Device-Side Lightweight Mutual Authentication and Key Agreement Scheme Based on Chameleon Hashing for Industrial Internet of Things

Qingyang Zhang, *Member, IEEE*, Xiaolong Zhou, Hong Zhong, *Member, IEEE*, Jie Cui, *Senior Member, IEEE*, Jiaxin Li, and Debiao He, *Member, IEEE*

***Abstract*— Several authentication and key agreement (AKA) schemes have been proposed to ensure secure communication in the Industrial Internet of Things (IIoT). However, most of these schemes face two primary problems. First, they cannot resist various attacks, such as impersonation and device capture attacks. Second, these schemes overlook the resource-constrained IIoT devices, failing to guarantee lightweight overhead for device operations. Therefore, we propose a novel and efficient AKA scheme. Utilizing the chameleon hash function and physical unclonable function, the proposed scheme implements a lightweight overhead for both authentication parties while maintaining the overhead of the gateway within a reasonable range. Furthermore, we implement device anonymity based on lightweight operations such as hash and XOR. In addition, we perform a rigorous security analysis using the widely accepted Real-Or-Random model, BAN logic, and Proverif tool. Finally, through heuristic analysis and experiments, we substantiate that our scheme surpasses the compared schemes in terms of both security attributes and system overhead.**

***Index Terms*— Industrial Internet of Things (IIoT), mutual authentication, key agreement, physical unclonable function, chameleon hash function.**

I. Introduction

THE Industrial Internet of Things (IIoT) refers to the deep integration of IoT technology with advanced manufacturing techniques to improve productivity. Industries such as automotive manufacturing, energy, and agriculture can leverage IIoT for real-time monitoring, data management, production optimization, and reducing operational costs. According to a relevant report [1], the global market size of the IIoT is estimated to surpass 300 billion by 2021 and is projected to exceed 1.7 trillion by 2030. However, the pervasive use of IIoT devices has introduced security challenges, particularly in the context of data exchange. For instance, smart meters transmit data to power management personnel through public channels and handle vast amounts of sensitive data [2]. Owing to the openness of networks in the industry, data exchanged during communication is susceptible to eavesdropping by adversaries. If an adversary obtains data from a smart meter, they may cause unpredictable losses to the factory.

Authentication and key agreement (AKA) schemes have been widely employed to address the issues of secure communication between untrusted entities. However, most of these still face two primary problems: resource heterogeneity and various types of attacks. IIoT devices are often resource-constrained owing to size constraints, whereas gateway resources are relatively abundant. Current public-key cryptography schemes impose a heavy burden on resource-constrained IIoT devices. The AKA scheme based on symmetric cryptography and hash operations is efficient but insufficiently flexible [3]. For instance, in schemes [4], [5], the device side must perform complex elliptic curve point multiplication operations, resulting in a high cost on the device side and being unsuitable for deployment in IIoT devices with limited resources.

Meanwhile, the current increasingly complex network situation of the IIoT reveals several security challenges that render existing AKA schemes inadequate. These challenges include the susceptibility of IIoT systems to various attacks, such as replay and device capture attacks, etc. Among them, to cope with device capture attacks, some schemes, such as [6] and [7], introduce fingerprint-based multi-factor authentication, using externally inputted fingerprints or identities to encrypt device local keys. The adversaries obtain the device and extract all local data, but cannot obtain the device key owing to the lack of user fingerprints and identity information. However, several devices in the IIoT do not support fingerprints or other external inputs because of size limitations or special work scenarios.

Manuscript received 25 February 2024; revised 21 June 2024, 21 July 2024, and 2 August 2024; accepted 11 August 2024. Date of publication 28 August 2024; date of current version 6 September 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62272002, Grant 62202005, Grant 62372002, and Grant 62325209; in part by the Fundamental Research Funds for Central Universities under Grant 2042023KF0203; in part by the Natural Science Foundation of Anhui Province, China, under Grant 2208085QF198; and in part by the University Synergy Innovation Program of Anhui Province under Grant GXXT-2022-049. The associate editor coordinating the review of this article and approving it for publication was Dr. Ming Li. *(Corresponding author: Hong Zhong.)*

Qingyang Zhang, Xiaolong Zhou, Hong Zhong, and Jie Cui are with the Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, School of Computer Science and Technology, and Anhui Engineering Laboratory of IoT Security Technologies, Anhui University, Hefei 230039, China (e-mail: zhongh@ahu.edu.cn).

Jiaxin Li is with the Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, School of Computer Science and Technology, Anhui University, Hefei, Anhui 230039, China, and also with the Security Research Institute, New H3C Group, Hefei 230088, China (e-mail: li.jiaxin@h3c.com).

Debiao He is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China, and also with Shanghai Key Laboratory of Privacy Preserving Computation, MatrixElements Technologies, Shanghai 201204, China (e-mail: hedebiao@163.com).

Digital Object Identifier 10.1109/TIFS.2024.3451357

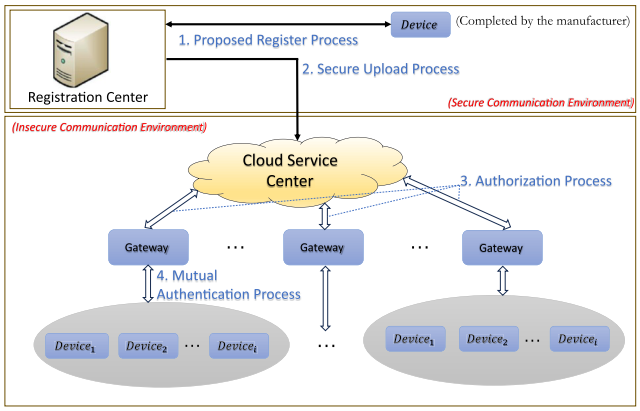


Fig. 1. System model.

Recently, some new techniques with special security features or performances that can be used to construct AKA schemes have been proposed. For instance, owing to its security features, the physical unclonable function (PUF) [8] is gradually being recognized and used to construct schemes. External access to the PUF through micro detection or other invasive techniques in the device can cause damage to the PUF [9]. Therefore, it is possible to protect the information stored within it, such as authentication information. Although the PUF provides good security, the direct delivery of responses in a public channel is also a potential threat to the IIoT. Meanwhile, considering the heterogeneous scenario analyzed above, the cost of the authentication process on the device side should be low, and the cost on the gateway side must be maintained within a reasonable range. The chameleon hash function [10] generates verification information with high efficiency, and the number of verification operations is not particularly high, making it suitable for this heterogeneous scenario.

Accordingly, by utilizing the PUF and the chameleon hash function, we aim to construct a novel and device-side lightweight AKA scheme for IIoT to protect devices from attacks, such as device capture attacks.

A. System Architecture

The system architecture of the proposed scheme is illustrated in Fig. 1. The model involves five entities: the Registration Center (RC), Cloud Service Center (CSC), gateway (GWN), and smart devices (SD_A and SD_B).

- Registration Center (RC): The RC has sufficient computing and communication resources and is responsible for initializing system parameters. When a device leaves the factory, it must be registered and reported uniformly at the RC. RC can delegate some of its permissions to the CSC, which is responsible for the initialization process of factory GWN parameters. The RC is completely trustworthy and is not compromised by adversaries.
- Cloud Service Center (CSC): In our model, a CSC can be responsible for a relatively large area, such as an industrial park. The CSC serves as a bridge connecting the RC and the GWN, replacing the RC to initialize the relevant data for the GWN.
- Gateway (GWN): The gateway is a trusted entity in the factory, responsible for authenticating all IIoT devices in a specific area. Before participating in authentication, the gateway must request the CSC to obtain the relevant parameters for the devices.
- Smart Devices (SD): In our model, each IIoT device is equipped with a PUF, and can perform storage and related cryptographic calculations. Because devices in a factory are considered untrusted, authentication is required prior to communication. Considering the following authentication scenario, the SD_A initiates an authentication request to the SD_B.

B. Research Contributions

- Focusing on the limited resources on the device side in the IIoT environment, an efficient anonymous AKA for the IIoT is proposed. Based on the chameleon hash function, the proposed scheme is lightweight for resource-constrained IIoT devices. The experimental simulation results demonstrate that our scheme has a lower resource overhead and is suitable for deployment in resource-constrained IIoT device environments. Specifically, compared to the simulation time of other schemes, our scheme achieved a performance improvement of approximately 9.48% to 73.70%, reducing the overhead significantly. On average, the device side overhead decreased by approximately 90.38% compared to other schemes.
- Considering the common and challenging security issues in the IIoT, we combine the PUF and chameleon hash functions to enhance the robustness of the scheme. The proposed scheme can resist device capture attacks and provide better resilience against PUF modeling attacks. We combine the identity information with the chameleon hash function, and only devices with trapdoor keys can construct random numbers to generate identity information and pass authentication. Even if identity information is leaked, it will not affect the implementation process of the authentication.
- We prove the security of our scheme based on the Real-Or-Random (ROR) model, as well as the proverif tool, BAN Logic. In addition, a heuristic security analysis is performed for the proposed scheme. It is demonstrated that our scheme, compared to other schemes, can resist privileged-insider attacks, device capture attacks, replay attacks, etc.

C. Paper Outline

The remainder of this paper is organized as follows: Section II discusses the latest studies on IIoT in recent years. Section III introduces the preliminary knowledge. Section IV describes the adversary model and design objectives. Section V provides a detailed description of the authentication process for the proposed scheme. Section VI presents the security analysis and proof of the proposed scheme. Section VII compares the performance overhead of the proposed scheme. Finally, Section VIII concludes this paper.

II. Related Work

In 2013, Guo et al. [11] designed a new authentication scheme based on the chameleon hash function. The chameleon hash function is usually constructed through ECC, which allows the entity with the trapdoor to compute a collision. When the verifier is given a predefined hash value, it believes that only entities with knowledge of trapdoors can compute random numbers that yield the same hash value. This scheme can generate a new temporary public key for each authentication, avoiding the use of fixed public keys. However, this scheme involves the issuance and verification of certificates, and the generation of temporary public keys also increases system overhead. This renders it unsuitable to deploy in resource-constrained IIoT devices.

