD}, PID_{TD})$.

2) Online Signature Generation

After comfirming the message $M$, TD obtains the latest timestamp $T_{\text{send}}$ and recovers its secret key $s_{TM} = H_1(PUF(c_{TM}))$. TD computes $\vartheta_{TM} = \vartheta_{\text{offline}} + s_{TM} h_3$ and obtains signature $\sigma_{TM} = (E_{TM}, \vartheta_{TM})$, where $h_3 = H_6(E_{TD}, M, T_{\text{send}}, \delta_{TD})$. Finally, TD initiates an authentication request and sends $(\sigma_{TD}, M, T_{\text{send}}, PID_{TD})$ to EAG.

#### 3.2.6 Signature Verification Phase

EAG first checks the legitimacy of $T_{\text{send}}$ and $PID_{TD}$. If they are illegal, the authentication message is rejected; otherwise, EAG restores the fuzzy extractor key $\delta_{TD} = Rep(PID_{TD}, \eta_{TD})$ of TD and verifies the legitimacy of the signature through the following equation:

$$\vartheta_{TD} \cdot P = E_{TD} + h_2' \cdot (V_{TD} + h_1' \cdot MPK_{TA}) + h_3' \cdot S_{TD}, \quad (1)$$

where $h_1' = H_3(MPK_{TA}, V_{TD}, PID_{TD})$, $h_2' = H_5(E_{TD}, S_{TD}, V_{TD}, PID_{TD})$, and $h_3' = H_6(E_{TD}, M, T_{\text{send}}, \delta_{TD})$. If

the equation holds, the received $\sigma_{TD}$ is a legal signature and the TD's identity authentication is successful. Otherwise, EAG refuses to receive messages from TD, and TD identity authentication fails.

The following equation proves the correctness of Eq. (1):

$$\begin{aligned}&\vartheta_{TD}\cdot P=(e_{TD}+d_{TD}\cdot h_2'+s_{TD}\cdot h_3')\cdot P=\\&e_{TD}\cdot P+(v_{TD}+msk_{TA}\cdot h_1')\cdot h_2'\cdot P+s_{TD}\cdot h_3'\cdot P=\\&E_{TD}+h_2'\cdot(V_{TD}+h_1'\cdot MPK_{TA})+h_3'\cdot S_{TD}.\end{aligned}\tag{2}$$

### 3.2.7 Batch Signature Generation Phase

When EAG receives multiple authentication request messages from different devices in a short period of time, EAG first checks the validity of the timeliness of these messages. If the timestamp of a message has expired, the authentication message is invalid. After that, EAG calculates $\vartheta_{\text{agg}}=\sum_{i=1}^{n}\vartheta_i$ and obtains an aggregate signature $\sigma_{\text{agg}}=(\vartheta_{\text{agg}},E_1,E_2,\cdots,E_n)$. Eventually, EAG forwards $(\sigma_{\text{agg}},M_i,T_i)_{i\in 1,2,\cdots,n}$ to EAS.

### 3.2.8 Batch Signature Verification Phase

After receiving $(\sigma_{\text{agg}},M_i,T_i)_{i\in 1,2,\cdots,n}$ from EAG, EAS calculates $\delta_i=Rep(PID_i,\eta_i)$, $h_1^i=H_3(MPK_{TA},V_i,PID_i)$, $h_2^i=H_5(E_i,S_i,V_i,PID_i)$, and $h_3^i=H_6(E_i,M_i,T_i,\delta_i)$. If Eq. (3) holds, all related devices are successfully authenticated. Otherwise, EAS rejects all the authentication messages.

$$\begin{aligned}&\vartheta_{agg}\cdot P=\sum_{i=1}^{n}\vartheta_i\cdot P=\\&\sum_{i=1}^{n}(e_i+d_i\cdot h_2^i+s_i\cdot h_3^i)=\\&\sum_{i=1}^{n}E_i+h_2^i(V_i+h_1^i\cdot MPK_{TA})+h_3^i\cdot S_i.\end{aligned}\tag{3}$$

### 3.2.9 Pseudonym Identity Update Phase

TA obtains TD's original pseudonymous identity $PID_{origin}$ from the security database and executes Algorithm 1 to generate a new pseudonymous identity $PID_{TD}^{new}$. Then, TA executes the relevant smart contract and updates TD's pseudonymous identity on the chain.

