IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received October 14, 2020, accepted October 21, 2020, date of publication October 27, 2020, date of current version November 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3034219

A Physically Secure, Lightweight Three-Factor and Anonymous User Authentication Protocol for IoT

ZHENHUA LIU[1], CHANGBO GUO[2], AND BAOCANG WANG[3]

[1]School of Mathematics and Statistics, Xidian University, Xi'an 710071, China
[2]School of Cyber Engineering, Xidian University, Xi'an 710071, China
[3]State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China

Corresponding author: Changbo Guo (gchangbo95@gmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61807026, Grant U19B2021, Grant U1736111, and Grant 61972457; in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2019JM-198; in part by the Plan for Scientific Innovation Talent of Henan Province under Grant 184100510012; in part by the Program for Science and Technology Innovation Talents in the Universities of Henan Province under Grant 18HASTIT022; in part by the National Cryptography Development Fund under Grant MMJJ20180111; and in part by the Key Research and Development Program of Shaanxi under Grant 2020ZDLGY08-04.

ABSTRACT Internet of Things has remarkable effects in human's daily life. It is important for users and sensors to securely access data collected by low-cost sensors via Internet in real-time IoT applications. There exist many authentication protocols for guaranteeing secure communication between users and sensors. However, in some protocols, the privacy of unattended sensors subjected to capture node attacks cannot be guaranteed. Moreover, the sensors subjected to physical tampering attacks can still execute normally the authentication process. Besides, an authentication protocol should be lightweight due to the restricted computing power and storage of the sensors. The idea of designing a more secure and lightweight authentication protocol engender this article. The proposed protocol can provide the physical security through physically unclonable function (PUF), require no additional phase to update challenge-response pairs (CRPs), and store a single CRP for each sensor. At the same time, the proposed protocol utilizes three factors, such as personal biometrics, smartcard and password, to strengthen the security contrasting with two factors, and manipulates some basic cryptographic operations, including bitwise-exclusive-OR (XOR) and hash function, to achieve the lightweight performance. Moreover, both formal security analysis based on Real-Or-Random (ROR) and informal security analysis demonstrate the security of the proposed protocol. Compared with the existing related protocols, the proposed protocol has the advantage in terms of security, functionality and computation costs. Finally, a NS3 simulation on measuring various network performance parameters indicates that the proposed protocol is practical in IoT environment.

INDEX TERMS Internet of Things, key agreement, physical unclonable function, mutual authentication, NS3 simulation.

I. INTRODUCTION

The Internet of Things is established by these objects that are capable of perceiving the surrounding environment and interacting with other objects via network [1]. As an infrastructure, these interconnected and intelligent things play vital roles in many fields, for instance medical system and industry [2], which makes it possible for human to build an intelligent and efficient society.

A. MOTIVATIONS

Data collection, processing and transmission in IoT system have been becoming more and more frequently. While coming with great convenience, the Internet of Things are taking up many security challenges. In the past three years, 20% of organizations suffered at least one IoT attack according to Gartner report [3]. Furthermore, the security issues on IoT could threat the national security, which are not astounding words. The cyberattacks on Ukraine's electric grids [4] are such one example among recent increasing IoT attacking incidents around the world.

A generic communication model for IoT applications is shown in Fig.1. GateWay links up all the intelligent objects

The associate editor coordinating the review of this manuscript and approving it for publication was Chien-Ming Chen.

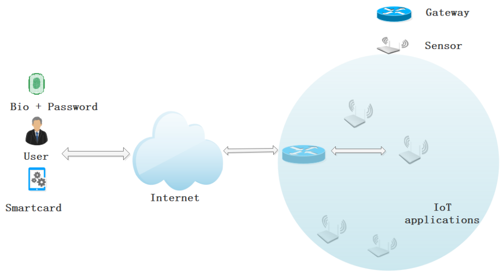


FIGURE 1. A generic communication model for IoT applications.

in a region by wireless or wired network, so that intelligent devices and users from other areas can communicate with each other through the Internet. In general, the security requirements of the IoT applications necessitate the confidentiality, integrity and authentication for users and data. Particularly, the authentication mechanism is indispensable to IoT [5].

On the one hand, some challenges come from the nature of Internet that IoT are based on, where the transmitted data between users and devices can not be well protected. These challenges are intensified by the characteristic that IoT devices are resource-limited, which invalidates some traditional authentication solutions.

On the other hand, it is critical for the network well-functioning to guarantee that the devices participating in the IoT network are trusted since a single compromised node could give rise to some security matters even undermine the whole system [6]. Since these sensors have limited power and are deployed in open and public places without being physically well-protecting, some adversaries can capture these devices easily to extract credentials from memory of captured sensors and launch sensors tempering attacks. Physical unclonable functions, which are generated by introducing variability into fabrication process of integrated circuits (IC) and make the IC unique, can be used as hardware security primitives to address these issues. Physical tampering attacks on the nodes with PUF can be detected since the behavior of distorting PUF will change the internal characteristics of sensor nodes with PUF and transform them into the unauthenticated nodes. When inputting a challenge, PUF can return a response. Both the input parameters and the return result are a string of bits and constitute a challenge-response pair (CRP). Some protocols [7]–[9] with PUF need to store a set of CRPs for each sensor node, which increases the storage space complexity of GateWay [10]. CRPs stored in GateWay for authentication are finite and consumed in every authentication phase, and thus CRPs will eventually get exhausted. Banerjee *et al.* [9] updated CRPs by executing an additional phase, which increases the communication cost. Gope and Sikdar [7] updated CRPs after every successful authentication, however their protocol was designed to authenticate between sensors and server.

Furthermore, since the adversaries could collect additional information (e.g., location, IP address) from users' and senosors' identity, it is essential for IoT authentication protocols to possess user/sensor anonymity and untraceability. Li *et al.* [11] declared that their protocol can obtain untraceability and user anonymity by using of ECC, which is not lightweight. Some authentication protocols supplying dynamic pseudonym identity [12], [15] based on hash and XOR operations were proposed to guarantee anonymity for user, but these protocols were not secure against the tracking attack.

Besides, due to limited battery life of IoT devices, it is significant to design lightweight authentication mechanism. Many lightweight protocols [16], [17] were proposed by using of bitwise XOR and hash operations. Additionally, compared with two-factor authentication protocols, three-factor authentication protocol using users' smart cards, passwords and biometrics can provide strong security against some attacks, such as stolen-smartcard attacks and password-guessing attacks.

B. CONTRIBUTIONS

Our contributions are given as following.

- The proposed protocol utilizes three factors: smart cards, passwords and biometrics for authentication, and employs hash algorithm, XOR operations and physically unclonable function to achieve lightweight and physical security. Different challenge-response pairs are used since CRPs would be updated after successfully executing the protocol. Moreover, additional procedure of updating challenge-response pairs is not required. Compared with other PUF-based protocols, the GateWay stores a single CRP for every sensor instead of storing a set of CRPs for every sensor.
- A formal security analysis on the basis of real-or-random (ROR) model is given. Besides, a security analysis is presented to indicate that the proposed protocol can prevent stolen *GWN*/user impersonation attack, smart card attack, physical attack, etc.
- Detailed comparisons of performance and functionality with the existing protocols are presented. Furthermore, these comparisons show that the proposed protocol requires less communication overhead and provides much more security.
- By using of NS3 [18] simulation tool, the simulation result of low end-to-end delay (EED) and high packet delivery rate (PDR) demonstrates that the proposed protocol is practical and suitable for IoT environment.

II. RELATED WORKS

According to Das *et al.*'s summary [19], the authenticated protocols for IoT should be against several common attacks, for instance, replay attack, man-in-the-middle attack, impersonation attack, stolen/lost smart card attack, online/offline password guessing attack, privileged-insider attack, sensors capture attack, and physical capture/tempering attack. Furthermore, these protocols should achieve user/sensor untraceability and anonymity and also provide friendly password/biometrics change.

Anonymity is a significant property of authentication protocol for IoT. Anonymity mainly contains: (1) User's/sensor's real identity could be revealed by adversaries; and (2) User/sensor untraceability ensures the adversary could not distinguish whether two communications from the same (unknown) partners [20]. Recently, different authentication protocols [13], [14] have been proposed to provide user anonymity and untraceability for various environment such as 5G network and fog computing environment. Xue *et al.* [12] and Fan *et al.* [15] claimed that their dynamic pseudonym identity-based authentication protocols could achieve user anonymity by only involving hash and XOR operations. However, according to Wang and Wang [20], the collusion between dishonest master nodes and malicious privileged users could breach the untraceability of any legitimate user among protocol [15]. At the same time, the identity of users could be offline guessed using eavesdropped information of a single execution of the protocol [12].

Several two-factor authentication protocols have been presented for IoT environment. Turkanović *et al.* [21] designed a lightweight two-factor authentication protocol for wireless sensors networks(WSNs). In their scheme, a user connects to a destination node directly to negotiate a session key. Chang and Le [22] and Farash *et al.* [23] specified that Turkanović *et al.*'s protocol [21] was susceptible to various type of attacks(e.g. stolen smart card attack, impersonation attack with sensor node capture, user traceability, etc.) and then proposed individually a new one to tackle those security shortcomings and vulnerabilities. Furthermore, Amin *et al.* [24] indicated that Farash *et al.*'s protocol [23] can not resist offline password guessing, user impersonation and stolen smart card attack, and cannot offer user anonymity, and then presented an improved three-factor protocol by using of password, smartcard, and biometrics. Li *et al.* [11] indicated Chang *et al.*'s protocol [22] could not offer mutual authentication and was vulnerable to the tracing attacks. To remove the disadvantages of Chang *et al.*'s protocol [22], Li *et al.* also presented an improved three-factor based solution [11].

Since the authentication protocols are executed on the resource-constrained sensors, the cryptographic primitive operations should be lightweight to protect the communication security. Das [25] observed that TinyPK protocol [26] was subjected to the "disguise as legitimate node" attack and presented an improved password-based solution using timestamps. Huang *et al.* [27] found some defects in Das *et al.*'s protocol and provided an improved solution. Meanwhile, Khan and Alghathbar [28] indicated that in Das *et al.*'s protocol users cannot change/update their passwords, and their protocol was suffered from gateway bypassing attack and privileged-insider attack and could not achieve mutual authentication. Thus, Vaidya *et al.* [29] presented an amended two-factor based solution, which was not secure against some common kinds of attacks. Afterwards, Das *et al.* [17] presented a new password-based user authentication protocol for large-scale hierarchical WSNs. Due to the use of hash and XOR, Das *et al.*'s protocol was lightweight and highly appropriate for WSNs. Moreover, Das *et al.*'s protocol had advantage of dynamically changing the user's password offline and could add nodes dynamically after initial nodes deployment phase. However, Turkanović and Hölbl [30] indicated that Das *et al.*'s protocol [17] was infeasible for real-world deployment and proposed an improved one. A signature-based protocol by utilizing elliptic curve cryptography (ECC) operation was presented by Challa *et al.* [31]. Their protocol provided better security compared with [21], [32], but required more computation costs.