In 2014, Turkanoví et al. [12] proposed a lightweight two-factor authentication scheme, which is suitable for the IoT due to its low computational and communication costs. However, the system is susceptible to privileged insider attacks and physical device capture attacks, among others. Porambage et al. [13] designed a system that allows users to authenticate directly with IoT devices. This means that the user and devices can achieve end-to-end authentication, resulting in significantly reduced communication overhead and gateway-side expenses. Both of these schemes are efficient but vulnerable to the aforementioned known attacks.

In 2015, Chang and Le [5] proposed two efficient two-factor user authentication schemes for wireless sensor networks, $P1$ and $P2$. $P1$ is a lightweight security authentication protocol. However, it has several security vulnerabilities, making it highly susceptible to attacks by adversaries. To address these issues, they also designed $P2$, which is primarily based on ECC and hash functions. $P2$ offers more security features than $P1$, such as adding perfect forward secrecy. Nevertheless, $P2$ still cannot defend against replay attacks, or physical device capture attacks, and does not ensure user anonymity.

In 2016, Che et al. [14] proposed an innovative privacy-preserving mutual authentication protocol based on PUF. This scheme provides complete end-to-end privacy-preserving and excellent security properties. This scheme retains path timing information rather than storing response bitstrings on the verifier. The authentication process of this scheme is divided into token recognition, verifier authentication, and token authentication stages, each of which involves complex calculations and comparison operations. Therefore, it is not suitable to deploy in resource-constrained devices as it has significant hardware overhead.

In 2018, Gope et al. [15] proposed a scheme for privacy-preserving authentication using PUF in a Radio frequency identification (RFID) system. Furthermore, they improved this scheme to make it suitable for noisy environments. The system has multiple responses and challenges as backup, effectively resisting denial of service (DoS) attacks and synchronization attacks. Subsequently, Gope and Sikdar [16] introduced a privacy-aware authentication and key agreement protocol designed for the smart grid environment. Their scheme ensures secure communication among system entities. It is noteworthy for its efficiency, primarily depending on PUF and hash functions. However, it has some security issues, such as being unable to resist internal privilege attacks.

In 2019, Zhang et al. [17] designed a seamless handover authentication scheme for 5G heterogeneous networks based on blockchain and chameleon hash functions. In this scheme, users and verifiers retrieve each other's chameleon hash results from the blockchain. Similar to [11], this scheme generates temporary chameleon hash function public keys for each authentication, ensuring that exposing random numbers of the chameleon hash function and temporary public keys does not pose a threat to system security. However, blockchain significantly increases system overhead [18] and is not suitable for deployment in IIoT devices.

In 2022, Wu et al. [19] implemented a Verifiable Threshold Predicate Encryption (VTPE) system based on Threshold Predicate Encryption (TPE) technology. Building upon the VTPE scheme, they propose a novel fingerprint-based authentication system that satisfies various security properties. However, Zhang et al. [20] pointed out security vulnerabilities in [19], such as the exposure of the user's real ID and user privacy disclosure, which could allow adversaries to impersonate legitimate users and deceive the server. Furthermore, they believe that the adversary can obtain the user's query and can successfully impersonate a legitimate user to deceive the server.

In previous research, many schemes ignored the limited resources of IIoT devices, which cannot meet the requirements of device-side lightweight. The overhead on the device side will have a significant impact on the efficiency of the system. Meanwhile, some schemes cannot guarantee the security of the system. Therefore, we propose an anonymous authentication protocol to strike a balance between security concerns and system overhead. The proposed scheme ensures system security while achieving device-side lightweight.

III. Preliminaries

In this section, to enhance our comprehension of the proposed scheme, we introduce the employment of the chameleon hash function and physical unclonable function in our proposed scheme.

A. Chameleon Hash Function

Each entity generate a public-private key pair (pk, x), where $x \in \mathbb{Z}_q^*$, and $pk = x \cdot P$. Here, P represents the base point of the elliptic curve cyclic group. The public key serves as the hash key and the private key as a trapdoor. Devices generate two random numbers m_0 and r_0, as initial values, where m_0 and $r_0 \in \mathbb{Z}_q^*$. The chameleon hash function is defined as $CH_{pk} = (m_0, r_0) = m_0 \cdot P + r_0 \cdot pk$. Then trapdoor $tp = (x, k^*)$, where $k^* = m_0 + r_0 \cdot x$. When an entity knows the public key, it can generate a hash value. However, only the holder of the trapdoor hash key can obtain the conflict for each input. The chameleon hash function has the following properties:

(1) One-wayness: Given an input (m_0, r_0, pk), calculating the output CH_{pk} is easy. However, calculating m_0 and r_0 from the output value $CH_{pk} = m_0 \cdot P + r_0 \cdot pk$ is impossible.

(2) Trapdoor Collisions: With the trapdoor tp, for initial inputs m_0 and r_0, and given input r_1, one can calculate m_1, satisfying $CH_{pk}(m_1, r_1) = CH_{pk}(m_0, r_0)$, where $m_1 = k^* - r_1 \cdot x$.

(3) Collision Resistance: In the absence of the trapdoor tp, for initial inputs m_0 and r_0, finding $CH_{pk}(m_1, r_1) = CH_{pk}(m_0, r_0)$ where $(m_1, r_1) \neq (m_0, r_0)$ is challenging.

B. Physical Uncloneable Function

Physical unclonable function is a security hardware component based on physical characteristics. The main principle of PUF is based on minor random changes in each chip, which are caused by insignificant irregularities in the chip manufacturing process. Given a random challenge C, PUF generates an unpredictable response R due to the random differences in its internal physical structure. This can be expressed as $R = PUF(C)$, providing each chip with unique features that can be utilized in security applications, such as identity verification and encryption key generation. In general, PUF possesses the following characteristics:

(1) Unpredictability: For a probabilistic polynomial-time adversary $\mathcal{A}$, who is allowed to access certain challenge-response pairs of the PUF. For a new challenge C, the response $R = PUF(C)$ is sent to the adversary $\mathcal{A}$ together with a random number R'. The adversary's advantage in correctly distinguishing which one is the PUF response R does not exceed $\frac{1}{2}$.

(2) Uniqueness: Due to the minor environmental differences during the PUF manufacturing process, PUFs produced in the same process are still distinct. Specifically, when given the same challenge C, different PUFs generated from the same manufacturing process produce responses that are unpredictable and unique.

(3) Stability: This refers to the ability of PUF to maintain stable and reliable output under various environmental conditions, such as temperature, humidity, and other influences.

(4) Tamper resistance: The response generation mechanism of PUF cannot be replicated by an invasive attack or probing into the internal cell structure of the PUF. Any invasive access will cause damage to the PUF and leave evidence of tampering with the intrusion [9].

IV. Adversary Model and Design Objectives

A. Threat Model

- Similar to other schemes, such as [21], [22], and [23], our scheme employs the widely recognized Dolev-Yao (DY) threat model. In the DY model, the adversary, denoted as $\mathcal{A}$, can intercept all data exchanged in the public channel. The adversary also can manipulate, delete, or replay data during the transmission process to achieve deceptive goals. In addition, suppose that the adversary is familiar with the entire process of our scheme.
- The IIoT devices may be located in remote or sparsely populated areas. Therefore, we assume that an adversary can physically capture devices and gain access to the local storage data of these smart devices. In addition, we assume that CSC and GWN are secure and trustworthy, while RC is semi-trustworthy.

TABLE I
NOTATIONS

Notation	Explanation
sk_A,sk_B	Secret key of SD_A and SD_B, respectively
pk_A,pk_B	Public key of SD_A and SD_B, respectively
ΔT	Maximum transmission delay
$h(\cdot)$	One-way hash function
C_A, C_B	Challenge of PUF of SD_A and SD_B, respectively
R_A,R_B	Response of PUF of SD_A and SD_B, respectively
ID_A	Identity of SD_A
ID_B	Identity of SD_B
X_{GWN-A}	Secret key of GWN for SD_A
X_{GWN-B}	Secret key of GWN for SD_B
r_i, m_i, a, B_0	Random numbers
$\oplus$	Bitwise XOR operation
$\|\|$	Concatenation operation

- We assume that the gateway and RC are in secure environments, meaning they cannot be accessed or compromised by adversaries. However, we also assume that the administrator of the RC could be an adversary and is capable of conducting an insider attack to compromise the registration request message.

B. System Objectives

The goal of our proposed scheme is to achieve anonymous and efficient identity authentication in the IIoT environment. We believe that authentication and key agreement schemes in the IIoT environment should fulfill the following security requirements.

1) Mutual Authentication: Due to untrustworthy communication entities, unauthorized malicious devices may leak sensitive data. Therefore, before establishing the session key and communication, mutual authentication must be implemented for SD_A, GWN, and SD_B.

2) Session Key Establishment: For entities engaged in frequent communication, using a symmetric key as a session key is to improve system efficiency. Therefore, after mutual authentication, it is necessary for SD_A and SD_B to establish a symmetric session key for subsequent communication.

3) Resisting Known Attacks: The scheme must be capable of resisting existing attacks, such as impersonation, man-in-the-middle, data tampering, replay, capture devices, insider privilege attacks, etc.

4) Efficient Performance: IIoT devices typically have limited resources, so it is important to minimize resource overhead on the device side. Meanwhile, although gateway resources are abundant, a gateway is often responsible for the security authentication and communication of multiple devices. Therefore, the resources of a gateway are not unlimited, and it is necessary to keep the gateway overhead within a small range.

V. The Proposed Scheme

In this section, we will provide a detailed introduction to the various stages of our scheme. The proposed scheme includes the following stages: 1) system initialization, 2) device registration, 3) upload and authorization, 4) authentication. The symbols used in the proposed scheme are listed in Table I.