**Algorithm 1:** Pseudonym Identity Update Algorithm

1. **input:** original pseudonym identity $PID_{origin}$.
2. **output:** updated pseudonym identity $PID_{TD}^{new}$.
3. **TA performs the following steps:**
4. TA converts $PID_{origin}$ to binary string *binary_pid*;
5. TA randomly selects $d$ different bits in *binary_pid* ($d$ is the maximum distance tolerated by the fuzzy extractor);
6. **For each** selected bit on *binary_pid*:
7. **if** the corresponding bit is 0 **then**
8. flip this bit to 1;
9. **else**
10. flip this bit to 0;
11. **end**
12. TA converts the updated *binary_pid* into an integer form and obtains a new pseudonym identity $PID_{TD}^{new}$.

### 3.2.10 Identity Revocation Phase

In the identity revocation phase, TA performs the following steps to revoke TD's identity:

1) When TD is detected as a malicious node, TA first obtains TD's original pseudonym identity $PID_{origin}$ and calculates TD's real identity $RID_{TD}=H_2(S_{TD},msk_{TA})\oplus PID_{origin}$.

2) TA triggers the smart contract to update the TD pseudonym identity status to "revoked".

3) TA adds TD's real identity to the malicious node blacklist $List_{malicious}$.

## 4 Evaluation

### 4.1 Setup

Regarding the identity authentication protocol designed in this paper, relevant experimental simulations are conducted in this section. All simulations in this section were performed on a personal laptop configured with an AMD Ryzen 75800H with Radeon Graphics 3.20 GHz 16.0 GB RAM, running the Windows 10 operating system. The simulations were implemented using the C programming language and simulated relevant cryptographic operations through the MIRACL cryptographic library. For the evaluation of computational and communication overheads, we selected a super singular elliptic curve defined over a finite field, where $p$ and $q$ are large prime numbers with 160 bits each. To obtain more accurate experimental results, each experiment was repeated 50 times, and the average of all test results was taken as the final experimental result. In the experiments, the proposed scheme was compared with identity authentication schemes from Refs [18–21], considering computational overheads, communication overheads, and security features for scheme comparison. The measured time cost of different operations is shown in Table 1.

### 4.2 Computation Cost Comparison

As shown in Table 2, in the authentication scheme proposed in this paper, the computational costs of signature gen-

▼**Table 1. Running time of cryptographic operations**

| Notations | Operation | Execution Time/ms |
|---|---|---|
| $T_{bp}$ | Bilinear paring operation | 3.642 5 |
| $T_{bm}$ | Scalar multiplication in bilinear pairing | 0.233 9 |
| $T_{ba}$ | Addition in bilinear pairing | 0.165 8 |
| $T_{em}$ | Scalar multiplication in ECC | 0.137 3 |
| $T_{ea}$ | Addition in ECC | 0.096 0 |
| $T_{pth}$ | Map to point hash operation | 3.813 3 |

ECC: elliptic curve cryptography

▼Table 2. Comparison of computation cost of schemes

| Scheme | Single Signature | Single Verification | Aggregate Verification |
|---|---|---|---|
| SHEN et al.[18] | $(3T_{bm} + 1T_{pth} + 1T_{bp})_{on}$ | $3T_{bp} + 2T_{pth}$ | $nT_{bp} + 1T_{bm} + (n-1)T_{ba}$ |
| KUMAR et al.[19] | $(3T_{bm} + 1T_{pth} + 1T_{bp})_{on}$ | $3T_{bp} + 2T_{pth} + 2T_{bm} + 1T_{ba}$ | $3T_{bp} + (n+1)T_{pth} + nT_{bm} + (3n-2)T_{ba}$ |
| KAMIL et al.[20] | $(3T_{em} + 1T_{ba})_{on}$ | $3T_{bp} + 3T_{em} + 1T_{ba}$ | $3T_{bp} + (2n+1)T_{em} + (2n-1)T_{ba}$ |
| ZHU et al.[21] | $(3T_{em} + 1T_{ba})_{on}$ | $4T_{em} + 3T_{ea}$ | $(5n+2)T_{em} + (7n+4)T_{ea}$ |
| Ours | $(T_{em})_{off}$ | $4T_{em} + 3T_{ea}$ | $(3n+1)T_{em} + (4n-1)T_{ea}$ |