Some authentication protocols based on PUF for WSNs and radio-frequency identification systems [33]–[37] had been proposed. Recently, Aman *et al.* [38] proposed a lightweight mutual device-to-device authentication protocol based on PUF for IoT system. Two sensor nodes can negotiate a session key with the help of a server without storing secret in memory by using of Aman *et al.*'s protocol. Garg *et al.* [8] designed a solution for mutual device-to-device authentication by using of PUF and ECC. Gope and Sikdar [7] pointed out that the noise problem during the PUF's operation should be considered, but both [38] and [8] cannot support noisy PUF environment. Banerjee *et al.* [9] presented a lightweight anonymous protocol based on PUF, which executed two extra phases to update pseduo-identity of user and challenge-response pairs of sensor node.

In summary, most of user authentication protocols cannot achieve security requirements for IoT environment. To solve these issues, we concentrate on constructing a new lightweight and three-factor user authentication protocol, which can guarantee anonymity and physical security for IoT system.

III. PRELIMINARIES

The required backgrounds are given below, including hash function, fuzzy extractor, physically unclonable function, etc.

A. HASH FUNCTION

The input parameter of hash function is an arbitrary length string and its output is a fixed length value. Furthermore, it is hard for a collusion-resistant hash function to seek two strings that would produce the same output, which is suitable for data integrity. The definition of hash function is given below [39], [40].

Definition 1: The advantage of an adversary $\mathcal{A}$ in searching hash collision is

$$Adv_{\mathcal{A}(t)}^{Hash} = \Pr[(a, b) \leftarrow_R \mathcal{A} : a \neq b \text{ and } h(a) = h(b)],$$

where $\Pr[X]$ *denotes the probability of an event* X*, and* $(a, b) \leftarrow_R \mathcal{A}$ *denotes that the pair* (a, b) *are randomly picked by* $\mathcal{A}$*. By an* (ϵ, t)*-adversary* $\mathcal{A}$ *attacking the collision* *resistance of* $h(\cdot)$*, it means that the runtime of* $\mathcal{A}$ *is at most* t *and* $Adv_{\mathcal{A}(t)}^{Hash} \leq \epsilon$.

B. FUZZY EXTRACTOR

Fuzzy Extractor [41] can be often used for biometric verification. Even if the biometric input changes slightly, Fuzzy Extractor can extract nearly the same uniform randomness R.

This techniques can be applied to not only biometric information, but also any keying material. A Fuzzy Extractor [7], [42]–[44] consists of two parts: probabilistic generation function *Gen(·)* and deterministic reproduction function *Rep(·)*.

- *Gen(·)*: The input of the function is a string R and the output are a key K and a helper data hd, i.e., $(K, hd) = Gen(R)$.
- *Rep(·)*: Given a helper data hd, this function can refactor the correlative key K generated by $Gen(\cdot)$ using a string R' that the Hamming distance between R' and R is at most d.

C. PHYSICALLY UNCLONABLE FUNCTION

PUF is based on the idea that even though ICs are produced by the same manufacturing process, they are actually slightly different due to normal manufacturing variability [45]. PUF takes a challenge as input and returns a response, i.e., $R = PUF(C)$. A challenge-response pair (CRP) is composed of a challenge and its response. PUF has the characteristics as follows:

- A PUF can give the same R for the same C, even if the same challenge is used multiple times.
- If the same challenge is passed to different PUF, each PUF will produce completely different response with highly probability.

D. INDISTINGUISHABLITY UNDER CHOSEN-PLAINTEXT ATTACK (IND-CPA)

The definition of IND-CPA is as follows [46]. Suppose that the single or multiple eavesdropper is represented by SE/ME, and $RO_{k_1}, RO_{k_2}, \cdots, RO_{k_N}$ are N different independent encryption random oracles related with encryption keys $k_1, k_2, \cdots, k_N$, respectively, where k_N is a security parameter.

Definition 2: Let $Adv_{\Omega,SE}^{IND-CPA}(k)$ *and* $Adv_{\Omega,ME}^{IND-CPA}(k)$ *be the advantage functions of SE and ME as follows.*

$$\begin{aligned}
Adv_{\Omega,SE}^{IND-CPA}(k) = \big|2\Pr[&SE \leftarrow RO_{k_1}; (b_0, b_1 \leftarrow_R SE);\\
&\alpha \leftarrow_R \{0,1\}; \beta \leftarrow_R RO_{k_1}(b_\alpha):\\
&SE(\beta) = \alpha] - 1\big|,\\
Adv_{\Omega,ME}^{IND-CPA}(k) = \big|2\Pr[&ME \leftarrow RO_{k_1}, RO_{k_2}, \cdots, RO_{k_N};\\
&(b_0, b_1 \leftarrow_R ME); \alpha \leftarrow_R \{0,1\};\\
&\beta_1 \leftarrow_R RO_{k_1}(b_\alpha), \cdots,\\
&\beta_N \leftarrow_R RO_{k_N}(b_\alpha):\\
&ME(\beta_1, \cdots, \beta_N) = \alpha] - 1\big|,
\end{aligned}$$

where Ω *is an encryption scheme.* Ω *is IND-CPA secure in the single or multiple eavesdropper setting if* $Adv_{\Omega,SE}^{IND-CPA}(k)$ $(Adv_{\Omega,ME}^{IND-CPA}(k))$ *is negligible (in* k*) for any probabilistic polynomial time SE(ME).*

E. NETWORK AND THREAT MODELS

Network Model: The network model demonstrates how the roles in the system communicate with each other [31]. As shown in Fig.1, all IoT sensor nodes can be linked to the Internet through the trusted gateway. Users and sensor nodes need to register with their corresponding gateway nodes. A registered user can mutually authenticate with a registered sensor node via gateway to access sensor nodes data using a negotiated session key.

Threat Model: Das *et al.* [19] discussed that the threat models were considered in the secure protocols for the IoT environment. Threat models defined the capabilities of an attacker $\mathcal{A}$ as follows.

- Under Dolev-Yao (DY) threat model [47], $\mathcal{A}$ can tamper with the data transmitted through the communication channel by intercepting, modifying, deleting the exchanged messages or creating new messages. Moreover, neither the user nor the sensor node is a reliable participant.
- CK-adversary model is the standard model currently used in key exchange protocols [48], [49]. Under CK-adversary model, besides delivering the messages, $\mathcal{A}$ can compromise the secret credentials including session keys. Thus, it is essential for the authenticated key exchange protocols to insure that adversaries cannot obtain the secret credential of other entities [50] from leakaged session short-term secrets or session key. The proposed protocol will consider the CK-adversary model.

Furthermore, $\mathcal{A}$ can capture sensors, perform physically tampering on sensors, and gain sensitive information stored in physically sensors since some sensors are placed in some unattended environment. Additionally, $\mathcal{A}$ can extract some secret information in lost or stolen smart cards.

IV. PROPOSED AUTHENTICATION KEY AGREEMENT PROTOCOL

The proposed protocol are described in this section. Tab.1 lists the notations used in the following contents. The proposed protocol can be divided into four phases: 1) *GWN* initialization, 2) sensor node registration, 3) user registration, 4) login and user authentication phase.

A. GWN INITIALIZATION PHASE

GWN generates a long-term key *LTK* and publishes the hash function $h(\cdot)$, symmetric encryption and decryption algorithms $E_k[\cdot]$ and $D_k[\cdot]$.

B. SENSOR NODE REGISTRATION

Sensor nodes are enrolled into the system over a secure channel. Fig.2 demonstrates this phase.

TABLE 1. Notations.

Notation	Description
U_i, ID_i, SC_i	The user U_i's identity and its smart card
Bio_i, PW_i	The user U_i's personal biometrics and password
GWN, LTK	The gateway node, its long term key
S_{GWN-ID_i}	U_i's secret key after registering in GWN
$S_{GWN-SID_j}$	S_i's secret key after registering in GWN
S_j, SID_j	The sensor node and its identity
$E_K[\cdot], D_K[\cdot]$	Applying symmetric encryption/decryption using key K
$Gen(\cdot), Rep(\cdot)$	Fuzzy extractor probabilistic generation and deterministic reproduction functions, respectively
σ_i, τ_i	corresponding to the output of the key K and the helper data hd after inputting biometric Bio_i to $Gen(\cdot)$,$(\sigma_i, \tau_i) = Gen(Bio_i)$
$\oplus, h(\cdot), \|\|$	The XOR, hash, and concatenation operation
$\mathcal{A}$	An attacker

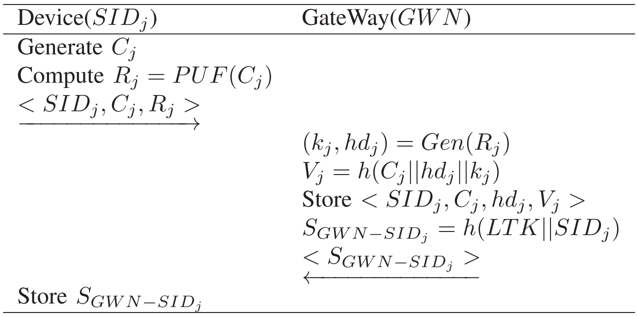

Device(SID_j)
GateWay(GWN)
Generate C_j
Compute $R_j = PUF(C_j)$
$\xrightarrow{< SID_j, C_j, R_j >}$
$(k_j, hd_j) = Gen(R_j)$
$V_j = h(C_j||hd_j||k_j)$
Store $< SID_j, C_j, hd_j, V_j >$
$S_{GWN-SID_j} = h(LTK||SID_j)$
$\xleftarrow{< S_{GWN-SID_j} >}$
Store $S_{GWN-SID_j}$

FIGURE 2. Sensor node registration.