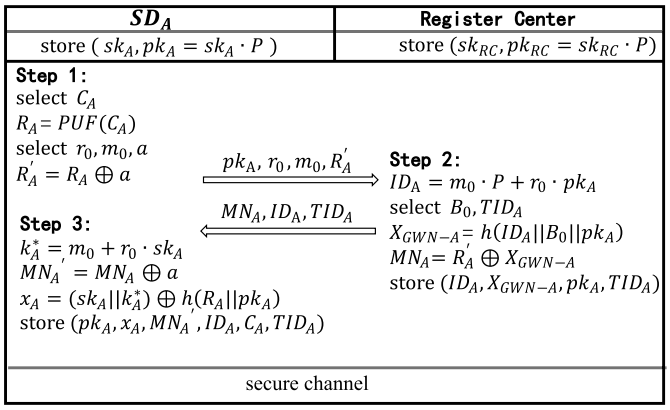


Fig. 2. Register process.

A. System Initialization

In this step, RC initializes some relevant parameters for the system. First, RC selects an elliptic curve $E : y^2 = x^3 + ax + b \ mod \ p$ over $\mathbb{F}_p$, where $a, b \in \mathbb{F}_p$ and $\mathbb{F}_p$ is a finite field determined by a large prime number p. Then RC uses points on E to construct a cyclic group $\mathbb{G}$ with the order q and P, where P is a generator of the group. Besides, RC selects a random number $sk_{RC} \in \mathbb{Z}_q^*$ as the private key and computes the corresponding public key $pk_{RC} = sk_{RC} \cdot P$. Eventually, RC chooses a hash function $h : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ and publishes the common parameters to the system, which include $\{q, \mathbb{G}, E, P, pk_{RC}, \mathbb{Z}_q^*, h\}$.

B. Smart Device Registration

After the device is produced by the manufacturer, it needs to be registered with the Register Center in a secure environment. The registration process for SD_A and SD_B is the same. Therefore, only the specific registration process of SD_A is shown in Fig. 2:

$$R_1 : SD_A \rightarrow RC : MSG_1 = \left\{pk_A, r_0, m_0, R_A'\right\}$$

SD_A randomly selects a challenge C_A and uses PUF to compute $R_A = PUF(C_A)$, and generates random values r_0, m_0, $a \in \mathbb{Z}_q^*$. Then, SD_A calculates $R_A' = R_A \oplus a$. Subsequently, SD_A submits a registration request (pk_A, r_0, m_0, R_A') to RC through a secure channel.

$$R_2 : RC \rightarrow SD_A : MSG_2 = \{MN_A, ID_A, TID_A\}$$

Upon receiving the registration request of SD_A, RC determines whether SD_A has previously registered by checking if ID_A already exists locally. If it exists, RC rejects the registration request. Otherwise, RC computes $ID_A = m_0 \cdot P + r_0 \cdot pk_A$. Then, RC selects a random value B_0, $TID_A \in \mathbb{Z}_q^*$ and calculates $X_{GWN-A} = h(ID_A||B_0||pk_A)$, $MN_A = R_A' \oplus X_{GWN-A}$. Nexts, RC stores $(ID_A, X_{GWN-A}, pk_A, TID_A)$ locally. Finally, RC sends (MN_A, ID_A, TID_A) to SD_A through a secure channel. It should be noted that only entities with trapdoor keys can construct random numbers that can calculate ID_A.

R_3: Final step

After receiving (MN_A, ID_A, TID_A) from RC, SD_A computes $k_A^* = m_0 + r_0 \cdot sk_A$, $MN_A' = MN_A \oplus a$ and $x_A = (sk_A||k_A^*) \oplus h(R_A||pk_A)$. Finally, it stores $(pk_A, x_A, MN_A', ID_A, C_A, TID_A)$ locally.

C. Upload and Authorization

After devices register with RC, RC cannot directly communicate with the gateways. Therefore, it is necessary to upload the data to CSC securely. This process is step 2 in Fig. 1. The RC securely transmits the device registration information $(ID_A, X_{GWN-A}, pk_A, TID_A)$ to the CSC. The CSC is responsible for an extensive geographical area, which can encompass multiple industrial parks, involving numerous factories.

Each factory contains multiple gateways, and these gateways are responsible for managing IIoT devices within their respective factories. The gateway can only serve devices after authorization. The authorization process involves CSC securely uploading device registration information to the gateway. The gateway is authorized by CSC and is then responsible for a specific area in the factory.

D. Authentication Step

This step is the key to the proposed scheme. If SD_A wants to securely communicate with SD_B, it needs to establish a secure session key SK through this step. The authentication step is shown in Fig. 3. The detailed authentication steps of the system are as follows:

$$A_1 : SD_A \rightarrow GWN : MSG_1 = \left\{L_1, m_1, DID_1', T_1, TID_A\right\}$$

The SD_A computes $R_A = PUF(C_A)$, $(sk_A||k_A^*) = x_A \oplus h(R_A||pk_A)$, $X_{GWN-A} = MN_A' \oplus R_A$. Then, it generates a random number $r_1 \in \mathbb{Z}_q^*$ and current system timestamp T_1, calculates $r_1' = h(r_1||T_1)$, $m_1 = k_A^* - r_1' \cdot sk_A$, $L_1 = r_1 \oplus h(X_{GWN-A}||T_1||m_1||pk_A)$, and $DID_1' = h(r_1'||m_1||T_1) \oplus ID_B$. Finally, the authentication request $MSG_1 = \left\{L_1, m_1, DID_1', T_1, TID_A\right\}$ is sent to GWN.

The GWN then checks whether TID_A exists locally. If it exists, the authentication process begins directly. Otherwise, GWN forwards the request to the CSC. Then, CSC verifies the request, and the verification process is similar to step A_2. If verification is passed, CSC sends $(ID_A, X_{GWN-A}, pk_A, TID_A)$ to the GWN through a secure channel. Subsequently, authentication and key agreement between devices are managed by the gateway. If the verification fails, CSC refuses to disclose device registration data to the GWN, and the entire authentication process is terminated. Furthermore, if CSC does not have relevant information on the device, CSC will base the certificate and sign the request using pk_{RC}, then forward it to RC.

$$A_2 : GWN \rightarrow SD_B : MSG_2 = \{L_2, M_2, T_2\}$$

Upon receiving the authentication request MSG_1, GWN generates the timestamp T_1' to verify T_1. If the verification fails, the authentication process terminates. Otherwise, GWN calculates $r_1 = L_1 \oplus h(X_{GWN-A}||T_1||m_1||pk_A)$, $r_1' = h(r_1||T_1)$, $ID_A' = m_1 \cdot P + r_1' \cdot pk_A$. If $ID_A' = ID_A$,

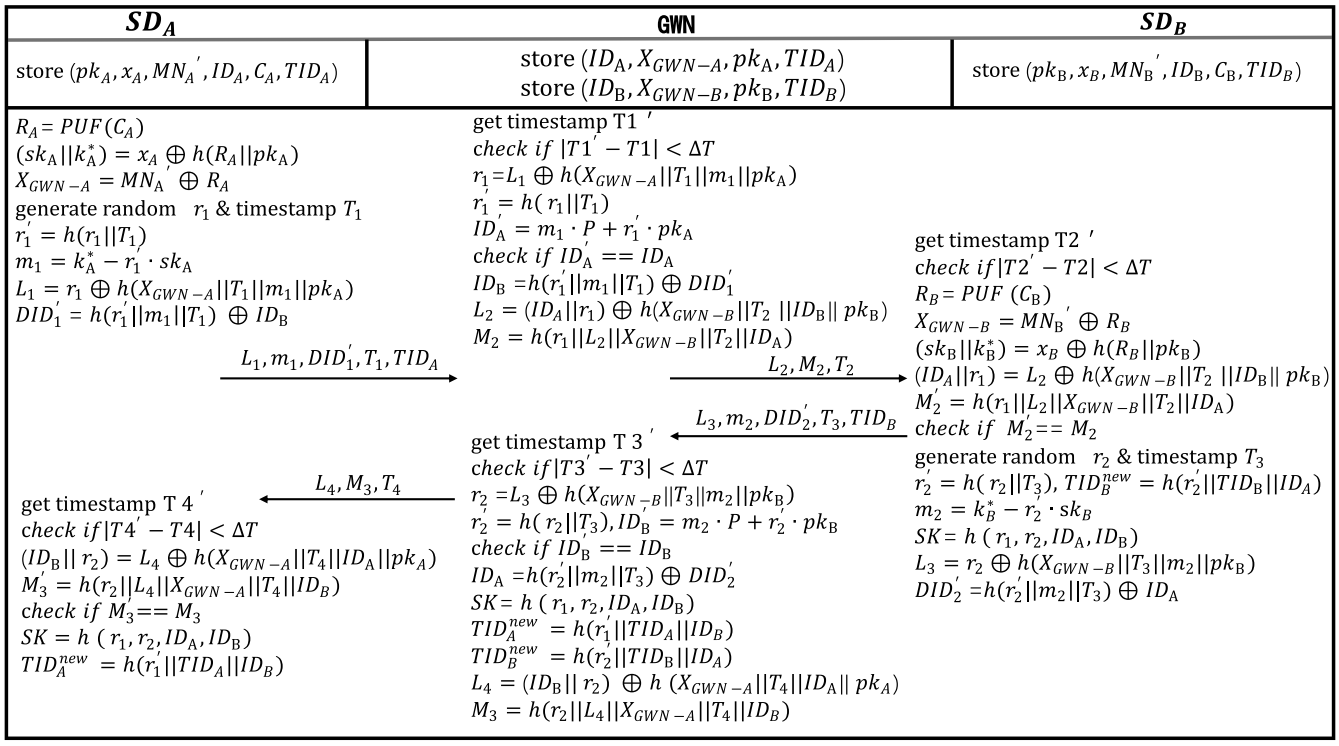


Fig. 3. Authentication process.

the authentication of SD_A is successful. Then, GWN generates timestamp T_2 and computes $ID_B = h(r_1'||m_1||T_1) \oplus DID_1'$, $L_2 = (ID_A||r_1) \oplus h(X_{GWN-B}||T_2||ID_B||pk_B)$, and $M_2 = h(r_1||L_2||X_{GWN-B}||T_2||ID_A)$. Finally, GWN sends MSG_2 to SD_B.