$(\cdot)_{on}$: The signature algorithm is executed online. $(\cdot)_{off}$: The signature algorithm is executed offline.

eration and signature verification algorithms are $(T_{em})_{off} \approx 0.14$ ms and $4T_{em} + 3T_{ea} \approx 0.84$ ms, respectively. Since the authentication schemes in Refs [18], [19] and [20] require complex operations like bilinear pairings or point-to-hash operations, the computational overhead of these schemes is considerable. In comparison, the computational costs in both signature generation and verification stages of the proposed scheme are lower. Moreover, considering the utilization of online/offline signature aggregation techniques in this paper, the primary time overhead in the signature generation stage occurs during the offline signing phase. In the online stage, TD only needs to perform very minimal modular multiplication and addition operations, enabling a more efficient execution of the signature algorithm compared to the scheme in Ref. [21]. Consequently, this scheme can better meet the real-time requirements of current IoT scenarios.

Regarding the computational costs of signature verification, the time required for the authentication schemes in Refs. [18 – 21], and the proposed scheme in this paper are 18.55 ms, 19.19 ms, 11.50 ms, 0.84 ms, and 0.84 ms respectively. By employing a reduced number of modular multiplication operations instead of bilinear pairing computations, the computational overhead of the proposed scheme is also superior to those in Refs. [18 – 20].

Besides, the computation time cost of aggregate verification in each scheme increases linearly with the number of aggregated signatures. Specifically, the scheme in Ref. [19] requires the highest computation cost while the proposed scheme has the best computational efficiency.

### 4.3 Communication Cost Comparison

To evaluate the signature length and communication overhead of the proposed scheme, this experiment introduces the following metrics: $|G|$, $|G_1|$ and $|Z_q|$, which represent the size of group elements based on elliptic curve, bilinear pairing, and integer field, respectively. In this experiment, the specifications for each metric are defined as follows: $|G|$ is 320 bits, $|G_1|$ is 1 024 bits, and $|Z_q|$ is 160 bits.

As shown in Table 3, compared to the authentication schemes in Refs. [18 – 21], the single signature of the proposed scheme is reduced by 76.57%, 76.57%, 76.57% and 50% respectively. Although the communication overhead of the scheme in Ref. [21] is the same as that of the proposed scheme, it is difficult for this scheme to meet the security requirements of dynamic anonymity and resistance to physical/cloning attacks in IoT scenarios. Also, we can find that the length of aggregate signature in our scheme is smaller than that of schemes in Refs [18], [20], and [21]. Besides, although the aggregate signature length in Ref. [19] is less than that in the proposed scheme, the scheme in Ref. [19] has less than ideal computational efficiency and cannot provide user anonymity protection.

In conclusion, in the IoT environment, there are numerous devices that often have limited resources, such as restricted storage space and computing power. Compared to traditional authentication schemes, the proposed scheme does not require the storage and management of a large number of certificates and has low computational and communication overhead. This can significantly reduce the demand for storage resources, simplify the hardware requirements of devices, and thereby lower the overall system costs.

### 4.4 Security Analysis

No adversary can forge any user's identity authentication message, even if he/she has access to the public message. Also, the privacy information of non-target users can be obtained. Hence, the unforgeability ensures that the validity of the authentication message represents the identity legality of the message sender.