- Step 1. When a sensor node S_j registers with GWN, S_j generates a random response C_j, computes $R_j = PUF(C_j)$, and sends $< SID_j, C_j, R_j >$ to GWN.
- Step 2. Upon getting the request of sensor node S_j, GWN computes $(k_j, hd_j) = Gen(R_j)$, $S_{GWN-SID_j} = h(LTK||SID_j)$, $V_j = h(C_j||hd_j||k_j)$, stores $< SID_j, C_j, hd_j, V_j >$ in its database, and sends $S_{GWN-SID_j}$ to S_j. Since only legitimate sensors could calculate the right k_j from C_j and only legitimate sensors could calculate the right V_j from k_j, V_j is used to verify sensors in Sec IV-D Step 3.

C. USER REGISTRATION

The following steps should be performed for users to acquire the services from registered sensor nodes. This phase is demonstrated in Fig.3.

- Step 1. This step is carried out in a secure channel, where user U_i sends his/her identity ID_i as a registration request to gateway node GWN.
- Step 2. On obtaining the registration request, GWN examines whether ID_i is in the database. If yes, U_i is a registered user. Otherwise, GWN computes $S_{GWN-ID_i} = h(LTK||ID_i)$, generates randomly x, computes $DID_i = E_{LTK}[ID_i||x]$, and sends $< DID_i, S_{GWN-ID_i} >$ to U_i.

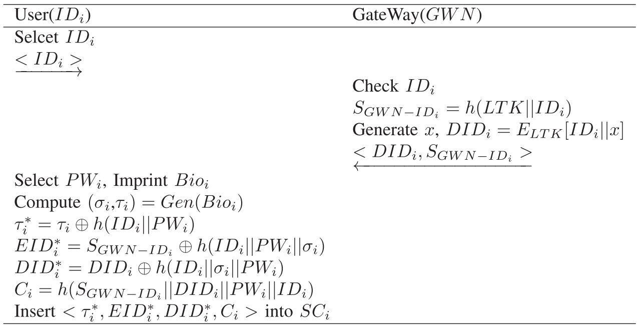

User(ID_i)
GateWay(GWN)
Selcet ID_i
$\xrightarrow{< ID_i >}$
Check ID_i
$S_{GWN-ID_i} = h(LTK||ID_i)$
Generate x, $DID_i = E_{LTK}[ID_i||x]$
$\xleftarrow{< DID_i, S_{GWN-ID_i} >}$
Select PW_i, Imprint Bio_i
Compute $(\sigma_i,\tau_i) = Gen(Bio_i)$
$\tau_i^* = \tau_i \oplus h(ID_i||PW_i)$
$EID_i^* = S_{GWN-ID_i} \oplus h(ID_i||PW_i||\sigma_i)$
$DID_i^* = DID_i \oplus h(ID_i||\sigma_i||PW_i)$
$C_i = h(S_{GWN-ID_i}||DID_i||PW_i||ID_i)$
Insert $< \tau_i^*, EID_i^*, DID_i^*, C_i >$ into SC_i

FIGURE 3. User registration.

- Step 3. When obtaining these parameters, U_i chooses a password PW_i and imprints his/her biometrics BIO_i. Afterwards, SC_i calculates $(\sigma_i, \tau_i) = Gen(Bio_i)$, $\tau_i^* = \tau_i \oplus h(ID_i||PW_i)$, $EID_i^* = S_{GWN-ID_i} \oplus h(ID_i||PW_i||\sigma_i)$, $DID_i^* = DID_i^* \oplus h(ID_i||\sigma_i||PW_i)$, $C_i = h(S_{GWN-ID_i}||DID_i|| PW_i||ID_i)$, and saves $< \tau_i^*, DID_i^*, EID_i^*, C_i >$ into SC_i.

D. LOGIN AND USER AUTHENTICATION PHASE

A registered U_i must accomplish the login and user authentication phase before accessing an enrolled device. After authenticating mutually, U_i, S_j and GWN compute a same session key. Fig.4 shows this phase.

- Step 1. A user U_i inputs her/his identity ID_i, password PW_i and biometrics Bio_i. SC_i of U_i calculates $\tau_i = \tau_i^* \oplus h(ID_i||PW_i)$, $\sigma_i = Rep(Bio_i, \tau_i)$, $S_{GWN-ID_i} = EID_i^* \oplus h(ID_i||PW_i||\sigma_i)$, $DID_i = DID_i^* \oplus h(ID_i||\sigma_i||PW_i)$ and examines $C_i' \overset{?}{=} C_i$. If the calculated C_i' is not equal to C_i stored in SC_i, it means that at least one of the three factors offered by U_i is incorrect, and then SC_i aborts the login phase. Otherwise, SC_i continues.
- Step 2. U_i gets the current timestamp T_1, computes $K_{ug} = h(S_{GWN-ID_i}||DID_i)$ and $M_1 = E_{K_{ug}}[SID_j||T_1]$, and transmits $< DID_i, M_1, T_1 >$ to GWN.
- Step 3. On getting the login request, GWN checks the timeliness of the received timestamp T_1 with the condition $|T_1^* - T_1| \le \Delta T$, where the received time of the message is T_1^* and the maximum of the allowable transmission delay is ΔT. If it holds, GWN will extract ID_i by calculating $ID_i||x = D_{LTK}[DID_1]$ with the long term key LKT. Then, GWN computes $S_{GWN-ID_i} = h(LTK||ID_i)$, which is the shared key between GWN and U_i, $K_{ug} = h(S_{GWN-SID_i}||DID_i)$ and $SID_j||T_1 = D_{K_{ug}}[M_1]$. If the timestamp in M_1 is not effective, GWN aborts. Otherwise, GWN lookups $< SID_j, C_j, hd_j, V_j >$ in the database, generates a nonce number r_g^*, and gets the current timestamp T_2. Then, GWN calculates $r_g = h(LTK||r_g^*)$, $S_{GWN-SID_j} = h(LTK||SID_j)$, $M_2 = h(S_{GWN-SID_j}||C_j) \oplus hd_j$, $K_{gs} = h(V_j||S_{GWN-SID_j})$ and $M_3 = E_{K_{gs}}[ID_i||r_g||T_2]$. At last, GWN sends $< C_j, M_2, M_3, T_2 >$ to S_j.

User(ID_i)	GateWay(GWN)	Sensor(SID_j)
Inputs ID_i, PW_i and Bio_i $\tau_i = \tau_i^* \oplus h(ID_i\|\|PW_i)$ $\sigma_i = Rep(Bio_i, \tau_i)$ $S_{GWN-ID_i} = EID_i^* \oplus h(ID_i\|\|PW_i\|\|\sigma_i)$ $DID_i = DID_i^* \oplus h(ID_i\|\|\sigma_i\|\|PW_i)$ $C_i' = h(S_{GWN-ID_i}\|\|DID_i\|\|PW_i\|\|ID_i)$ Check $C_i' \overset{?}{=} C_i$ $K_{ug} = h(S_{GWN-ID_i}\|\|DID_i)$ $M_1 = E_{K_{ug}}[SID_j\|\|T_1]$ $Msg_1 =< DID_i, M_1, T_1 >$ to GWN $\xrightarrow{\qquad\qquad}$		
	Check T_1 $ID_i\|\|x = D_{LTK}[DID_i]$ $S_{GWN-ID_i} = h(LTK\|\|ID_i)$ $K_{ug} = h(S_{GWN-SID_i}\|\|DID_i)$ $SID_j\|\|T_1 = D_{K_{ug}}[M_1]$ Lookup $< SID_j, C_j, hd_j, V_j >$ Generate number $r_g^*, r_g = h(LTK\|\|r_g^*)$ $S_{GWN-SID_j} = h(LTK\|\|SID_j)$ $M_2 = h(S_{GWN-SID_j}\|\|C_j) \oplus hd_j$ $K_{gs} = h(V_j\|\|S_{GWN-SID_j})$ $M_3 = E_{K_{gs}}[ID_i\|\|r_g\|\|T_2]$ $Msg_2 =< C_j, M_2, M_3, T_2 >$ to SID_j $\xrightarrow{\qquad\qquad}$	
		Check T_2 $hd_j = h(S_{GWN-SID_j}\|\|C_j) \oplus M_2$ $k_j = Rep(PUF(C_j), hd_j)$ $V_j = h(C_j\|\|hd_j\|\|k_j)$ $K_{gs} = h(V_j\|\|S_{GWN-SID_j})$ $ID_i\|\|r_g\|\|T_2 = D_{K_{gs}}[M_3]$ Generate r_j, T_3 $C_j^{new} = h(C_j\|\|r_g), R_j^{new} = PUF(C_j^{new})$ $SK = h(ID_i\|\|SID_j\|\|r_g\|\|r_j)$ $M_4 = h(SK\|\|r_j\|\|r_g\|\|T_3)$ $M_5 = E_{K_{gs}}[R_j^{new}\|\|r_j\|\|M_4\|\|T_3]$ $Msg_3 =< M_5, T_3 >$ to GWN $\xleftarrow{\qquad\qquad}$
	Check T_3 $R_j^{new}\|\|r_j\|\|M_4\|\|T_3 = D_{K_{gs}}[M_5]$ $SK = h(ID_i\|\|SID_j\|\|r_g\|\|r_j)$ $M_4 \overset{?}{=} h(SK\|\|r_j\|\|r_g\|\|T_3)$ $C_j^{new} = h(C_j\|\|r_g)$ $(k_j^{new}, hd_j^{new}) = Gen(R_j^{new})$ $V_j^{new} = h(C_j^{new}\|\|hd_j^{new}\|\|k_j^{new})$ Replace $< SID_j, C_j^{new}, hd_j^{new}, V_j^{new} >$ $DID_i^{new} = E_{LTK}[ID_i\|\|r'_g]$ $M_6 = h(SK\|\|r_j\|\|r_g\|\|T_4)$ $M_7 = E_{K_{ug}}[DID_i^{new}\|\|r_j\|\|r_g\|\|M_6\|\|T_4]$ $Msg_4 =< M_7, T_4 >$ to ID_i $\xleftarrow{\qquad\qquad}$	
Check T_4 $DID_i^{new}\|\|r_j\|\|r_g\|\|M_6\|\|T_4 = D_{K_{ug}}[M_7]$ $SK = h(ID_i\|\|SID_j\|\|r_g\|\|r_j)$ Check $M_6 \overset{?}{=} h(SK\|\|r_j\|\|r_g\|\|T_4)$ Replace $DID_i^* = DID_i^{new} \oplus h(ID_i\|\|\sigma_i\|\|PW_i)$		

FIGURE 4. Login and authentication.