$$A3: SD_B \rightarrow GWN: MSG_3 = \left\{L_3, m_2, DID_2', T_3, TID_B\right\}$$

Upon receiving MSG_2, SD_B obtains the timestamp T_2' to verify T_2. If the verification fails, the authentication process terminates. Otherwise, SD_B calculates $R_B = PUF(C_B)$, $X_{GWN-B} = MN_B' \oplus R_B$, $(sk_B||k_B^*) = x_B \oplus h(R_B||pk_B)$, $(ID_A||r_1) = L_2 \oplus h(X_{GWN-B}||T_2||ID_B||pk_B)$, $M_2' = h(r_1||L_2||X_{GWN-B}||T_2||ID_A)$, and then checks if $M_2' = M_2$. If it matches, authentication of GWN is successful. Next, SD_B generates a random number $r_2 \in \mathbb{Z}_q^*$ and timestamp T_3, computes $r_2' = h(r_2||T_3)$, $TID_B^{new} = h(r_2'||TID_B||ID_A)$ and $m_2 = k_B^* - r_2' \cdot sk_B$. The session key is derived as $SK = h(r_1||r_2||ID_A||ID_B)$. SD_B further computes $L_3 = r_2 \oplus h(X_{GWN-B}||T_3||m_2||pk_B)$ and $DID_2' = h(r_2'||m_2||T_3) \oplus ID_A$. Finally, SD_B sends MSG_3 to GWN.

$$A_4: GWN \rightarrow SD_A: MSG_4 = \{L_4, M_3, T_4\}$$

Upon receiving MSG_3, GWN acquires timestamp T_3' for T_3 verification. In case of verification failure, the authentication process is terminated. If successful, GWN proceeds with the computation of $r_2 = L_3 \oplus (X_{GWN-B}||T_3||m_2||pk_B)$, $r_2' = h(r_2||T_3)$, and $ID_B' = m_2 \cdot P + r_2' \cdot pk_B$. If ID_B' equals ID_B, then the authentication of SD_B is deemed successful. Subsequently, GWN compute $ID_A = h(r_2'||m_2||T_3) \oplus DID_2'$, $SK = h(r_1||r_2||ID_A||ID_B)$, $TID_A^{new} = h(r_1'||TID_A||ID_B)$, $TID_B^{new} = h(r_2'||TID_B||ID_A)$, $L_4 = (ID_B||r_2) \oplus h(X_{GWN-A}||T_4||ID_A||pk_A)$ and $M_3 = h(r_2||L_4||X_{GWN-A}||T_4||ID_B)$. Finally, GWN transmits MSG_4 to SD_A.

A_5: Final step

Upon receiving MSG_4, SD_A obtains the timestamp T_4' for T_4 verification. If the verification fails, the authentication process is terminated. Otherwise, SD_A calculates $(ID_B || r_2) = L_4 \oplus h(X_{GWN-A} || T_4 || ID_A || pk_A)$, $M_3' = h(r_2||L_4||X_{GWN-A}||T_4||ID_B)$, and then checks if $M_3' = M_3$. If they match, authentication of GWN is considered successful. Finally, SD_A computes $SK = h(r_1, r_2, ID_A, ID_B)$ and $TID_A^{new} = h(r_1'||TID_A||ID_B)$.

VI. Security Analysis and Proof

In this section, we will establish the security of the scheme using a combination of various security-proof tools and methods. To comprehensively evaluate security, we depend on a combination of proof methods working in conjunction. This paper employs the Real-Or-Random (ROR) model, BAN Logic, Proverif, and heuristic security analysis methods to demonstrate the security of the proposed scheme.

A. Formal Security Analysis Using Real-Or-Random Model

The ROR model, a widely-recognized model [4], [5], [20], [24], serves as the foundation for our formal security analysis, demonstrating the session key security of the proposed scheme. Subsequently, we will proceed to model and analyze the capabilities of the adversary in the ROR model.

1) Participants: The proposed scheme involves three participants: SD_A, GWN, and SD_B, which we can instantiate as A_i, GWN_k, and B_j, respectively. Let $\mathcal{H}$ be the set of instances, and $\mathcal{H}^s$ be the s-th instance of $\mathcal{H}$.

2) Partnering: Two instances $\mathcal{H}^1$ and $\mathcal{H}^2$ are considered partnered if the following three conditions are satisfied simultaneously:

1) Both $\mathcal{H}^1$ and $\mathcal{H}^2$ are in an accept state; 2) Both $\mathcal{H}^1$ and $\mathcal{H}^2$ mutually authenticate each other and share the same session identifier (sk_{id}); 3) $\mathcal{H}^1$ and $\mathcal{H}^2$ are mutual partners of each other.

3) Freshness: We say that the entity $\mathcal{H}^s$ is fresh when its session keys for both $\mathcal{H}^2$ and his partner have not been exposed to the adversary.

We define some queries corresponding to the adversary's attack capabilities as below:

- $Execute\left(A_i, GWN_k, B_j\right)$ After the adversary executes this query, they can obtain all messages transmitted between the three entities.
- $Send\left(\mathcal{H}^s, m\right)$ It simulates an active attack where adversary $\mathcal{A}$ forges a message m and sends it to $\mathcal{H}^s$. If m passes $\mathcal{H}^s$'s verification, it outputs the response from $\mathcal{H}^s$.
- $Reveal\left(\mathcal{H}^s\right)$ This query simulates the scenario where the session key SK is leaked to adversary $\mathcal{A}$, which represents a known session key attack.
- $Corrupt\left(A_i, B_j\right)$ This query simulates the scenario where a device A_i or B_j is compromised, and its local data is leaked to the adversary.
- $Test\left(\mathcal{H}^s\right)$ This query is used to test the semantic security of the session key between A_i, GWN_k, and B_j. When $\mathcal{A}$ executes this query, if SK is fresh or has not yet been generated between the entities, it outputs $\perp$. Otherwise, a fair and unbiased coin is flipped. If $b = 1$, it returns the session key SK and if $b = 0$, it returns a random number with the same number of bits as the session key.

Theorem 1: When $\mathcal{A}$ assumes the role of a probabilistic polynomial-time (PPT) adversary operating within the ROR model against our proposed protocol, the advantage of $\mathcal{A}$ in compromising the semantic security of our scheme is expressed as:

$$\frac{1}{2}Adv_{\mathcal{A}}^{\mathcal{P}}(t) \le \frac{q_h^2}{2^l+1} + \frac{q_p^2}{2^\theta+1} + \frac{q_s}{2^n}$$

Here, q_h, q_p, and q_s denote the quantities of HO (Hash Oracle) queries, PO (PUF Oracle) queries, and send queries, respectively. Meanwhile, θ, l, and n represent the bit lengths of the hash function, PUF, and the device's private key, respectively.

Proof. Our proof steps are similar to those in the previous authentication schemes [20], [24], [25]. We define five games, denoted as $Gm_j^{\mathcal{A}}$, $j = 1, 2, 3, 4, 5$. We denote $Succ_{Gm_j^{\mathcal{A}}}$ as an event where the adversary correctly guesses the bit c in $Gm_j^{\mathcal{A}}$, and define the corresponding $\mathcal{A}$'s advantage probability as $Adv_{\mathcal{A}}^{\mathcal{P}}(t) = Pr\left[Succ_{Gm_j^{\mathcal{A}}}\right]$.

- $Gm_1^{\mathcal{A}}$: In $Gm_1^{\mathcal{A}}$, the model encounters an adversary's actual attack, and bit c is randomly generated before the start of the $Gm_1^{\mathcal{A}}$ game. According to the definition of random oracle semantics security, we can obtain the following:

$$Adv_{\mathcal{A}}^{\mathcal{P}}(t) = 2Pr\left[Succ_{Gm_1^{\mathcal{A}}}\right] - 1 \tag{1}$$

- $Gm_2^{\mathcal{A}}$: In this game, we simulate the scenario where the scheme is under the adversary's eavesdropping attack. The authentication phase's MSG_1, MSG_2, MSG_3, and MSG_4 are all intercepted by adversary $\mathcal{A}$. Subsequently, $\mathcal{A}$ executes a $Test$ query and checks whether the $Test$ query output is a session key SK or a random number. $SK = h(r_1||r_2||ID_A||ID_B)$, where r_1 and r_2 are both transmitted encrypted during the authentication process. Therefore, adversary $\mathcal{A}$ cannot gain any additional advantage in winning the $Gm_1^{\mathcal{A}}$ game through the $Test$ query. So we have:

$$Pr\left[Succ_{Gm_1^{\mathcal{A}}}\right] = Pr\left[Succ_{Gm_2^{\mathcal{A}}}\right] \tag{2}$$

- $Gm_3^{\mathcal{A}}$: In this game, the adversary can launch an active attack on the system by executing $Send$ query and $Hash$ query. This attack allows the adversary to forge messages with the aim of passing the receiver's verification. However, each message is hashed and related to secret random numbers, keys, and timestamps. It is very difficult for adversary $\mathcal{A}$ to find the correct collision probability. According to the birthday paradox, we have:

$$\mid Pr\left[Succ_{Gm_2^{\mathcal{A}}}\right] - Pr\left[Succ_{Gm_3^{\mathcal{A}}}\right] \mid \le \frac{q_h^2}{2 \mid Hash \mid} \tag{3}$$

- $Gm_4^{\mathcal{A}}$: In this game, HO queries are replaced with PO queries, and the analysis process is similar to $Gm_3^{\mathcal{A}}$. We get:

$$\mid Pr\left[Succ_{Gm_4^{\mathcal{A}}}\right] - Pr\left[Succ_{Gm_3^{\mathcal{A}}}\right] \mid \le \frac{q_p^2}{2^\theta+1} \tag{4}$$