In the security model of certificateless public key cryptography, there are mainly two types of attackers:

1) Type I attacker $A_1$: This type of attacker is an external attacker that can obtain the user's private key, but does not have the ability to obtain the system's master key;

2) Type II attacker $A_2$: This type of attacker simulates an honest but passive KGC, capable of obtaining the system's master key, but without the ability to replace any user's pub-

▼Table 3. Comparison of communication cost of schemes

| Scheme | Single Signature Length/bits | Aggregate Signature Length/bits |
|---|---|---|
| SHEN et al.[18] | $2\|G_1\| = 2\,048$ | $(n+1)\|G_1\| = (n+1)2\,048$ |
| KUMAR et al.[19] | $2\|G_1\| = 2\,048$ | $(n+1)\|G_1\| = (n+1)2\,048$ |
| KAMIL et al.[20] | $2\|G_1\| = 2\,048$ | $2\|G_1\| = 2\,048$ |
| ZHU et al.[21] | $2\|G\| + 2\|Z_q\| = 960$ | $2n\|G\| + 2\|Z_q\| = 640n + 320$ |
| Ours | $1\|G\| + 1\|Z_q\| = 480$ | $n\|G\| + 1\|Z_q\| = 320n + 160$ |

lic key.

The security (unforgeability) of the certificateless signature algorithm is mainly proved by the games between the attacker $A \in \{A_1, A_2\}$ and the challenger $C \in \{C_1, C_2\}$.

Lemma 1: If there exists an adversary $A_1$ who can forge a valid signature with a non-negligible advantage $\varepsilon_1$, we can build a challenger $C_1$ who can solve the hardness of ECDL problem with an obvious advantage $\varepsilon_1' \geqslant \left(1 - \frac{1}{e}\right)\left(\frac{\varepsilon_1}{e(q_1 + q_2 + 1)q_{H_2}}\right)$, where $q_1$, $q_2$ and $q_{H_2}$ denote the number of Partial-Private-Key-Extract query, Private-Key-Extract query, and $H_2$ oracle query, respectively.

Proof: In the beginning, the challenger $C_1$ takes an instance $(P, aP)$ of the ECDL problem as input, and its purpose is to compute $a$.

**Setup:** In this stage, $C_1$ executes the system initialization algorithm and obtains the public parameters of the system $Params = (q, G, P, MPK, H_1, H_2, H_3, F_{EXT})$. $H_i (i = 1, \cdots, 6)$ are six random prediction machines. $C_1$ then picks $PID^*$ as the target user.

**Query:** In this stage, the challenger $C_1$ can query the following oracles adaptively and polynomially.

• *Create-User Query*: When $C_1$ receives *Create-User Query* with $PID_i$ as input, it first checks whether there is a tuple $(PID_i, S_i, V_i, s_i, d_i, \delta_i, flag)$ in the list $L_{user}$. If so, $C_1$ sends $PK_i = (S_i, V_i)$ directly to $A_1$. If no, $C_1$ performs the following operations: 1) If $PID_i \neq PID^*$, $C_1$ randomly selects $s_i$, $d_i$, $h_1^i \in Z_q$, calculates $V_i = d_i P - h_1^i MPK$, $S_i = s_i P$, and $< \delta_i, \eta_i > = Gen(PID_i)$, and sets $flag = False$=; 2) If $PID_i = PID^*$, $C_1$ randomly selects $v_i$, $s_i$, $h_1^i \in Z_q$ and calculates $V_i = v_i \cdot P$ and $S_i = s_i \cdot P$, letting $d_i = \perp$ and $flag = False$. Finally, $C_1$ returns $PK_i = (S_i, V_i)$ to $A_1$ and adds $(PID_i, S_i, V_i, s_i, d_i, \delta_i, flag)$ and $(MPK, V_i, PID_i, h_1^i)$ to the lists $L_{user}$ and $L_1$, respectively.

• *$H_1$ Query*: When $C_1$ receives an *$H_1$ Query* from $A_1$ with input $(MPK, V_i, PID_i)$, it first checks for the existence of tuples $(MPK, V_i, PID_i, h_1^i)$ in list $L_1$. If so, $C_1$ will directly send $h_1^i$ to $A_1$. If not, $C_1$ will submit the corresponding *Create-User* query with $PID_i$ as input, then find $h_1^i$ from list $L_1$, and send it to $A_1$.

• *$H_2$ Query*: When $C_1$ receives an *$H_2$ Query* from $A_1$ with input $(E_i, S_i, V_i, PID_i)$, it first checks whether a tuple $(E_i, S_i, V_i, PID_i, h_2^i)$ exists in list $L_2$. If so, $C_1$ sends $h_2^i$ directly to $A_1$. If it does not exist, $C_1$ randomly selects $h_2^i \in Z_q$ and sends it to $A_1$, adding $(E_i, S_i, V_i, PID_i, h_2^i)$ to list $L_2$.