- Step 4. When getting GWN's message, S_j examines the freshness of T_2. S_j calculates $hd_j = h(S_{GWN-SID_j}||C_j) \oplus M_2$. Subsequently, S_j computes $k_j = Rep(PUF(C_j), hd_j)$ by using of physically unclonable function and fuzzy extractor. Then, S_j computes $V_j = h(C_j||hd_j||k_j)$, $K_{gs} = h(V_j||S_{GWN-SID_j})$ and $ID_i||r_g||T_2 = D_{K_{gs}}[M_3]$. If the timestamp in M_3 is invalid, S_j aborts. Otherwise, S_j produces a nonce number r_j and the current timestamp T_3, and calculates a new challenge $C_j^{new} = h(C_j||r_g)$, a new corresponding response $R_j^{new} = PUF(C_j^{new})$, a session key $SK = h(ID_i||SID_j||r_g||r_j)$, $M_4 = h(SK||r_j||r_g||T_3)$ and $M_5 = E_{K_{gs}}[R_j^{new}||r_j||M_4||T_3]$. Then, S_j submits $< M_5, T_3 >$ to GWN.
- Step 5. On getting S_j's message, GWN examines the freshness of T_3. If it holds, GWN decrypts $R_j^{new}||r_j||M_4||T_3 = D_{K_{gs}}[M_5]$. After verifying the correctness of the timestamp of M_5, GWN calculates a session key $SK = h(ID_i||SID_j||r_g||r_j)$ and checks $M_4 \overset{?}{=} h(SK||r_j||r_g||T_3)$. If the equation holds, GWN calculates $C_j^{new} = h(C_j||r_g)$, $(k_j^{new}, hd_j^{new}) = Gen(R_j^{new})$, $V_j^{new} = h(C_j^{new}||hd_j^{new}||k_j^{new})$, and replaces the previous record with the new one $< SID_j, C_j^{new}, hd_j^{new}, V_j^{new} >$.

Then, GWN computes $DID_i^{new} = E_{LTK}[ID_i||r'_g]$ as a new dynamic identity of ID_i, $M_6 = h(SK||r_j||r_g||T_4)$ and $M_7 = E_{K_{ug}}[DID_i^{new}||r_j||r_g||M_6||T_4]$, and sends $< M_7, T_4 >$ to U_i.

- Step 6. When getting the message of GWN, U_i examines the freshness of T_4. If it holds, U_i calculates $DID_i^{new}||r_j||r_g||M_6||T_4 = D_{K_{ug}}[M_7]$. After checking the freshness of the timestamp in M_7, U_i computes a session key $SK = h(ID_i||SID_j||r_g||r_j)$ and checks $M_6 \overset{?}{=} h(SK||r_j||r_g||T_4)$. If the equation holds, U_i calculates $DID_i^* = DID_i^{new} \oplus h(ID_i||\sigma_i||PW_i)$ and replaces the previous one.

E. USER PASSWORD AND BIOMETRIC CHANGE PHASE

Before a registered user U_i updates his/her password and/or biometric, he/she should accomplish the phase in Sec.IV-D Step.1. Suppose PW_i^{new} and Bio_i^{new} are the password and biometric U_i wants to update. Then SC_i computes $(\sigma_i^{new}, \tau_i^{new}) = Gen(Bio_i^{new})$, $\tau_i^{**} = \tau_i^{new} \oplus h(ID_i||PW_i^{new})$, $EID_i^{new} = S_{GWN-ID_i} \oplus h(ID_i||PW_i^{new}||\sigma_i^{new})$, $DID_i^{**} = DID_i \oplus h(ID_i||\sigma_i^{new} \; ||PW_i^{new})$, $C_i^{new} = h(S_{GWN-ID_i}||DID_i||PW_i^{new}||ID_i)$. SC_i replaces $< \tau_i^*, DID_i^*, EID_i^*, C_i >$ with new calculated results $< \tau_i^{new}, DID_i^{**}, EID_i^{new}, C_i^{new} >$.

V. SECURITY ANALYSIS

A. FORMAL SECURITY ANALYSIS BASED ON REAL-OR-RANDOM MODEL

In this subsection, formal security analysis are given from Real-Or-Random (ROR) model [51].

ROR model: There are three communication participants including GWN, S_j, U_i in this protocol. The model considers the followings [16].

Participants. $\pi_{U_i}^u$, π_{GWN}^g and $\pi_{S_j}^s$ are the oracles of u, g and s associated with U_i, GWN and S_j.

Accepted state. Let π^ω be one of instances of participants $\pi_{U_i}^u, \pi_{GWN}^g$ and $\pi_{S_j}^s$. When upon receiving the final expected protocol message, π^ω comes into an accept state. If all communications that π^ω involves including sending and receiving messages are arranged in succession, which are distinguished with the session identification sid of π^ω.

Partnering. Two instances π^{ω_1} and π^{ω_2} are partners if the following occurs: 1) π^{ω_1} and π^{ω_2} will be in the accept state; 2) π^{ω_1} and π^{ω_2} will share the same sid after authenticating mutually successfully; and 3) π^{ω_1} and π^{ω_2} will also be mutual partners.

Freshness. $\pi_{U_i}^u$ and $\pi_{S_j}^s$ are in the state of freshness if adversary $\mathcal{A}$ does not acquire the session key SK between U_i and S_j.

According to the ROR model that coincides with the DY threat model [47], all the transmission messages in a public channel can be completely controlled by the adversaries. What $\mathcal{A}$ can do are eavesdropping, altering, deleting, and even inserting fabricated messages during communication. Additionally, the following queries [16], [40], [52] can be performed by the adversary.

- *Execute*(π^u, π^g, π^s): By executing the query, $\mathcal{A}$ can monitor all the exchanged messages among U_i, GWN and S_j. An eavesdropping attack is modeled in this query, where monitored participants can not discover that they are under this attack.
- *Send*(π^s, m): This query model the ability of adversary $\mathcal{A}$ to deliver a message m to its participant π^s. If the message m is constructed meticulously, participant π^s regard $\mathcal{A}$ as a legitimate participant and return a legitimate message. An active attack is modeled in this query.
- *Reveal*(π^s): π^s and its partner will share same session key SK after authenticating successfully. $\mathcal{A}$ can obtain session key SK by executing this query.
- *CorruptSC*($\pi_{U_i}^u$): U_i stores its credentials $< \tau_i^*, EID_i^*, DID_i^*, C_i >$ in smart card SC_i. This query models the case that $\mathcal{A}$ can reveal the credential information from the stolen and lost smart card by side channel attack.
- *CorruptSD*(π^s): Sensor node S_j stores the credentials $< SID_j, S_{GWN-SID_j} >$. This query models the case that $\mathcal{A}$ can retrieve the credentials $< SID_j, S_{GWN-SID_j} >$ from a captured sensor node S_j. The weak corruption model, where instance's short-term keys are not corrupted, can be supported by both the queries *CorruptSC* and *CorruptSD* according to survey of Chang and Le [22].
- *Test*(π^s): The semantic security, which follows indistinguishability in the ROR model, of the session key SK established by GWN, U_i and S_j following the indistinguishability in the ROR model is decided by this query. After $\mathcal{A}$ executes this query, π^s returns a session key or a random key according to the result of flipping an unbiased coin c. If $c = 1$ and SK is fresh, which means that *Reveal*(π^s) is not requested by $\mathcal{A}$, the outcome is SK. Otherwise, the outcome is a random key. If SK is not fresh, π^s returns a null value.

Semantic security of the session key: In the ROR model, the goal of $\mathcal{A}$ is to distinguish an instance's real SK from a random key. $\mathcal{A}$ can perform many *Test*($\cdot$) queries to π^u or π^s. After performing many *Test*($\cdot$) queries to π^u or π^s, $\mathcal{A}$ guesses a bit c'. If $c' = c$, $\mathcal{A}$ wins the game. Suppose that the probabilistic for $\mathcal{A}$ to win the game is $|Pr[SUC]|$ [9]. The advantage of $\mathcal{A}$ in breaking the semantic security of the proposed authenticated key agreement (AKE) protocol, called P, in time t is defined as $Adv_{P,\mathcal{A}}^{AKE}(t) = |2\Pr[SUC] - 1|$.

Random Oracle: $\mathcal{A}$ can query $PUF(\cdot)$ and $h(\cdot)$, which are random oracles, written $\mathcal{HO}$.

Security Proof: The semantic security of the proposed protocol can be proved by the following theorem according to PUF and collision-resistant hash function, and password obey Zipf's law [53] and the above described ROR model.

Theorem 1: If $\mathcal{A}$ is a polynomial-time adversary running against the proposed protocol P under the ROR model, *which uses the Zipf's law for user-chosen passwords, l_1 and l_2 denote the number of bits in the biometric secret key σ_i and the secret user identity ID_i, respectively. $Adv_{P,\mathcal{A}}^{AKE}$ denotes $\mathcal{A}$'s advantage in breaking P's semantic security in order to derive the session key between a legal registered user U and an accessed sensor node, then*

$$\begin{aligned} Adv_{P,\mathcal{A}}^{AKE}(t) \leq & \frac{q_h^2}{|Hash|} + \frac{q_p^2}{|PUF|} \\ & + 2(max\{C' \cdot q_s^{s'}, \frac{q_s}{2^{l_1}}, \frac{q_2}{2^{l_2}}\} \\ & + Adv_{\Omega}^{IND-CPA}(k)), \end{aligned}$$

where q_h, q_p and q_s are the number of hash, PUF and Send *queries, $|Hash|$ and $|PUF|$ denote the range space of $h(\cdot)$ and $PUF(\cdot)$, and C' and s' are the Zipf's parameters.*

Proof: The following proof is similar to those proofs presented in [9], [16], [24]. There are six games, written $G_i (i = 0, 1, \cdots, 5)$, defined on sequence. $\mathcal{A}$ estimates the bit c in game G_i, which are defined as following.