- $Gm_5^{\mathcal{A}}$: In the final game, with the ability to use the $Corrupt$ oracle, $Gm_3^{\mathcal{A}}$ can be transformed into $Gm_4^{\mathcal{A}}$. The adversary can attempt to acquire the information stored on SD_A $\{pk_A, x_A, MN'_A, ID_A, C_A, TID_A\}$. Again, $\mathcal{A}$ may try to acquire the sk_A and R_A from the information stored in SD_A. Meanwhile, $R_A = PUF(C_A)$, $X_{GWN-A} = MN'_A \oplus R_A$, $r_1 = L_1 \oplus h\left(X_{GWN-A}||T_1||m_1||pk_A\right)$. The adversary can only use q_h and q_s queries and cannot use the PUF in SD_A, as attempting to do so can easily lead to damage to the device's PUF. The adversary can only generate the correct collisions through queries to compute r_1 and r_2. $Gm_3^{\mathcal{A}}$ and $Gm_4^{\mathcal{A}}$ are indistinguishable when $\mathcal{A}$ cannot successfully construct the correct message. Therefore, we have:

$$\mid Pr\left[Succ_{Gm_3^{\mathcal{A}}}\right] - Pr\left[Succ_{Gm_4^{\mathcal{A}}}\right] \mid \le \frac{q_s^2}{2^n} \tag{5}$$

After $\mathcal{A}$ has exhausted all query options and attempted various attacks to compromise the security of our scheme, its only remaining path to victory is the successful guessing of bit b by querying the $Test$ query. Therefore, we have:

$$\mid Pr\left[Succ_{Gm_5^{\mathcal{A}}}\right] = \frac{1}{2} \tag{6}$$

Based on the triangle inequality $\mid a \pm b \mid \le \mid a \mid + \mid b \mid$, and equations (1)-(6), we deduce:

$$\frac{1}{2}Adv_{\mathcal{A}}^{\mathcal{P}}(t) \le \frac{q_h^2}{2^l+1} + \frac{q_p^2}{2^\theta+1} + \frac{q_s}{2^n}$$

TABLE II
BASIC LOGICAL SYMBOLS IN BAN LOGIC

Notation	Explanation
$P \lhd X$	P sees X, principal P receives message X and can read and use it.
$P \mid\equiv X$	P believes X
$P \mid\Rightarrow X$	P has jurisdiction over message X
$\sharp(X)$	refers to the message X is fresh.
$P \overset{K}{\longleftrightarrow} Q$	P and Q share a common key k
$\langle X \rangle_K$	X has been encrypted using K.
$P \mid\sim X$	P said X means that P has previously sent message X.
$P \overset{Y}{=} Q$	Y is a secret known to both P and Q.

B. BAN Logic

BAN Logic is an essential tool for security proofs, as introduced in [26], and serves as a powerful analysis tool to clearly establish whether a scheme meets its intended objectives. It can be employed to demonstrate the authenticity of participants and the security of shared keys among them [27]. However, it is important to acknowledge that BAN Logic has its limitations, including its inability to express certain events, limited to a specific domain, not applicable to all security issues, etc [28]. The primitives of BAN Logic are provided in Table II.

The fundamental rules of BAN Logic are as follows:

- R1 (Message-meaning rule): $\frac{P\mid\equiv P\overset{K}{\longleftrightarrow}Q, P\lhd\{X\}_K}{P\mid\equiv Q\mid\sim X}$ or $\frac{P\mid\equiv P\overset{Y}{=}Q, P\lhd\langle X\rangle_Y}{P\mid\equiv Q\mid\sim X}$
- R2 (Nonce-verification rule): $\frac{P\mid\equiv\sharp(X), P\mid\equiv Q\mid\sim X}{P\mid\equiv Q\mid\equiv X}$
- R3 (Jurisdiction rule): $\frac{P\mid\equiv Q\mid\Rightarrow X, P\mid\equiv Q\mid\equiv X}{P\mid\equiv X}$
- R4 (Freshness-conjuncatenation rule): $\frac{P\mid\equiv\sharp(X)}{P\mid\equiv\sharp(X,Y)}$
- R5 (Session keys rule): $\frac{P\mid\equiv\#(X), P\mid\equiv Q\mid\equiv X}{P\mid\equiv P\overset{K}{\longleftrightarrow}Q}$

First and foremost, we delineate eight demonstrative goals for the proposed scheme (A_i, B_j and G represent SD_A, SD_B, and Gateway, respectively). Meanwhile, it should be noted that the gateway has $r1$, $r2$, ID_A, and ID_B, so the gateway can calculate SK also.

- $Goal1: A_i \mid\equiv B_j \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} B_j\right)$
- $Goal2: A_i \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} B_j\right)$
- $Goal3: B_j \mid\equiv A_i \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} B_j\right)$
- $Goal4: B_j \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} B_j\right)$
- $Goal5: G \mid\equiv \left(G \overset{SK}{\longleftrightarrow} A_i\right)$
- $Goal6: G \mid\equiv A_i \mid\equiv \left(G \overset{SK}{\longleftrightarrow} A_i\right)$
- $Goal7: G \mid\equiv \left(G \overset{SK}{\longleftrightarrow} B_j\right)$
- $Goal8: G \mid\equiv B_j \mid\equiv \left(G \overset{SK}{\longleftrightarrow} B_j\right)$

Secondly, in the proposed scheme, message communication can be transformed into several ideal forms as follows:

- $MSG_1: A_i \rightarrow GW: \left\{L_1, m_1, DID_1', T_1, TID_A\right\}:$ $\left\{\langle r_1\rangle_{(X_{G-A}||pk_A)}, m_1, \langle ID_B\rangle_{r_1'}, T_1, TID_A\right\}:$
- $MSG_2: G \rightarrow B_j: \{L_2, M_2, T_2\}:$ $\left\{\langle ID_A||r_1\rangle_{(X_{G-B}||ID_B||pk_B)}, \langle r_1||ID_A\rangle_{X_{G-B}}, T_2\right\}$
- $MSG_3: B_j \rightarrow GW: \left\{L_3, m_2, DID_2', T_3, TID_B\right\}:$ $\left\{\langle r_2\rangle_{(X_{G-B}||pk_B)}, m_2, \langle ID_A\rangle_{r_2'}, T_3, TID_B\right\}$
- $MSG_4: GW \rightarrow A_i: \{L_4, M_3, T_4\}:$ $\left\{\langle(ID_B||r_2)\rangle_{(X_{G-A}||ID_A||pk_A)}, \langle r_2||ID_A\rangle_{X_{G-A}}, T_4\right\}$

Then, we list the necessary assumptions for the proposed scheme:

- $A1: G \mid\equiv \sharp\{T_1\}$
- $A2: B_j \mid\equiv \sharp\{T_2\}$
- $A3: G \mid\equiv \sharp\{T_3\}$
- $A4: A_i \mid\equiv \sharp\{T_4\}$
- $A5: A_i \mid\equiv \left(A_i \overset{(X_{G-A}||ID_A||pk_A)}{\longleftrightarrow} G\right)$
- $A6: G \mid\equiv \left(G \overset{(X_{G-A}||ID_A||pk_A)}{\longleftrightarrow} A_i\right)$
- $A7: G \mid\equiv \left(G \overset{(X_{G-B}||ID_B||pk_B)}{\longleftrightarrow} B_j\right)$
- $A8: B_j \mid\equiv \left(B_j \overset{(X_{G-B}||ID_B||pk_B)}{\longleftrightarrow} G\right)$
- $A9: G \mid\equiv A_i \mid\Rightarrow \{r_1, SK\}$
- $A10: G \mid\equiv B_j \mid\Rightarrow \{r_2, SK\}$
- $A11: A_i \mid\equiv B_j \mid\Rightarrow \{r_2, SK\}$
- $A12: A_i \mid\equiv G \mid\Rightarrow \{r_2, SK\}$
- $A13: B_j \mid\equiv G \mid\Rightarrow \{r_1, SK\}$
- $A14: B_j \mid\equiv A_i \mid\Rightarrow \{r_1, SK\}$

Then, the validation process of the proposal unfolds as follows: Building upon MSG_1, we acquire:

- S1: G $\lhd MSG_1$

Based on assumptions $A6$, S1, and R1, we can deduce the following:

- S2: G$\mid\equiv A_i \sim MSG_1$

According to $A1$ and R4, we get:

- S3: G$\mid\equiv \sharp\ MSG_1$

Based on S2, S3, and R2, we get:

- S4: G$\mid\equiv A_i \mid\equiv MSG_1$

Based on MSG_2, we can conclude:

- S5: $B_j \lhd MSG_2$

According to A8, S5, and R1, we have:

- S6:$B_j \mid\equiv G \mid\sim MSG_2$

Combining A2 and R4, we can infer:

- S7:$B_j \mid\equiv \sharp\ MSG_2$

Based on S6, S7, and R2, we can get:

- S8:$B_j \mid\equiv G \mid\equiv MSG_2$

According to MSG_3, we obtain:

- S9: $G \lhd MSG_3$

Based on A7, S9, and R1, we have:

- S10: G$\mid\equiv B_j \sim MSG_3$

Based on A3 and R4, we can infer:

- S11: G$\mid\equiv \sharp\ MSG_3$

Based on S10, S11, and R2, we can conclude:

- S12: G$\mid\equiv B_j \mid\equiv MSG_3$

Based on MSG_4, we get:

- S13: $A_i \lhd MSG_4$

According to A5, S13 and R1, we have:

- S14: $A_i \mid\equiv G \sim MSG_4$

Based on A1, A2, A3, A4, and R4, we can deduce:

- S15: $A_i \mid\equiv \sharp MSG_4$

Based on S14, S15, and R2, we can conclude:

- S16: $A_i \mid\equiv G \mid\equiv MSG_4$

Based on S4, A6, A9, and R3, we get:

- S17: $G \mid\equiv MSG_1$

From S3, S4, S17 and R5, we get:

- S18: $G \mid\equiv A_i \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} G\right)$ **(Goal 6)**

From S18, A9 and R3, we get:

- S19: $G \mid\equiv \left(G \overset{SK}{\longleftrightarrow} A_i\right)$ **(Goal 5)**

From S12, A10 and R3, we get:

- S20: $G \mid\equiv MSG_1$

From S11, S12, S20 and R5, we get:

- S21: $G \mid\equiv B_j \mid\equiv \left(B_j \overset{SK}{\longleftrightarrow} G\right)$ **(Goal 8)**

From S21, A10 and R3, we get:

- S22: $G \mid\equiv \left(G \overset{SK}{\longleftrightarrow} B_j\right)$ **(Goal 7)**

From S8, A13 and R3, we get:

- S23: $B_j \mid\equiv MSG_2$

From S7, S8, S23 and R5, we get:

- S24: $B_j \mid\equiv G \mid\equiv \left(G \overset{SK}{\longleftrightarrow} B_j\right)$

From S24, A13 and R3, we get:

- S25: $B_j \mid\equiv \left(G \overset{SK}{\longleftrightarrow} B_j\right)$

From S16, A12 and R3, we get:

- S26: $A_i \mid\equiv MSG_4$

From S15, S16, S26 and R5, we get:

- S27: $A_i \mid\equiv G \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} G\right)$

From S27, A12 and R3, we get:

- S28: $A_i \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} G\right)$

From S18 and S24, we get:

- S29: $A_i \mid\equiv B_j \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} B_j\right)$ **(Goal 1)**

From S21 and S27, we get:

- S30: $B_j \mid\equiv A_i \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} B_j\right)$ **(Goal 3)**

From S29, A12 and R3, we get:

- S31: $A_i \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} B_j\right)$ **(Goal 2)**

Based on S30, A14 and R3, we get:

- S32: $B_j \mid\equiv \left(A_i \overset{SK}{\longleftrightarrow} B_j\right)$ **(Goal 4)**

Based on the above analysis, we have demonstrated eight objectives and concluded that this scheme ensures mutual authentication among SD_A, GWN, and SD_B. Furthermore, it has been proven that a session key SK has been established between them.

```
Verification summary:

Query not attacker(DataA[]) is true.

Query not attacker(DataB[]) is true.

Query not attacker(DataC[]) is true.

Query not attacker(DataD[]) is true.

Query not attacker(DataE[]) is true.

Query not attacker(DataF[]) is true.

Query not attacker(DataG[]) is true.

Query inj-event(NodeA_GWN_end(t)) ==> inj-event(NodeA_GWN_begin(t)) is true.

Query event(GWN_Device_end(t)) ==> event(GWN_Device_begin(t)) is true.

Query event(GWN_NodeA_end(t)) ==> event(GWN_NodeA_begin(t)) is true.

Query inj-event(Device_GWN_end(t)) ==> inj-event(Device_GWN_begin(t)) is true.
```

Fig. 4. Proof results of proverif.

C. Formal Security Proof Using Proverif

Proverif is a mature automated security proof tool [29], capable of verifying reachability properties and correspondence assertions in protocols. However, to the best of our knowledge, Proverif may have limitations in analyzing certain properties [30], such as device capture attacks. Our analysis results are presented in Fig. 4.

In the proverif proof, we define the parameters DataA, DataB, DataC, DataD, DataE, and DataF to test whether r_1, ID_A, X_{GWN-A}, r_2, ID_B, X_{GWN-B}, and SK in the system has leaked to the adversary. We define eight events to represent different phases of the authentication process and use four correspondence assertions to detect the sequence of events. For the specific proof explanation and code see link.[1] The experimental results show that each parameter has not been leaked to the adversary, and the consistency assertion is successful, indicating that the authentication process is accurate.

D. Heuristic Security Analysis

Based on the formal security proof mentioned above, we conduct a heuristic security analysis (informal) on the proposed scheme:

1) Mutual Authentication: The authentication of SD_A and GWN is based on the construction of M_2 and ID_A, and the authentication of the GWN and SD_B is based on the construction of M_3 and ID_B. The entity authenticates the other party by checking the message. The GWN checks ID_A and ID_B to authenticate SD_A and SD_B. Then both SD_A and SD_B check the message M_2 and M_3 to authenticate GWN. Thus, the three-party SD_A, GWN, and SD_B complete mutual authentication.

2) Session Key Establishment: In the proposed scheme, SD_A and SD_B both establish a session key $SK = h(r_1||r_2||ID_A||ID_B)$, where r_1 and r_2 are random numbers from SD_A and SD_B respectively. Furthermore, only an adversary possessing the private keys ID_A, ID_B, pk_A, pk_B, X_{GWN-A} and X_{GWN-B} can decrypt the ciphertext to obtain r_1, and r_2. Hence, calculating the session key SK is difficult.

[1] https://github.com/qyzhang92/Codes-for-DSLAKA-IIoT

3) Device Anonymity: In the communication process of the scheme, SD_A and SD_B will not expose the real ID in the public channel. It should be noted that even if the ID is leaked, due to the nature of the chameleon hash function, the adversary cannot forge a message that can be authenticated through GWN. Among them, MSG_1, MSG_2, MSG_3, and MSG_4 all involve random numbers and timestamps, which increase the randomness of the messages. Meanwhile, multiple devices in the system simultaneously undergo authentication through the same gateway. Therefore, the adversary cannot distinguish whether different messages come from the same device or the same round of communication, further strengthening the anonymity of the device.

4) Protection Against Data Tampering Attacks: MSG_1: The adversary $\mathcal{A}$ cannot calculate r_1 without knowledge of X_{GWN-A} and pk_A. And because the adversary does not have the trapdoor of the chameleon hash function, the forgery of r_1 cannot be completed. Assuming $\mathcal{A}$ constructs L_1' and transmits it to GWN. However, the random number r_1 calculated by L_1' cannot pass the verification of GWN. So the adversary cannot successfully tamper with MSG_1.

MSG_2: The gateway authentication SD_B and ensures data integrity through check if $M_2 \stackrel{?}{=} M_2'$, the adversary does not have X_{GWN-B}, ID_A, ID_B and r_1, so M_2 cannot be forged. If MSG_2 is tampered with, the calculated M_2' by SD_B will fail the verification. The analysis of MSG_3 and MSG_4 is similar to that of MSG_1 and MSG_2, so we will not repeat the analysis here.

5) Protection Against Replay Attacks: The scheme maintains synchronization by employing a widely accepted timestamp verification scheme [31], [32], [33]. When the adversary attempts to replay previous messages, the data recipient verifies whether $| T_1' - T_1 | < \Delta T$. If this condition is fulfilled, the verification continues to the next step. Otherwise, the process is terminated.

6) Protection Against Impersonation Attacks: Impersonating SD_A: Based on the properties of PUF and the chameleon hash function, the adversary cannot use PUF to compute R_A. Therefore the adversary cannot compute the trapdoor sk_A and k_A^*. Then, the adversary cannot construct the correct r_1. Meanwhile, without X_{GWN-A}, the adversary cannot decrypt L_1 to obtain r_1. As a result, calculating $ID_A' = m_1 \cdot P + r_1' \cdot pk_A$ is impossible. Since the situation of SD_B and SD_A is similar, the adversary cannot impersonate SD_A and SD_B.

Impersonating GWN: We have assumed that the gateway is secure, and the adversary cannot obtain local data from the gateway. Therefore, the adversary cannot forge L_2 and M_2 that can pass SD_B verification, and this also applies to SD_A. Thus, this means that the adversary cannot impersonate the GWN.

7) Device Capture Attacks: Assuming SD_A is acquired by an adversary, the adversary can obtain SD_A's all local data $(pk_A, x_A, MN_A', ID_A, C_A, TID_A)$ by using the power analysis attacks [24]. Note that $x_A = (sk_A || k_A^*) \oplus h(R_A || pk_A)$, and the R_A can be calculated by PUF (C_A). However, external usage of micro detection or other invasive techniques to access the device's PUF may result in damage, rendering the PUF unusable [34]. Then, the adversary cannot calculate R_A, sk_A, and k_A^*. Therefore, the forged m_1 and r_1 cannot be verified through the chameleon hash of the gateway, and even the ID_A of SD_A remains unknown.

TABLE III
COMPARISON OF SECURITY PROPERTIES

	[35]	[4]	[5]	[36]	[25]	[37]	[38]	ours
Device-Side Lightweight	✗	✗	✗	✗	✗	✗	✗	✓
Man-in-the-middle Attack	✓	✓	✓	✓	✓	✓	✓	✓
Replay Attack	✓	✓	✓	✗	✓	✓	✓	✓
Impersonation Attacks	✓	✓	✗	✓	✓	✓	✓	✓
Privileged-Insider Attack	✗	✓	✓	✓	✓	✓	✗	✓
Anonymity	✗	✓	✓	✓	✓	✓	✓	✓
Device Capture Attack	✗	✓	✗	✓	✓	✓	✓	✓
Forward Secret	✓	✓	✓	✓	✓	✓	✓	✓
No Password Exposure	✗	✓	✗	✓	✓	✓	✗	✓
Eavesdropping Attacks	✓	✓	✗	✓	✓	✓	✓	✓
Data Tampering Attack	✓	✓	✗	✓	✓	✓	✓	✓

TABLE IV
BASIC OPERATION TIMES

Operation	Description	Time Cost (ms)
T_{cm}	Chebyshev chaotic-map	7.983
T_{sm}	Scalar multiplication	7.716
T_{msm}	Multi-Scalar multiplication	10.734
T_h	Hash function	0.0013
T_{puf}	PUF generation	0.655

8) Forward Secret: The session key SK is generated based on random numbers r_1, r_2, ID_A, ID_B. These random numbers are temporarily generated by each device every time. This means that even if the session key SK is leaked during the current communication, the previous session keys remain secure.

9) Internal Privilege Attack: Assume that a privileged-insider administrator personnel acquires the registration request $\{pk_A, r_0, m_0, R_A'\}$, and then leaks the request to the adversary $\mathcal{A}$. Now, we assume that $\mathcal{A}$ acquires the local data of SD_A, denoted as $(pk_A, x_A, MN_A', ID_A, C_A, TID_A)$, through device capture attacks. However, calculating R_A and X_{GWN-A} from R_A' and MN_A is difficult without knowledge of a. Subsequently, $\mathcal{A}$ gaining access or guessing the the private key (sk_A, k_A^*) is computationally infeasible without knowledge of R_A. Therefore, the system can effectively resist internal privilege attacks.