• *$H_3$ Query*: When $C_1$ receives an *$H_3$ Query* from $A_1$ with input $(E_i, M_i, t_i, \delta_i)$, it first checks whether a tuple $(E_i, M_i, t_i, \delta_i, h_3^i)$ exists in list $L_3$. If so, $C_1$ sends $h_3^i$ directly to $A_1$. If it does not exist, $C_1$ randomly selects $h_3^i \in Z_q$ and sends it to $A_1$, adding $(E_i, M_i, t_i, \delta_i, h_3^i)$ to list $L_3$.

• *Partial-Private-Key-Extract*: When $C_1$ receives *Partial-Private-Key-Extract Query* with input $PID_i$ from $A_1$, it determines whether $PID_i$ and $PID^*$ are equal. If they are equal, the $C_1$ query is terminated. Otherwise, $C_1$ checks whether a tuples $(PID_i, S_i, V_i, s_i, d_i, \delta_i, flag)$ exists in list $L_{user}$. If so, $C_1$ directly sends $(d_i, \delta_i, V_i)$ obtained by list $L_{user}$ to $A_1$. If not, $C_1$ submits $PID_i$ as input to the corresponding *Create-User Query*, then finds $(d_i, \delta_i, V_i)$ from the list $L_{user}$, and sends it to $A_1$

• *Secret-Value-Extract*: When $C_1$ receives a *Secret-Value-Extract Query* from $A_1$ with $PID_i$ as input, $C_1$ checks whether a tuple $(PID_i, S_i, V_i, s_i, d_i, \delta_i, flag)$ exists in list $L_{user}$. If so, $C_1$ sends $(s_i, S_i)$ obtained by list $L_{user}$ to $A_1$. Otherwise, $C_1$ submits $PID_i$ as input to the corresponding *Create-User Query*, and then finds $(s_i, S_i)$ from the list $L_{user}$ and sends it to $A_1$.

• *Private-Key-Extract Query*: When $C_1$ receives a *Private-Key-Extract Query* with $PID_i$ as input from $A_1$, it determines whether $PID_i$ and $PID^*$ are equal. If equal, the inquiry is terminated; otherwise, $C_1$ checks whether a tuple $(PID_i, S_i, V_i, s_i, d_i, \delta_i, flag)$ exists in the list $L_{user}$. If so, $C_1$ directly sends $SK_i = (s_i, d_i, \delta_i)$ obtained by list $L_{user}$ to $A_1$. Otherwise, $C_1$ submits $PID_i$ as input to the corresponding *Create-User Query*, and then finds $SK_i = (s_i, d_i, \delta_i)$ from list $L_{user}$, and sends it to $A_1$.

• *Replace-Public-Key Query*: When $C_1$ receives $A_1$ *Replace-Public-Key Query* with $(PID_i, S_i', V_i')$ as input from $A_1$, it determines whether $PID_i$ and $PID^*$ are equal. If equal, the inquiry is terminated; otherwise, $C_1$ gets $(PID_i, S_i, V_i, s_i, d_i, \delta_i, flag)$ from list $L_{user}$ and replaces $(S_i, V_i)$ in the list with $(S_i', V_i')$.