G_0: $\mathcal{A}$ launches this initial attack to the proposed protocol P. Since the bit c is guessed randomly, we have

$$Adv_{P,\mathcal{A}}^{AKE}(t) = |2\,\Pr[SUC_0] - 1| \qquad (1)$$

G_1: Eavesdropping attack is modeled in this game, where $\mathcal{A}$ can query $Execute(\pi^u, \pi^s)$ oracle to intercept the messages Msg_1, Msg_2, and Msg_3 during the login and authentication process. After this game, $\mathcal{A}$ can make some *Test* queries and decide the bit c' according to the intercepted messages. Notice that the session key $SK = h(ID_i||SID_j||r_g||r_j)$ is generated between a user U and a sensor node. To compute SK, $\mathcal{A}$ requires the short-term secrets (r'_g, r_j) and the long term secrets $(LTK, S_{GWN-ID_i}, S_{GWN-SID_j})$, which are unknown to $\mathcal{A}$. Consequently, the probability for $\mathcal{A}$ to win game G_1 does not increase by launching eavesdropping attack. Then, it follows:

$$\Pr[SUC_1] = \Pr[SUC_0] \qquad (2)$$

G_2: With the exception of simulating *Send* and hash queries in G_2, both the games G_1 and G_2 are "indistinguishable". An active attack is modeled in this game, where the goal of $\mathcal{A}$ is to cheat a legitimate participant into the belief that a revised message is sent by a legitimate participant. $\mathcal{A}$ examines the presence of hash collisions by making many queries to the random oracles. It is obvious that no hash collision occurs since all the transmitted messages are generated by involving some random nonces. Assume that $\mathcal{A}$ executes q_h number of the *Send* queries. Then, according to the birthday paradox, it follows:

$$|\Pr[SUC_2] - \Pr[SUC_1]| \leq \frac{q_h^2}{2|Hash|} \qquad (3)$$

G_3: This game regarded as an extension of G_2 simulates *PUF* queries. According to G_2, it follows:

$$|\Pr[SUC_3] - \Pr[SUC_2]| \leq \frac{q_p^2}{2|PUF|} \qquad (4)$$

G_4: The *CorruptSC* and *CorruptSD* queries are simulated in this game. Through the queries to these oracles, $\mathcal{A}$ can extract $< \tau_i^*, EID_i^*, DID_i^*, C_i >$ stored in SC_i and obtain the credentials $< SID_j, S_{GWN-SID_j} >$ from a captured sensor node S'_j. However, for an un-compromised sensor node S_j, $< SID_j, S_{GWN-SID_j} >$ are distinct. U_i uses both password PW_i and biometrics BIO_i. However, the probability of guessing the biometrics secret key σ_i of l_1 bits (respectively, BIO_i) is approximately $\frac{1}{2^l}$. Utilizing the Zipf's law on passwords, $\mathcal{A}$ can also attempt to guess low-entropy passwords. If only the trawling guessing attacks is considered, the advantage of $\mathcal{A}$ will be over 0.5 when $q_s = 10^7$ or 10^8. If the targeted guessing attacks, where $\mathcal{A}$ can make use of the target user's personal information, is considered, the advantage of $\mathcal{A}$ will be over 0.5 when $q_s \leq 10^6$. In practice, if the number of incorrect passwords exceeds a certain number, the system will prevent further input. Since the games G_3 and G_4 are same without guessing attacks, it follows:

$$|\Pr[SUC_4] - \Pr[SUC_3]| \leq max\{C' \cdot q_s^{s'}, \frac{q_s}{2^{l_1}}, \frac{q_s}{2^{l_2}}\} \qquad (5)$$

G_5: Using the decryption information of M_1, M_3, M_5 and M_7, $\mathcal{A}$ attempts to determine the session $SK = h(ID_i||SID_j||r_g||r_j)$ by capturing the messages Msg_1, Msg_2 and Msg_3. However, the secret keys are required to decrypt M_1, M_3, M_5 and M_7. On the basis of the IND-CPA, we have following

$$|\Pr[SUC_4] - \Pr[SUC_5]| \leq Adv_{\Omega}^{IND-CPA}(k) \qquad (6)$$

After executing all the oracles and querying the *Test* query, the probabilistic of $\mathcal{A}$ to guess the bit c is

$$\Pr[SUC_5] = \frac{1}{2} \qquad (7)$$

From Eqs.(1), (2) and (7), we have

$$\begin{aligned} \frac{1}{2} Adv_{P,\mathcal{A}}^{AKE}(t) &= |\Pr[SUC_0] - \frac{1}{2}| \\ &= |\Pr[SUC_1] - \frac{1}{2}| \\ &= |\Pr[SUC_1] - \Pr[SUC_5]| \qquad (8) \end{aligned}$$

Furthermore, the triangular inequality shows

$$\begin{aligned} |\Pr[SUC_1] - \Pr[SUC_5]| &\leq |\Pr[SUC_1] - \Pr[SUC_3]| \\ &\quad + |\Pr[SUC_3] - \Pr[SUC_5]| \\ &\leq |\Pr[SUC_1] - \Pr[SUC_2]| \end{aligned}$$

$$
\begin{aligned}
&+|\Pr[SUC_2]-\Pr[SUC_3]| \\
&+|\Pr[SUC_3]-\Pr[SUC_4]| \\
&+|\Pr[SUC_4]-\Pr[SUC_5]| \\
\le\ &\frac{q_h^2}{2|Hash|}+\frac{q_p^2}{2|PUF|} \\
&+max\{C'\cdot q_s^{s'},\frac{q_s}{2^{l_1}},\frac{q_2}{2^{l_2}}\} \\
&+Adv_{\Omega}^{IND-CPA}(k) \qquad (9)
\end{aligned}
$$

Finally, from Eqs.(8) and (9), we can obtain the required result:

$$
\begin{aligned}
Adv_{P,\mathcal{A}}^{AKE}(t) \le\ &\frac{q_h^2}{|Hash|}+\frac{q_p^2}{|PUF|} \\
&+2(max\{C'\cdot q_s^{s'},\frac{q_s}{2^{l_1}},\frac{q_2}{2^{l_2}}\} \\
&+Adv_{\Omega}^{IND-CPA}(k)).
\end{aligned}
$$

□

B. INFORMAL SECURITY ANALYSIS

The informal security analysises of the proposed protocol against some well-known attacks are described as follows.

- *Stolen Smart Card Attack*: Suppose that the lost/stolen smart card SC_i of a legitimate user U_i is obtained by $\mathcal{A}$. Then, by launching power analysis attack [54], $\mathcal{A}$ can get the credentials $<\tau_i^*, EID_i^*, DID_i^*, C_i>$ from SC_i's memory. $\mathcal{A}$ cannot calculate the correct identity and password using these credentials, since $\tau_i^*, EID_i^*, DID_i^*$ and C_i are calculated from three long unknown random strings S_{GWN-ID_i}, σ_i and τ_i. Therefore, our protocol can avoid smart card lost/stolen attack.
- *User Anonymity and Untraceability*: Suppose that the login request message $Msg_1 =< DID_i, M_1, T_1 >$ is captured by $\mathcal{A}$, where $M_1 = E_{K_{ug}}[SID_j||T_1]$, $DID_i = E_{LTK}[ID_i||x]$. $\mathcal{A}$ can not extract ID_i from Msg_1 without LTK, and U_i's dynamic identity DID will change after U_i accomplishes a successful communication with GWN and S_j. Therefore, our protocol can achieve user anonymity and untraceability.
- *Sensor Node Anonymity and Untraceability*: Sensor node's identity can be revealed by eavesdroping attack. When U_i accesses the sensor data of S_j, SID_j is encrypted in message M_1, which only can be decrypted by GWN and U_i. C_j in Msg_2 will change after a valid execution of the proposed protocol is performed, thus $\mathcal{A}$ cannot trace the sensor node. Therefore, the proposed protocol can achieve sensor node anonymity and untraceability.
- *User Impersonation Attack*: An adversary can fake messages to assure other legitimate participants (e.g., GWN, and sensor nodes) that senders are legitimate entity in this attack. Suppose $\mathcal{A}$ intercept U_i's login request message $Msg_1 =< DID_i, M_1, T_1 >$ and attempts to fake legitimate login request messages, written $Msg_1' = < DID_i', M_1', T_1' >$, using the current timestamp T_1'. However, since the proposed protocol guarantees user anonymity, $\mathcal{A}$ cannot compute a valid M_1' and a secret key K_{ug} without the long-term shared secret S_{GWN-ID_i}. Thus, it is impossible for $\mathcal{A}$ to generate a legitimate message using the intercepted login message. As a consequence, user impersonation attack is impossible in the proposed protocol.
- *GWN Impersonation Attack*: $\mathcal{A}$ can launch this attack by intercepting $Msg_2 =< C_j, M_3, M_5, T_2 >$ and $Msg_4 =< M_7, T_4 >$. However, without the long-term key $S_{GWN-SID_j}$, $S_{GWN-SID_i}$ and V_j, $\mathcal{A}$ can not generate these valid messages. Hence, the proposed protocol can prevent the GWN impersonation attack.
- *Sensing Node Impersonation Attack*: $\mathcal{A}$ can intercept the message $Msg_3 =< M_5, T_3 >$, and try to impersonate S_j. Since $\mathcal{A}$ cannot recreate a secret key K_{gs}, which can be computed using $S_{GWN-SID_j}$ and V_j, he or she cannot generate Msg_3. Thus, the proposed protocol can resist this kind of attack.
- *Privileged-Insider Attack*: Let $\mathcal{A}$ be a privileged-insider attacker within GWN, who could acquire the secret credentials S_{GWN-ID_i} of a registered user U_i during the user registration process. However, $\mathcal{A}$ fails to retrieve U_i's PW_i and σ_i even though $\mathcal{A}$ acquire the stolen/lost SC_i of U_i. This demonstrates that the proposed protocol can prevent the privileged-insider attack.
- *Ephemeral Secret Leakage (ESL) Attack*: The SK among their participants could not be compromised after leakaging session-temporary secrets and long-term secrets. In the proposed protocol, the session key $SK = h(ID_i||SID_j||r_g||r_j)$ is established among U_i and SID_j. The security of the session key SK is then based on the following two cases:
  - *Case 1*: Even if $\mathcal{A}$ has the short term secret credentials r_g^* and r_j, $\mathcal{A}$ cannot calculate session key SK correctly without the long-term key.
  - *Case 2*: If $\mathcal{A}$ has the long-term key LTK, it is also impossible for $\mathcal{A}$ to calculate SK without the short term secrets.
- *Resilience Against Sensing Node Capture Attack*: Among this type attack, adversaries can capture the sensor, then make use of the extracted information $< S_{GWN-SID_j}, SID_j >$ stored in S_j. However, since the information $S_{GWN-SID_j}$, and SID_j are generated randomly, and distinct and independent for all the deployed sensor nodes. $\mathcal{A}$ cannot calculate the session keys of a user and other non-compromised sensors using the extracted information. Furthermore, owing to the characteristic of $PUF(\cdot)$, $\mathcal{A}$ cannot calculate V_j from C_j, which is used to produce the message Msg_2 and decrypt the message Msg_3. Thus, the proposed protocol can prevent sensing node capture attack.
- *Resilience Against Physical Tempering Attacks*: Physical tempering attack that tamper the sensor with PUF will change the behavior of sensor node and invalid PUF. Consequently, during the execution of the proposed authentication protocol, PUF cannot produce the desired

output [7]. Therefore, the gateway can comprehend such attempts at tampering. Besides, since PUF cannot be recreated [45], the proposed protocol is resilient against the tempering attack.