VII. PERFORMANCE ANALYSIS

A. System Security Property Analysis

We analyzed and compared the proposed scheme and the security attributes of [4], [5], [25], [35], [36], [37], and [38], all of which are from the latest research in recent years. In Table III, Note ✓ indicates that the scheme has the security attribute, and note ✗ indicates that the scheme does not have the security attribute. In Table III, it can be clearly observed that our scheme has more security attributes than [25], [35], [36], and [38] and has similar security attributes compared to [4] and [37]. In summary, the proposed scheme performs the best in terms of system security.

B. Comparison of Simulation Calculation Costs

We define T_{cm}, T_{sm}, T_{msm}, T_h and T_{puf} as the operation times for Chebyshev chaotic-map, scalar multiplication,

TABLE V
COMPARISON OF COMPUTATIONAL COSTS (MS)

Ref	SD_A	GWN	SD_B	Cloud	Total	Theoretical	Simulation	Device-Side
[4]	$2T_{cm}+15T_h$	$10T_h$	$2T_{cm}+6T_h$	–	$4T_{cm}+31T_h$	31.612	33.412	32.845
[5]	$7T_h+2T_{sm}$	$9T_h$	$5T_h+2T_{sm}$	–	$21T_h+4T_{sm}$	30.891	36.246	35.642
[36]	$6T_h+2T_{sm}$	$7T_h+T_{sm}$	$3T_h$	–	$16T_h+3T_{sm}$	23.169	26.442	18.442
[25]	$3T_{sm}+8T_h$	$9T_h$	$2T_{sm}+4T_h$	$T_{sm}+10T_h$	$5T_{sm}+31T_h$	46.336	54.829	44.847
[37]	$5T_{sm}+16T_h$	$2T_{sm}+11T_h$	$4T_{sm}+7T_h$	–	$11T_{sm}+34T_h$	84.920	90.997	74.037
[38]	$8T_h+3T_{sm}$	$7T_h+T_{sm}$	$4T_h+2T_{sm}$	–	$19T_h+6T_{sm}$	46.321	51.859	42.920
Ours	$8T_h+T_{puf}$	$12T_h+2T_{msm}$	$8T_h+T_{puf}$	–	$29T_h+2T_{msm}+2T_{puf}$	22.815	23.935	3.988

multi-scalar multiplication, hash function (using SHA-3) and PUF generation operation, respectively (elliptic curve using nist-256). We conducted 1000 iterations of each cryptographic operation and calculated the average time taken on a desktop computer with an Intel i7-11700 2.50 GHz CPU and 16 GB of memory running in Python 3.9.8 while ignoring the time required for XOR operations. The computation results are presented in Table IV. In Table V, we have calculated and compared the theoretical computational cost and the overhead of a one-time authentication simulation of the system. In addition, "–" indicates that this entity is not involved in system authentication.

In Table V, it is evident that our scheme involves lightweight operations on the device side, while ECC point multiplication is to be handled by the GWN. Therefore the proposed scheme is better suited for deployment in resource-constrained IIoT environments. We obtain the theoretical time by calculating the time required for each basic operation.

To enhance the credibility of the calculation cost results, we conducted a Python simulation to determine the total authentication overhead, providing a more meaningful basis for comparison and reference. The simulation overhead of our proposed scheme is 23.935 ms, significantly lower than that of most other schemes and slightly less than the overhead of [36]. The last column of Table V, labeled "Device-side", represents the total computation overhead for both the SD_A and SD_B devices. When calculating the proportion of performance improvement, for example, let the average device-side time of all compared schemes be denoted as $T_{avg} = 41.46$ ms, the device-side overhead of our scheme denoted as $T_{our} = 3.988$ ms, compute $(1-(T_{our}/T_{avg}))\cdot 100\% = 90.38\%$. This indicates that the device-side overhead of our scheme decreased by approximately 90.38% compared to other schemes on average.

Fig. 5 is also a comparison of computational costs, but it more intuitively displays the cost comparison between different entities of each scheme. Fig. 5 clearly demonstrates that the device side of our proposed scheme is extremely lightweight, and the gateway overhead is also controlled within a reasonable range. For specific code implementation details, see Link.[2]

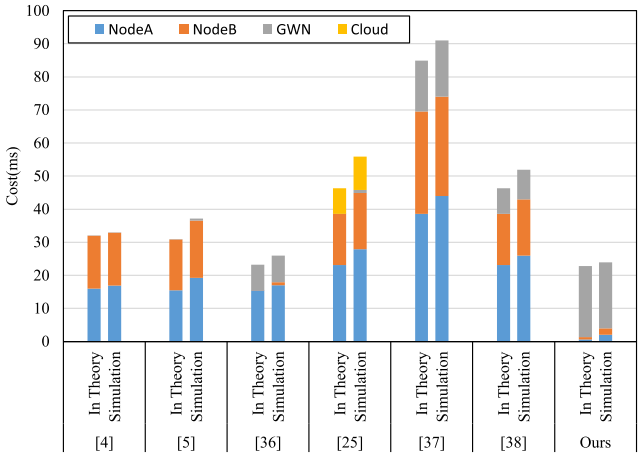


Fig. 5. Computation costs.

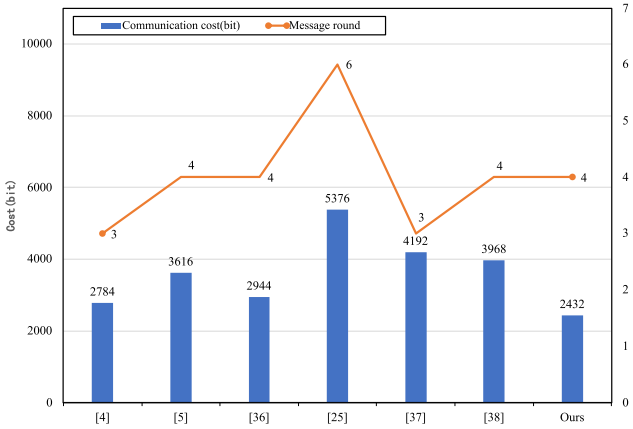


Fig. 6. Communication rounds and overhead.

C. Communication Overhead Comparison

We conducted a comparative analysis of the communication rounds and communication overhead in the authentication process for different schemes. In our scheme, the lengths of identity, random numbers, and AES symmetric keys are all set to 128 bits. The ECC point, private key, PUF response, challenge, hash results, and system timestamp have lengths of 512 bits, 256 bits, 128 bits, 128 bits, 256 bits, and 32 bits, respectively. The comparison of communication overhead and transmission rounds is presented in Fig. 6. The proposed scheme involves four messages with the following sizes: $|MSG_1| = (128+256+128+32+128) = 672$ bits, $|MSG_2| = (256+256+32) = 544$ bits, $|MSG_3| = (128+256+128+32+128) = 672$ bits, and $|MSG_4| = (256+256+32) = 544$ bits. Hence, the total communication cost of our scheme is $(672+544+672+544) = 2432$ bits. In the absence of considering communication rounds, it is evident that the communication cost of the proposed scheme is

[2]https://github.com/qyzhang92/Codes-for-DSLAKA-IIoT

lower than that of other schemes. However, the communication rounds have a certain impact on communication efficiency. Therefore, in practical environments, the communication efficiency of the schemes [4], [37] may be superior to our proposed scheme.

VIII. Conclusion

In this paper, we discuss the current situation of the IIoT and its security challenges. To address these challenges, we propose a novel lightweight anonymous AKA scheme for the IIoT environment based on the PUF and chameleon hash functions. The proposed scheme is lightweight in terms of computation and communication, and the resource cost on the device side is significantly lower than that on the gateway. Compared to other related schemes, our scheme has lower resource costs and is more suitable for resource-constrained IIoT devices. We use the BAN logic, ROR model, Proverif tool, and heuristic security analysis to perform a rigorous security analysis and proof, demonstrating that our scheme can resist multiple known attacks.

Acknowledgment

The authors are very grateful to the anonymous referees for their detailed comments and suggestions regarding this article.

References

[1] Precedence Research. (Sep. 2022). *Industrial IoT Market Global Industry Analysis, Size, Share, Growth, Trends, Regional Outlook, and Forecast2022-2030*. [Online]. Available: https://www.precedenceresearch.com/industrial-iot-market

[2] J. Cui, F. Wang, Q. Zhang, C. Gu, and H. Zhong, "Efficient batch authentication scheme based on edge computing in IIoT," *IEEE Trans. Netw. Service Manage.*, vol. 20, no. 1, pp. 357–368, Mar. 2023.

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

[4] J. Srinivas, A. K. Das, M. Wazid, and N. Kumar, "Anonymous lightweight chaotic map-based authenticated key agreement protocol for industrial Internet of Things," *IEEE Trans. Depend. Secure Comput.*, vol. 17, no. 6, pp. 1133–1146, Nov. 2020.

[5] C.-C. Chang and H.-D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 357–366, Jan. 2015.

[6] S. Qiu, D. Wang, G. Xu, and S. Kumari, "Practical and provably secure three-factor authentication protocol based on extended chaotic-maps for mobile lightweight devices," *IEEE Trans. Depend. Secure Comput.*, vol. 19, no. 2, pp. 1338–1351, Mar. 2022.

[7] A. K. Das, M. Wazid, N. Kumar, A. V. Vasilakos, and J. J. P. C. Rodrigues, "Biometrics-based privacy-preserving user authentication scheme for cloud-based industrial Internet of Things deployment," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4900–4913, Dec. 2018.

[8] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, "Physical unclonable functions and applications: A tutorial," *Proc. IEEE*, vol. 102, no. 8, pp. 1126–1141, Aug. 2014.

[9] Y. Zheng, W. Liu, C. Gu, and C.-H. Chang, "PUF-based mutual authentication and key exchange protocol for peer-to-peer IoT applications," *IEEE Trans. Depend. Secure Comput.*, vol. 20, no. 4, pp. 1–18, Jul. 2022.