• *Sign Query*: When $C_1$ receives the *Sign Query* with $(PID_i, M_i)$ as input from $A_1$, it determines whether $PID_i$ and $PID^*$ are equal. If $PID_i \neq PID^*$ and $s_i \neq \perp$, $C_1$ selects $e_i$, $h_2^i$ and $h_3^i \in Z_q$ at random and calculates $E_i = e_i \cdot P$ and $\vartheta_i = e_i + d_i \cdot h_2^i + s_i \cdot h_3^i$, where $h_2^i = H_2(MPK, V_i, PID_i)$ and $h_3^i = H_3(E_i, S_i, V_i, PID_i)$. $C_1$ then sends the signature $\sigma_i = (\vartheta_i, E_i)$ to $A_1$ and adds $(E_i, S_i, V_i, PID_i, h_2^i)$ and $(E_i, M_i, t_i, \delta_i, h_3^i)$ to the lists $L_2$ and $L_3$, respectively. If $PID_i = PID^*$, $C_1$ gets $(PID_i, S_i, V_i, s_i, d_i, \delta_i, flag)$ and $(MPK, V_i, PID_i, h_1^i)$ from the lists $L_{user}$ and $L_1$, where: $s_i = \perp$ and $d_i = \perp$. $\vartheta_i$, $h_2^i$ and $h_3^i \in Z_q$ are randomly selected and $E_i = \vartheta_i \cdot P - h_2^i(V_{TM} + h_1^i \cdot MPK) - h_3^i \cdot S_i$ is calculated. Finally, $C_1$ sends the signature $\sigma_i = (\vartheta_i, E_i)$ to $A_1$ and adds $(E_i, S_i, V_i, PID_i, h_2^i)$ and $(E_i, M_i, t_i, \delta_i, h_3^i)$ to the list $L_2$ and $L_3$, respectively.

**Forgery:** When $C_1$ receives forged signature $\sigma^* = (E^*, \vartheta^*)$ from $A_1$ about $(PID^*, M^*)$, it determines whether $PID_i$ and $PID^*$ are equal. If not, $C_1$ terminates the game. Otherwise, $C_1$ replays $A_1$ and gets a new forged signature $\sigma^{*(2)} = (E^*, \vartheta^*)$ about $(PID^*, M^*)$. From this, $C_1$ can obtain:

$$\begin{cases} \vartheta^* = e^* + h_2^*(v^* + a \cdot h_1^*) + s^* \cdot h_3^* \\ \vartheta^{*(2)} = e^* + h_2^{*(2)}(v^* + a \cdot h_1^*) + s^* \cdot h_3^*. \end{cases} \tag{4}$$

In turn, $C_1$ outputs $a = \frac{1}{h_1^*}\left(\frac{\vartheta^* - \vartheta^{*(2)}}{h_2^* - h_2^{*(2)}} - v^*\right)$ as the solution to the elliptic curve discrete logarithm problem.

Next, we define the following events:

Event $E_1$: $E_1$ indicates that $C_1$ does not terminate the game

during the query phase.

Event $E_2$: $E_2$ indicates that $C_1$ does not terminate the game during the Forgery phase.

Event $E_3$: $E_3$ indicates that the forged signature $\sigma^*$ and $\sigma^{*(2)}$ about $(PID^*, M^*)$ are valid signatures

Based on game analysis, we can calculate:

The probability of $E_1$ is $Pr(E_1) \geqslant \left(1 - \frac{1}{q_1 + q_2 + 1}\right)^{q_1 + q_2}$.

The probability of $E_2$ is $Pr(E_2) = \frac{1}{q_1 + q_2 + 1}$.

According to the Forking lemma, if one legitimate signature can be output by an advantage $\varepsilon_1$, the probability of two legitimate signatures being output is $Pr(E_3) \geqslant (1 - \frac{1}{e}) \frac{\varepsilon_1}{q_{H_2}}$.

From this, we can get the advantages of solving ECDLP problems:

$$\varepsilon_1' = Pr(E_1 \wedge E_2 \wedge E_3) \geqslant \left(1 - \frac{1}{e}\right)\left(\frac{\varepsilon_1}{e(q_1 + q_2 + 1)q_{H_2}}\right). \quad (5)$$

If malicious attacker $A_1$ can successfully forge two signatures with the probability of $\varepsilon_1'$, it can be inferred that $C_1$ has the ability to solve the elliptic curve discrete logarithm problem. However, the existence of this capability is in apparent contradiction with the fact that the elliptic curve discrete logarithm problem is considered to be difficult to solve. Therefore, it can be inferred that the probability of $A_1$ forging a signature successfully is negligible. Therefore, we can conclude that this scheme can effectively defend against the threat of Class I attackers. Proof completes.