- *Resilience Against Forward Secrecy*: Suppose that $\mathcal{A}$ has obtained the session key SK according to the secret values r_g, r_j under CK-adversary model. All other values are used independently, and therefore, $\mathcal{A}$ will not retrieve any sessions keys previously established after compromision of a particular session. Thus, the proposed scheme provide forward secrecy.

VI. PERFORMANCE COMPARISONS

In this section, security features and performance of the proposed protocol with some known protocols [9], [11], [21], [31], [55] are compared.

A. FUNCTIONALITY COMPARISONS

According to Hussain *et al.*'s comments [56], Das *et al.*'s biometric authentication protocol [55] was vulnerable to the traceability, stolen verifier and stolen smart device attacks. A legal but dishonest user in the system can easily launch the traceability attack. Moreover, the dishonest user after stealing the verifier table and/or parameters stored in sensor node can compute any session key shared among sensor node and users. In addition, Hussain *et al.* indicated that Das *et al.*'s protocol cannot provide the perfect forward secrecy.

Banerjee *et al.* [9] presented a physically secure authentication protocol based on PUF. However, we can demonstrate that Banerjee *et al.*'s protocol cannot resist stolen verifier attack. Let $\mathcal{A}$ be an insider of GWN. Due to his privileges of stealing the verifier table from GWN database, $\mathcal{A}$ may get the challenge-response pairs (ID_d, C_d, R_d). Now based on the verifier information, $\mathcal{A}$ can calculate the session key shared between a user ID_u and a sensor node ID_d as follows:

- Step 1: GWN sends $M_2 =< C_d, Q_g, Auth_{R_d}, Auth_g >$ to SD. $\mathcal{A}$ intercepts the message.
- Step 2: $\mathcal{A}$ computes $k_{ud} = Q_g \oplus R_d$ using the stolen verifier.
- Step 3: SD sends $M_3 =< HQ_R, Q_d, Q_{R'}, Auth_d >$ to User. $\mathcal{A}$ intercepts the message.
- Step 4: $\mathcal{A}$ computes $k_{du} = Q_d \oplus R_d$.
- Step 5: $\mathcal{A}$ can successfully calculate the session key $SK = h(k_{du}||k_{ud}||R_d)$.

Moreover, the challenge-response pairs are finite and will be dissipated fully. Hence, an additional phase of renewaling challenge-response pairs are required. In the proposed protocol, both the procedures of pseduo-identity renewal and challenge-response pairs renewal are not necessary. Moreover, different challenge-response pairs are used since CRPs would update after successful execution of the authentication. Besides, GateWay only stores a single CRPs for every sensor instead of storing a set of CRPs for every sensor, which decreases space complexity for gateway to store CRPs.

Li *et al.* [11] proposed a robust and energy efficient three-factor authentication protocol. However, their protocol was not secure against replay attack, where an adversary can intercept valid login request messages of user and replay to GWN. Due to the lack of checking for the received timestamp, GWN will not refuse the login request. Moreover, the adversary can perform the Denial-Of-Service attack by performing replay attack, where a legitimate user will receive a lot of messages from a legitimate sensor node.

Challa *et al.*'s protocol [31] was a signature-based user authentication key agreement protocol. However, their protocol could not provide the traceability and resist the offline password guessing attack, stolen smart card attack. Wazid *et al.*'s [52] and Banerjee *et al.*'s [16] protocols were vulnerable to physical capture/tempering attacks. Tab.2 summarizes the differences about functionality and security features between our protocol and other protocols [9], [11], [16], [21], [31], [52], [55]. From Tab.2, it is obvious that the protocols presented in [9], [11], [16], [21], [31], [52], [55] are either not secure against some attacks or shortage of some important functionalities.

B. COMPUTING OVERHEAD COMPARISONS

We denote that T_m, T_s, T_f and T_h are the time consumption of ECC point multiplication, a symmetric encryption/decryption, fuzzy extractor and a hash operation, respectively. Based on experimental results reported in [57], [58], the appropriate time required to perform each operation are shown in Tab.3.

To compute the communication overhead, the length of sensor node's identity, user's identity, output of hash function, nonce number are 160 bits. The length of timestamps and ECC point are 32 and 320 bits, respectively. Besides, the block size of symmetric cryptography and PUF challenge-response pairs are 128 bits. The communication overhead including the number of messages exchanged in the proposed protocol and other protocols are presented in Tab.4. The proposed protocol requires lower communication overhead as compared to [11], [21], [31], [52], and requires more than [9], [16], [55].

Tab.5 shows the comparison results of performance. Both [11] and [31] require more computation cost compared with others due to the application of ECC and the fuzzy extractor operations. Since achieving better security and possessing more functionality features, our protocol costs more in communication as compared to the protocols [9], [16], [21], [52], [55].

VII. NS-3 SIMULATION

NS-3 is an open, extensible and discrete-event network simulation platform for for networking research and education [18]. NS-3 can provide a simulation engine for users to simulate the real system in a easy way so that users focus on how to model the Internet protocols and network. Several external animators, data analysis and visualization tools can be used with NS-3 at C++ and Python development.

TABLE 2. Security and functionality features comparisons.

Feature	Turkanović et al. [21]	Li et al. [11]	Challa et al. [31]	Banerjee at al. [9]	Banerjee at al. [16]	Wazid at al. [52]	Das et al. [55]	Ours
$\mathcal{F}_1$	✓	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_2$	✗	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_3$	✓	✓	✗	✓	✓	✓	✗	✓
$\mathcal{F}_4$	✗	✓	✗	✓	✓	✓	✓	✓
$\mathcal{F}_5$	✓	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_6$	✓	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_7$	✓	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_8$	✓	✓	✓	✗	✓	✓	✓	✓
$\mathcal{F}_9$	✗	✓	✗	✓	✓	✓	✓	✓
$\mathcal{F}_{10}$	✓	✓	✓	✓	✓	✓	✗	✓
$\mathcal{F}_{11}$	✗	✓	✗	✓	✓	✓	✓	✓
$\mathcal{F}_{12}$	✓	✓	✓	✓	✓	✓	✗	✓
$\mathcal{F}_{13}$	✓	✗	✓	✓	✓	✓	✓	✓
$\mathcal{F}_{14}$	✓	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_{15}$	✗	✗	✓	✓	✓	✓	✓	✓
$\mathcal{F}_{16}$	✗	✗	✗	✓	✗	✗	✗	✓
$\mathcal{F}_{17}$	✗	✗	✗	✓	✓	✓	✓	✓
$\mathcal{F}_{18}$	✗	✓	✗	✓	✓	✓	✓	✓
$\mathcal{F}_{19}$	N/A	✓	✓	✓	✓	✓	✓	✓
$\mathcal{F}_{20}$	2	3	3	3	3	3	3	3

$\mathcal{F}_1$: user anonymity; $\mathcal{F}_2$: sensor node anonymity; $\mathcal{F}_3$: untraceability; $\mathcal{F}_4$: resilience against offline password guessing attack; $\mathcal{F}_5$: fast detection of erroneous inputs; $\mathcal{F}_6$: session key agreement; $\mathcal{F}_7$: mutual authentication; $\mathcal{F}_8$: resilience against stolen verifier attack; $\mathcal{F}_9$: resilience against impersonation attack; $\mathcal{F}_{10}$: resilience against sensor node physical capture attack; $\mathcal{F}_{11}$: resilience against privileged insider attack; $\mathcal{F}_{12}$: resilience against forward secrecy; $\mathcal{F}_{13}$: resilience reply attack; $\mathcal{F}_{14}$: resilience against man-in-middle attack; $\mathcal{F}_{15}$: resistant to ESL attack; $\mathcal{F}_{16}$: resilience against physical capture/tempering attack; $\mathcal{F}_{17}$: formal security proof under the ROR model; $\mathcal{F}_{18}$: resilience stolen smart card attack; $\mathcal{F}_{19}$: local password and biometric change; $\mathcal{F}_{20}$: number of factors used; ✓ the protocol is secure or supports that functionality feature; ✗ the protocol is insecure or does not support that feature; N/A not applicable.

TABLE 3. Rough operation time.

Notation	Description	Rough computation time (ms)
T_h	One-way hash function	0.5
T_m	ECC point multiplication	63.075
T_a	ECC point addition	16.229
T_s	Symmetric encryption & decryption	8.7
$T_f \approx T_m$	Fuzzy extractor operation	63.075
T_p	Physical Unclonable Function operation	0.43

TABLE 4. Communication overhead comparisons.

Protocol	No. of bits	No. of messages
Turkanović et al. [21]	2720	4
Li et al. [11]	2688	4
Challa et al. [31]	2528	3
Banerjee et al. [9]	2048	3
Banerjee at al. [16]	2304	3
Wazid at al. [52]	2592	4
Das et al. [55]	1536	3
Ours	2490	4

Tab.6 lists the simulation parameters. The simulation is programmed under NS3(3.29) in Linux 64 bit. At the beginning of the simulation, users and sensors are randomly distributed in an area of 100*100 square meters, where users are allowed to move randomly at speed of 2-4m/s and both GWN and sensor will not change their positions at the rest time of the simulation. Fig.5 is generated by the visualization tool of NS3, showed a scenario that 10 users, a gateway and 60 sensors, colored in green, blue and red, respectively, are distributed in a area of 100*100 square meters. 802.11ah is used to simulate wireless communication in IoT environment. Besides, messages length are 55, 68, 88 and 100 bytes, respectively. Every user sends the messages

TABLE 5. Computing overhead comparisons.