[10] H. Krawczyk and T. Rabin, "Chameleon signatures," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*. San Diego, CA, USA: The Internet Society, 2000.

[11] S. Guo, D. Zeng, and Y. Xiang, "Chameleon hashing for secure and privacy-preserving vehicular communications," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 11, pp. 2794–2803, Nov. 2014.

[12] M. Turkanović, B. Brumen, and M. Hölbl, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion," *Ad Hoc Netw.*, vol. 20, pp. 96–112, Sep. 2014.

[13] P. Porambage, C. Schmitt, P. Kumar, A. Gurtov, and M. Ylianttila, "Two-phase authentication protocol for wireless sensor networks in distributed IoT applications," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2014, pp. 2728–2733.

[14] W. Che, M. Martin, G. Pocklassery, V. Kajuluri, F. Saqib, and J. Plusquellic, "A privacy-preserving, mutual PUF-based authentication protocol," *Cryptography*, vol. 1, no. 1, p. 3, Nov. 2016.

[15] P. Gope, J. Lee, and T. Q. S. Quek, "Lightweight and practical anonymous authentication protocol for RFID systems using physically unclonable functions," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 11, pp. 2831–2843, Nov. 2018.

[16] P. Gope and B. Sikdar, "Privacy-aware authenticated key agreement scheme for secure smart grid communication," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3953–3962, Jul. 2019.

[17] Y. Zhang, R. H. Deng, E. Bertino, and D. Zheng, "Robust and universal seamless handover authentication in 5G HetNets," *IEEE Trans. Depend. Secure Comput.*, vol. 18, no. 2, pp. 858–874, Mar. 2021.

[18] F. Wang, J. Cui, Q. Zhang, D. He, C. Gu, and H. Zhong, "Blockchain-based lightweight message authentication for edge-assisted cross-domain industrial Internet of Things," *IEEE Trans. Depend. Secure Comput.*, vol. 21, no. 4, pp. 1587–1604, Jun. 2023.

[19] Y. Wu et al., "Attacks and countermeasures on privacy-preserving biometric authentication schemes," *IEEE Trans. Depend. Secure Comput.*, vol. 20, no. 2, pp. 1744–1755, Mar. 2023.

[20] H. Zhang, X. Li, S.-Y. Tan, M. J. Lee, and Z. Jin, "Privacy-preserving biometric authentication: Cryptanalysis and countermeasures," *IEEE Trans. Depend. Secure Comput.*, vol. 20, no. 6, pp. 5056–5069, Jan. 2023.

[21] F. Liu et al., "Lightweight batch authentication protocol for bus-NB-IoT hierarchical network in smart grid using physically unclonable function," in *Proc. IEEE 9th Int. Conf. Big Data Secur. Cloud (BigDataSecurity), IEEE Int. Conf. High Perform. Smart Comput. (HPSC), IEEE Int. Conf. Intell. Data Secur. (IDS)*, May 2023, pp. 107–114.

[22] B. Palaniswamy, S. Camtepe, E. Foo, and J. Pieprzyk, "An efficient authentication scheme for intra-vehicular controller area network," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3107–3122, 2020.

[23] P. Kumar, A. Braeken, A. Gurtov, J. Iinatti, and P. H. Ha, "Anonymous secure framework in connected smart home environments," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 4, pp. 968–979, Apr. 2017.

[24] M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo, "Secure remote user authenticated key establishment protocol for smart home environment," *IEEE Trans. Depend. Secure Comput.*, vol. 17, no. 2, pp. 391–406, Mar. 2020.

[25] C. Wang, D. Wang, Y. Duan, and X. Tao, "Secure and lightweight user authentication scheme for cloud-assisted Internet of Things," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 2961–2976, 2023.

[26] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Trans. Comput. Syst.*, vol. 8, no. 1, pp. 18–36, 1990.

[27] Q. Zhang, J. Wu, H. Zhong, D. He, and J. Cui, "Efficient anonymous authentication based on physically unclonable function in industrial Internet of Things," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 233–247, 2023.

[28] C. Boyd and W. Mao, "On a limitation of BAN logic," in *Advances in Cryptology—EUROCRYPT'93*, T. Helleseth, Ed., Berlin, Germany: Springer, 1994, pp. 240–247.

[29] X. Allamigeon B. Blanchet, V. Cheval, and B. Smyth. (2010). *ProVerif: Cryptographic Protocol Verifier in the Formal Model*. [Online]. Available: http://prosecco.gforge.inria.fr/personal/bblanche/proverif

[30] D. Wang, D. He, P. Wang, and C.-H. Chu, "Anonymous two-factor authentication in distributed systems: Certain goals are beyond attainment," *IEEE Trans. Depend. Secure Comput.*, vol. 12, no. 4, pp. 428–442, Jul. 2015.

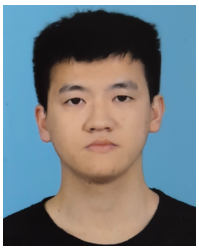
[31] M. Wazid, A. K. Das, V. Odelu, N. Kumar, M. Conti, and M. Jo, "Design of secure user authenticated key management protocol for generic IoT networks," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 269–282, Feb. 2018.

[32] D. Kumar and H. S. Grover, "Cryptanalysis of a secure and lightweight authentication protocol for wearable devices environment," in *Proc. 2nd Int. Conf. Inventive Syst. Control (ICISC)*, Jan. 2018, pp. 1063–1068.

[33] J. Srinivas, A. K. Das, N. Kumar, and J. J. P. C. Rodrigues, "Cloud centric authentication for wearable healthcare monitoring system," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 5, pp. 942–956, Sep. 2020.

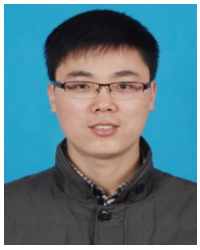
[34] S. Li, T. Zhang, B. Yu, and K. He, "A provably secure and practical PUF-based end-to-end mutual authentication and key exchange protocol for IoT," *IEEE Sensors J.*, vol. 21, no. 4, pp. 5487–5501, Oct. 2020.
[35] U. Chatterjee et al., "Building PUF based authentication and key exchange protocol for IoT without explicit CRPs in verifier database," *IEEE Trans. Depend. Secure Comput.*, vol. 16, no. 3, pp. 424–437, May 2019.
[36] M. Shuai, N. Yu, H. Wang, and L. Xiong, "Anonymous authentication scheme for smart home environment with provable security," *Comput. Secur.*, vol. 86, pp. 132–146, Sep. 2019.
[37] J. Srinivas, A. K. Das, M. Wazid, and A. V. Vasilakos, "Designing secure user authentication protocol for big data collection in IoT-based intelligent transportation system," *IEEE Internet Things J.*, vol. 8, no. 9, pp. 7727–7744, May 2021.
[38] X. Li, J. Niu, M. Z. A. Bhuiyan, F. Wu, M. Karuppiah, and S. Kumari, "A robust ECC-based provable secure authentication protocol with privacy preserving for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3599–3609, Aug. 2018.

Qingyang Zhang (Member, IEEE) was born in Anhui, China, in 1992. He received the B.Eng. and Ph.D. degrees in computer science from Anhui University in 2014 and 2021, respectively. He is currently an Associate Professor with the School of Computer Science and Technology, Anhui University. He has more than 30 scientific publications in reputable journals (e.g., PROCEEDINGS OF THE IEEE, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, and IEEE TRANSACTIONS ON COMPUTERS) and international conferences. His research interests include edge computing, computer systems, and security.



Xiaolong Zhou is currently a Research Student with the School of Computer Science and Technology, Anhui University. His research interests include the security of the Industrial Internet of Things.

Hong Zhong (Member, IEEE) was born in Anhui, China, in 1965. She received the Ph.D. degree in computer science from the University of Science and Technology of China in 2005. She is currently a Professor and the Ph.D. Supervisor of the School of Computer Science and Technology, Anhui University. She has more than 200 scientific publications in reputable journals (e.g., IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, and IEEE TRANSACTIONS ON BIG DATA), academic books, and international conferences. Her research interests include applied cryptography, the IoT security, vehicular ad hoc networks, cloud computing security, and software-defined networking (SDN).



Jie Cui (Senior Member, IEEE) was born in Henan, China, in 1980. He received the Ph.D. degree from the University of Science and Technology of China in 2012. He is currently a Professor and the Ph.D. Supervisor of the School of Computer Science and Technology, Anhui University. He has more than 150 scientific publications in reputable journals (e.g., IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, IEEE TRANSACTIONS ON CLOUD COMPUTING, and IEEE TRANSACTIONS ON MULTIMEDIA), academic books, and international conferences. His current research interests include applied cryptography, the IoT security, vehicular ad hoc networks, cloud computing security, and software-defined networking (SDN).

Jiaxin Li received the master's degree from Anhui University, where he is currently pursuing the Ph.D. degree. He holds the position of the Director of Government Affairs of H3C Information Security Technology Company Ltd. His research interests include the security of vehicular ad hoc networks, data security, and the security of the Industrial Internet of Things.

Debiao He (Member, IEEE) received the Ph.D. degree in applied mathematics from the School of Mathematics and Statistics, Wuhan University, Wuhan, China, in 2009. He is currently a Professor with the School of Cyber Science and Engineering, Wuhan University, and Shanghai Key Laboratory of Privacy Preserving Computation, MatrixElements Technologies, Shanghai, China. He has published more than 100 research papers in refereed international journals and conferences, such as IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, and *Usenix Security Symposium*. His work has been cited more than 10,000 times on Google Scholar. His main research interests include cryptography and information security, in particular, cryptographic protocols. He was a recipient of the 2018 IEEE Systems Journal Best Paper Award and the 2019 IET Information Security Best Paper Award. He is on the Editorial Board of several international journals, such as *Journal of Information Security and Applications*, *Frontiers of Computer Science*, and *Human-Centric Computing and Information Sciences*.