## 5 Conclusions

This paper proposes a blockchain-based identity authentication protocol for IoT devices in Web3, addressing security vulnerabilities in current centralized authentication methods. Based on blockchain and certificateless cryptography, the authentication process between IoT terminal devices and authentication nodes is designed, incorporating technologies such as physical unclonable functions and fuzzy extractors into the authentication protocol to achieve security features lacking in current identity authentication protocols, such as resistance to physical/cloning attacks and dynamic anonymity. Simulation results demonstrate that compared to existing solutions, the proposed identity authentication protocol has the advantages of low computational/communication overhead. Additionally, the security analysis of the protocol shows its excellent performance against various malicious attacks.

**References**

[1] LIU Z T, XIANG Y X, SHI J, et al. Make Web3.0 connected [J]. IEEE transactions on dependable and secure computing, 2022, 19(5): 2965 – 2981. DOI: 10.1109/TDSC.2021.3079315
[2] RAY P P. Web3: A comprehensive review on background, technologies, applications, zero-trust architectures, challenges and future directions [J]. Internet of Things and cyber-physical systems, 2023, 3: 213 – 248. DOI: 10.1016/j.iotcps.2023.05.003
[3] CHEN C, ZHANG L, LI Y H, et al. When digital economy meets Web3.0: applications and challenges [J]. IEEE open journal of the computer society, 2022, 3: 233 – 245. DOI: 10.1109/OJCS.2022.3217565
[4] GUPTA M. Integration of IoT and Blockchain for user Authentication [J]. Scientific journal of metaverse and blockchain technologies, 2023, 1(1): 72 – 84. DOI: 10.36676/sjmbt.v1i1.10
[5] GAO H M, DUAN P F, PAN X F, et al. Blockchain-enabled supervised secure data sharing and delegation scheme in Web3.0 [J]. Journal of cloud computing, 2024, 13(1): 21. DOI: 10.1186/s13677-023-00575-8
[6] SAMIR E, WU H Y, AZAB M, et al. DT-SSIM: A decentralized trustworthy self-sovereign identity management framework [J]. IEEE Internet of Things journal, 2022, 9(11): 7972 – 7988. DOI: 10.1109/JIOT.2021.3112537
[7] YIN J, XIAO Y, PEI Q Q, et al. SmartDID: A novel privacy-preserving identity based on blockchain for IoT [J]. IEEE Internet of Things journal, 2023, 10(8): 6718 – 6732. DOI: 10.1109/JIOT.2022.3145089
[8] BAO Z J, HE D B, KHAN M K, et al. PBidm: privacy-preserving blockchain-based identity management system for industrial Internet of Things [J]. IEEE transactions on industrial informatics, 2023, 19(2): 1524 – 1534. DOI: 10.1109/TII.2022.3206798
[9] VERMA G K, KUMAR N, GOPE P, et al. SCBS: a short certificate-based signature scheme with efficient aggregation for industrial-internet-of-things environment [J]. IEEE Internet of Things journal, 2021, 8(11): 9305 – 9316. DOI: 10.1109/JIOT.2021.3055843
[10] AL-RIYAMI S S, PATERSON K G. Certificateless public key cryptography [C]//Advances in Cryptology: ASIACRYPT 2003. Berlin, Heidelberg: Springer, 2003, 2894: 452 – 473. DOI: 10.1007/978-3-540-40061-5_29
[11] DING X Y, WANG X X, XIE Y, et al. A lightweight anonymous authentication protocol for resource-constrained devices in Internet of Things [J]. IEEE Internet of Things journal, 2022, 9(3): 1818 – 1829. DOI: 10.1109/JIOT.2021.3088641
[12] WANG W Z, XU H, ALAZAB M, et al. Blockchain-based reliable and efficient certificateless signature for IIoT devices [J]. IEEE transactions on industrial informatics, 2022, 18(10): 7059 – 7067. DOI: 10.1109/TII.2021.3084753
[13] LI X, JIANG C, DU D J, et al. A novel revocable lightweight authentication scheme for resource-constrained devices in cyber – physical power systems [J]. IEEE Internet of Things journal, 2023, 10(6): 5280 – 5292. DOI: 10.1109/JIOT.2022.3221943
[14] WANG Z L, ZHOU Y W, QIAO Z R, et al. An anonymous and revocable authentication protocol for vehicle-to-vehicle communications [J]. IEEE Internet of Things journal, 2023, 10(6): 5114 – 5127. DOI: 10.1109/JIOT.2022.3222469
[15] ZHOU Y W, CAO L, QIAO Z R, et al. An efficient identity authentication scheme with dynamic anonymity for VANETs [J]. IEEE Internet of Things journal, 2023, 10(11): 10052 – 10065. DOI: 10.1109/JIOT.2023.3236699
[16] ALI U, BIN IDRIS M Y I, FRNDA J, et al. Enhanced lightweight and secure certificateless authentication scheme (ELWSCAS) for Internet of Things environment [J]. Internet of Things, 2023, 24: 100923. DOI: 10.1016/j.iot.2023.100923
[17] IQBAL A, ZUBAIR M, KHAN M A, et al. An efficient and secure certificateless aggregate signature scheme for vehicular ad hoc networks [J]. Future Internet, 2023, 15(8): 266. DOI: 10.3390/fi15080266
[18] SHEN L M, MA J F, LIU X M, et al. A secure and efficient ID-based aggregate signature scheme for wireless sensor networks [J]. IEEE Internet of Things journal, 2017, 4(2): 546 – 554. DOI: 10.1109/JIOT.2016.2557487
[19] KUMAR P, KUMARI S, SHARMA V, et al. A certificateless aggregate