Schemes	User	GateWay node	Sensor node	Total cost
Turkanović et al. [21]	$7T_h$ $\approx 3.5ms$	$5T_h$ $\approx 2.5ms$	$7T_h$ $\approx 3.5ms$	$19T_h$ $\approx 9.5ms$
Li et al. [11]	$2T_m + T_f + 2T_s + 17T_h$ $\approx 210.125ms$	$T_m + 4T_s + 8T_h$ $\approx 101.875ms$	$4T_h + 2T_s$ $\approx 19.4ms$	$3T_m + T_f + 8T_s + 19T_h$ $\approx 331.4ms$
Challa et al. [31]	$5T_m + T_f + 5T_h$ $\approx 380.95ms$	$5T_m + 4T_h$ $\approx 317.38ms$	$4T_m + 3T_h$ $\approx 253.8ms$	$14T_m + T_f + 12T_h$ $\approx 952.13ms$
Banerjee et al. [9]	$T_f + 17T_h$ $\approx 71.575ms$	$8T_h$ $\approx 4ms$	$6T_h + T_f + T_p$ $\approx 66.505ms$	$2T_f + 31T_h + T_p$ $\approx 142.08ms$
Banerjee at al. [16]	$12T_h + 3T_s + T_f$ $\approx 95.18ms$	$5T_h + 5T_s$ $\approx 46ms$	$2T_h + 2T_s$ $\approx 17.5ms$	$19T_h + 10T_s + T_f$ $\approx 159.58ms$
Wazid at al. [52]	$13T_h + 2T_s + T_f$ $\approx 86.98ms$	$5T_h + 4T_s$ $\approx 37.3ms$	$4T_h + 2T_s$ $\approx 19.4ms$	$22T_h + 8T_s + T_f$ $\approx 143.68ms$
Das et al. [55]	$T_f + 14T_h$ $\approx 70.075ms$	$9T_h$ $\approx 4.5ms$	$7T_h$ $\approx 3.5ms$	$T_f + 30T_h$ $\approx 78.075ms$
Ours	$7T_h + T_f + 2T_s$ $\approx 83.975ms$	$11T_h + T_f + 5T_s$ $\approx 112.075ms$	$6T_h + T_f + 2T_s + 2T_p$ $\approx 84.335ms$	$24T_h + 2T_p + 3T_f + 9T_s$ $\approx 290.385ms$

TABLE 6. Simulation parameters.

Parameter	Description	
Platform	NS3(3.29) / Linux 64bit	
Network scenarios	No. of users	No. of sensors
1	5	20
2	5	40
3	5	60
4	10	20
5	10	40
6	10	60
Mobility	random (2-4 m/s)	
Simulation Time	1200 s	

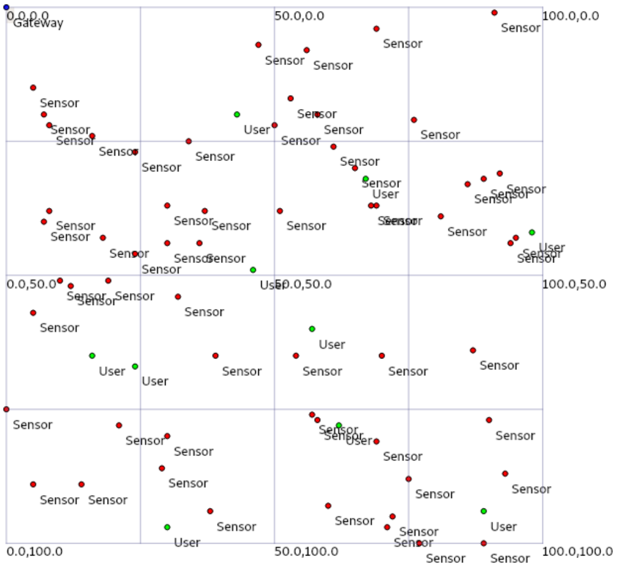


FIGURE 5. Scenario 6.

every 4s. To examine the proposed protocol's performance, network throughput, end-to-end delay and packet delivery rate are considered. Simulation results about the performance are listed in Tab.7.

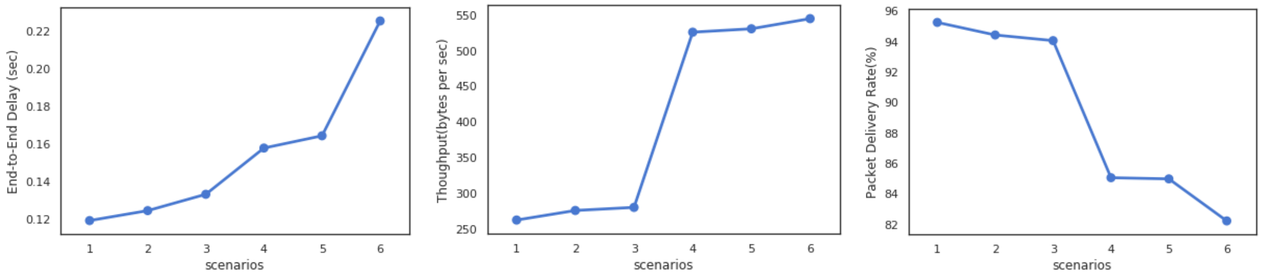
End-to-End Delay : Fig.6a shows the variation of EED in different scenarios. EED can be computed as $[\sum_{k=1}^{n}(T_i^r - T_i^s)]/n$, where n is the number of the received packets. $T_i^r - T_i^s$ is the delay for ith packet, where T_i^r and T_i^s are the received and sent timestamps. With the number of transmitted messages increasing, EED increases according to Fig.6a.

Network Throughput : Fig.6b shows the variation of network throughput in different scenarios. The different scenarios are plotted along the horizontal axis. Network Throughput is calculated as $([\sum(Q_i^r \times l_i)]/T_w)$. T_w is the whole time of simulation. Q_i^r is the length of received packet if the i-th kind and l_i means the packet length of i-th kind. From Fig.6b, it is shown that throughput increases along with the increasing of the number of users and sensors.

Packet Delivery Rate : Fig.6c shows the variation of PDR in different scenarios. Packet delivery rate is the quotient between the number of packet successfully received by the destination and the number of packet sent by the sender. From Fig.6c, it is demonstrated that PDR becomes less if number of sensors and users becomes more due to more congestion.

TABLE 7. Simulation results.

Network scenarios	delay (second)	throughput (bytes per second)	packet delivery rate (%)
1	0.1191	262.72	93.25
2	0.1244	276.11	94.42
3	0.1331	280.43	94.05
4	0.1576	525.53	55.07
5	0.1641	530.29	84.99
6	0.2251	544.48	82.24



(a) End-to-End Delay (b) Throughput (c) Packet Delivery Rate

FIGURE 6. Results.

VIII. CONCLUSION

We have proposed a new lightweight three-factor anonymous user authentication protocol using PUF for IoT environment. Both formal analysis using the ROR model and informal security analysis demonstrate that the proposed protocol is resistant to various known attacks, such as physical capture/tempering attacks and tracking attack. In addition, some security weaknesses were pointed out in several existing user authentication protocols for IoT. Compared with the existing protocols for IoT, the proposed protocol can provide stronger security. Furthermore, simulation using NS3 showed that the proposed protocol is practical and efficient in real IoT environment. In the future, more experiment needed to be carried out to test the proposed protocol by deploying sensor nodes and GWN in IoT environment.

ACKNOWLEDGMENT

The authors would like to thank the handled editor and reviewers for their great support and valuable suggestions.

REFERENCES

[1] A. Botta, W. de Donato, V. Persico, and A. Pescapé, “Integration of cloud computing and Internet of Things: A survey,” *Future Gener. Comput. Syst.*, vol. 56, pp. 684–700, Mar. 2016.

[2] B.-C. Chifor, I. Bica, V.-V. Patriciu, and F. Pop, “A security authorization scheme for smart home Internet of Things devices,” *Future Gener. Comput. Syst.*, vol. 86, pp. 740–749, Sep. 2018.

[3] D. Maresch and J. Gartner, “Make disruptive technological change happen—The case of additive manufacturing,” *Technol. Forecasting Social Change*, vol. 155, Jun. 2020, Art. no. 119216.

[4] G. Liang, S. R. Weller, J. Zhao, F. Luo, and Z. Y. Dong, “The 2015 ukraine blackout: Implications for false data injection attacks,” *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 3317–3318, Jul. 2017.

[5] M. El-hajj, A. Fadlallah, M. Chamoun, and A. Serhrouchni, “A survey of Internet of Things (IoT) authentication schemes,” *Sensors*, vol. 19, no. 5, p. 1141, Mar. 2019.

[6] M. El-hajj, M. Chamoun, A. Fadlallah, and A. Serhrouchni, “Analysis of authentication techniques in Internet of Things (IoT),” in *Proc. 1st Cyber Secur. Netw. Conf. (CSNet)*, Oct. 2017, pp. 1–3.

[7] P. Gope and B. Sikdar, “Lightweight and privacy-preserving two-factor authentication scheme for IoT devices,” *IEEE Internet Things J.*, vol. 6, no. 1, pp. 580–589, Feb. 2019.

[8] S. Garg, K. Kaur, G. Kaddoum, and K.-K.-R. Choo, “Toward secure and provable authentication for Internet of Things: Realizing industry 4.0,” *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4598–4606, May 2020.

[9] S. Banerjee, V. Odelu, A. K. Das, S. Chattopadhyay, J. J. P. C. Rodrigues, and Y. Park, “Physically secure lightweight anonymous user authentication protocol for Internet of Things using physically unclonable functions,” *IEEE Access*, vol. 7, pp. 85627–85644, 2019.

[10] U. Chatterjee, V. Govindan, R. Sadhukhan, D. Mukhopadhyay, R. S. Chakraborty, D. Mahata, and M. M. Prabhu, “Building PUF based authentication and key exchange protocol for IoT without explicit CRPs in verifier database,” *IEEE Trans. Dependable Secure Comput.*, vol. 16, no. 3, pp. 424–437, May 2019.

[11] X. Li, J. Peng, J. Niu, F. Wu, J. Liao, and K.-K.-R. Choo, “A robust and energy efficient authentication protocol for industrial Internet of Things,” *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1606–1615, Jun. 2018.