signature scheme for healthcare wireless sensor network [J]. Sustainable computing: Informatics and systems, 2018, 18: 80 – 89. DOI: 10.1016/j.suscom.2017.09.002
[20] KAMIL I A, OGUNDOYIN S O. On the security of privacy-preserving authentication scheme with full aggregation in vehicular ad hoc network [J]. Security and privacy, 2020, 3(1): e104. DOI: 10.1002/spy2.104
[21] ZHU F, YI X, ABUADBBA A, et al. Certificate-based anonymous authentication with efficient aggregation for wireless medical sensor networks [J]. IEEE Internet of Things journal, 2022, 9(14): 12209 – 12218. DOI: 10.1109/JIOT.2021.3134693

## Biographies

**WU Zhihui** received his master's degree from Xidian University, China. He is the Deputy General Manager of Guangzhou Lianrong Information Technology Co. He has been responsible for project management and technical development in the fields of data security, privacy computing and blockchain technology for many years. He has led or participated in more than ten research projects, and published two papers and 11 invention patents. He received 2023 Blockchain Innovator of the Year Award.

**HONG Yuxuan** received his BE degree in information security from Xidian University, China in 2021. He is currently pursuing his ME degree with College of Guangzhou Institute of Technology, Xidian University. His research interests include identity authentication and blockchain.

**ZHOU Enyuan** is currently pursuing his PhD degree in Department of Computing in The Hong Kong Polytechnic University, China. He received his BE degree in information security from Northeastern University, China and MSc degree in cyberspace security (supervised by Prof. PEI Qingqi) from Xidian University, China. His current research interests include Blockchain, Database, and knowledge graph. He has published several papers in prestigious journals and conferences in data management field such as VLDB and IEEE TKDE.

**LIU Lei** received his BEng degree in electronic information engineering from Zhengzhou University, China in 2010, and his MSc and PhD degrees in communication and information systems from Xidian University, China in 2013 and 2019, respectively. He is currently an associate professor with the Guangzhou Institute of Technology, Xidian University. His research interests include vehicular ad hoc networks, edge intelligence and distributed computing.

**PEI Qingqi** (qqpei@mail.xidian.edu.cn) is a full professor and PhD supervisor of Xidian University, China. He serves as the director of the Blockchain Application and Evaluation Research Center of Xidian University and the executive director of the Shaanxi Key Laboratory of Blockchain and Secure Computing. His research interests focus on cognitive networks, data security, and blockchain. He has led or participated in more than 30 national, provincial and ministerial projects. He has published more than 100 journal or conference papers and obtained more than 60 patents (including five international PCT patents) and 40 registered software copyrights. He was awarded one second prize of national technology invention awards and three first prizes of provincial or ministerial scientific and technological awards.