[12] K. Xue, P. Hong, and C. Ma, “A lightweight dynamic pseudonym identity based authentication and key agreement protocol without verification tables for multi-server architecture,” *J. Comput. Syst. Sci.*, vol. 80, no. 1, pp. 195–206, Feb. 2014.

[13] T.-Y. Wu, Z. Lee, M. S. Obaidat, S. Kumari, S. Kumar, and C.-M. Chen, “An authenticated key exchange protocol for multi-server architecture in 5G networks,” *IEEE Access*, vol. 8, pp. 28096–28108, 2020.

[14] C.-M. Chen, Y. Huang, K.-H. Wang, S. Kumari, and M.-E. Wu, “A secure authenticated and key exchange scheme for fog computing,” *Enterprise Inf. Syst.*, pp. 1–16, Jan. 2020, doi: 10.1080/17517575.2020.1712746.

[15] R. Fan, D.-J. He, X.-Z. Pan, and L.-D. Ping, “An efficient and DoS-resistant user authentication scheme for two-tiered wireless sensor networks,” *J. Zhejiang Univ. Sci. C*, vol. 12, no. 7, pp. 550–560, Jul. 2011.

[16] S. Banerjee, V. Odelu, A. K. Das, J. Srinivas, N. Kumar, S. Chattopadhyay, and K.-K.-R. Choo, ''A provably secure and lightweight anonymous user authenticated session key exchange scheme for Internet of Things deployment,'' *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8739–8752, Oct. 2019.

[17] A. K. Das, P. Sharma, S. Chatterjee, and J. K. Sing, ''A dynamic password-based user authentication scheme for hierarchical wireless sensor networks,'' *J. Netw. Comput. Appl.*, vol. 35, no. 5, pp. 1646–1656, Sep. 2012.

[18] nsnam.org. (2018). *NS-3.28*. [Online]. Available: https://www.nsnam.org

[19] A. K. Das, S. Zeadally, and D. He, ''Taxonomy and analysis of security protocols for Internet of Things,'' *Future Gener. Comput. Syst.*, vol. 89, pp. 110–125, Dec. 2018.

[20] D. Wang and P. Wang, ''On the anonymity of two-factor authentication schemes for wireless sensor networks: Attacks, principle and solutions,'' *Comput. Netw.*, vol. 73, pp. 41–57, Nov. 2014.

[21] 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.

[22] 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. 2016.

[23] M. S. Farash, M. Turkanović, S. Kumari, and M. Hölbl, ''An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment,'' *Ad Hoc Netw.*, vol. 36, pp. 152–176, Jan. 2016.

[24] R. Amin, S. H. Islam, G. P. Biswas, M. K. Khan, L. Leng, and N. Kumar, ''Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks,'' *Comput. Netw.*, vol. 101, pp. 42–62, Jun. 2016.

[25] M. L. Das, ''Two-factor user authentication in wireless sensor networks,'' *IEEE Trans. Wireless Commun.*, vol. 8, no. 3, pp. 1086–1090, Mar. 2009.

[26] R. Watro, D. Kong, S.-F. Cuti, C. Gardiner, C. Lynn, and P. Kruus, ''TinyPK: Securing sensor networks with public key technology,'' in *Proc. 2nd ACM workshop Secur. Ad Hoc Sensor Netw. (SASN)*. New York, NY, USA: ACM, 2004, pp. 59–64.

[27] H.-F. Huang, Y.-F. Chang, and C.-H. Liu, ''Enhancement of two-factor user authentication in wireless sensor networks,'' in *Proc. 6th Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, Oct. 2010, pp. 27–30.

[28] M. K. Khan and K. Alghathbar, ''Cryptanalysis and security improvements of 'Two-factor user authentication in wireless sensor networks,''' *Sensors*, vol. 10, no. 3, pp. 2450–2459, Mar. 2010.

[29] B. Vaidya, D. Makrakis, and H. T. Mouftah, ''Improved two-factor user authentication in wireless sensor networks,'' in *Proc. IEEE 6th Int. Conf. Wireless Mobile Comput., Netw. Commun.*, Oct. 2010, pp. 600–606.

[30] M. Turkanovic and M. Holbl, ''An improved dynamic password-based user authentication scheme for hierarchical wireless sensor networks,'' *Electron. Electr. Eng.*, vol. 19, no. 6, pp. 109–116, Jun. 2013.

[31] S. Challa, M. Wazid, A. K. Das, N. Kumar, A. Goutham Reddy, E.-J. Yoon, and K.-Y. Yoo, ''Secure signature-based authenticated key establishment scheme for future IoT applications,'' *IEEE Access*, vol. 5, pp. 3028–3043, 2017.

[32] P. Porambage, A. Braeken, C. Schmitt, A. Gurtov, M. Ylianttila, and B. Stiller, ''Group key establishment for enabling secure multicast communication in wireless sensor networks deployed for IoT applications,'' *IEEE Access*, vol. 3, pp. 1503–1511, 2015.

[33] P. Tuyls and L. Batina, ''RFID-Tags for anti-counterfeiting,'' in *Topics Cryptology–CT-RSA*, D. Pointcheval, Ed. Berlin, Germany: Springer, 2006, pp. 115–131.

[34] G. E. Suh and S. Devadas, ''Physical unclonable functions for device authentication and secret key generation,'' in *Proc. 44th ACM/IEEE Design Autom. Conf.*, Jun. 2007, pp. 9–14.

[35] P. F. Cortese, F. Gemmiti, B. Palazzi, M. Pizzonia, and M. Rimondini, ''Efficient and practical authentication of PUF-based RFID tags in supply chains,'' in *Proc. IEEE Int. Conf. RFID-Technology Appl.*, Jun. 2010, pp. 182–188.

[36] G. Hammouri, E. Öztürk, and B. Sunar, ''A tamper-proof and lightweight authentication scheme,'' *Pervas. Mobile Comput.*, vol. 4, no. 6, pp. 807–818, Dec. 2008.

[37] Y. S. Lee, H. J. Lee, and E. Alasaarela, ''Mutual authentication in wireless body sensor networks (WBSN) based on physical unclonable function (PUF),'' in *Proc. 9th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jul. 2013, pp. 1314–1318.

[38] M. N. Aman, K. C. Chua, and B. Sikdar, ''Mutual authentication in IoT systems using physical unclonable functions,'' *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1327–1340, Oct. 2017.

[39] 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. Dependable Secure Comput.*, vol. 17, no. 2, pp. 391–406, Mar. 2020.

[40] P. Sarkar, ''A simple and generic construction of authenticated encryption with associated data,'' *ACM Trans. Inf. Syst. Secur.*, vol. 13, no. 4, p. 33, Dec. 2010.

[41] Y. Dodis, L. Reyzin, and A. Smith, ''Fuzzy extractors: How to generate strong keys from biometrics and other noisy data,'' in *Proc. Int. Conf. theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 2004, pp. 523–540.

[42] Y. Dodis, J. Katz, L. Reyzin, and A. Smith, ''Robust fuzzy extractors and authenticated key agreement from close secrets,'' in *Proc. Annu. Int. Cryptol. Conf.* Berlin, Germany: Springer, 2006, pp. 232–250.

[43] C. Bösch, J. Guajardo, A. Sadeghi, J. Shokrollahi, and P. Tuyls, ''Efficient helper data key extractor on FPGAs,'' in *Proc. Int. Workshop Cryptograph. Hardw. Embedded Syst.* Berlin, Germany: Springer, vol. 2008, pp. 181–197.

[44] J. Delvaux, D. Gu, I. Verbauwhede, M. Hiller, and M. M. Yu, ''Efficient fuzzy extraction of PUF-induced secrets: Theory and applications,'' in *Proc. Int. Conf. Cryptograph. Hardw. Embedded Syst.* Berlin, Germany: Springer, 2016, pp. 412–431.

[45] 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.

[46] S. Wu and K. Chen, ''An efficient key-management scheme for hierarchical access control in E-Medicine system,'' *J. Med. Syst.*, vol. 36, no. 4, pp. 2325–2337, Aug. 2012.

[47] D. Dolev and A. Yao, ''On the security of public key protocols,'' *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.

[48] R. Canetti and H. Krawczyk, ''Analysis of key-exchange protocols and their use for building secure channels,'' in *Proc. Adv. Cryptol. (EUROCRYPT)*. Berlin, Germany: Springer, 2001, pp. 453–474.

[49] R. Canetti and H. Krawczyk, ''Universally composable notions of key exchange and secure channels,'' in *Proc. Adv. Cryptol. (EUROCRYPT)*. Berlin, Germany: Springer, 2002, pp. 337–351.

[50] V. Odelu, A. Kumar Das, M. Wazid, and M. Conti, ''Provably secure authenticated key agreement scheme for smart grid,'' *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1900–1910, May 2018.

[51] M. Abdalla, P. Fouque, and D. Pointcheval, ''Password-based authenticated key exchange in the three-party setting,'' in *Public Key Cryptography—PKC*, S. Vaudenay, Ed. Berlin, Germany: Springer, 2005, pp. 65–84.

[52] 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.

[53] D. Wang, H. Cheng, P. Wang, X. Huang, and G. Jian, ''Zipf's law in passwords,'' *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 11, pp. 2776–2791, Nov. 2017.

[54] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, ''Examining smart-card security under the threat of power analysis attacks,'' *IEEE Trans. Comput.*, vol. 51, no. 5, pp. 541–552, May 2002.

[55] 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.

[56] S. Hussain and S. A. Chaudhry, ''Comments on 'Biometrics-based privacy-preserving user authentication scheme for cloud-based industrial Internet of Things deployment,''' *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10936–10940, Dec. 2019.

[57] D. He, N. Kumar, M. Khan, and J.-H. Lee, ''Anonymous two-factor authentication for consumer roaming service in global mobility networks,'' *IEEE Trans. Consum. Electron.*, vol. 59, no. 4, pp. 811–817, Nov. 2013.

[58] Q. Jiang, J. Ma, G. Li, and L. Yang, ''An efficient ticket based authentication protocol with unlinkability for wireless access networks,'' *Wireless Pers. Commun.*, vol. 77, no. 2, pp. 1489–1506, Jul. 2014.

ZHENHUA LIU received the B.S. degree from Henan Normal University, in 2000, and the master's and Ph.D. degrees from Xidian University, China, in 2003 and 2009, respectively. He is currently a Professor and the Ph.D. Supervisor. His research interests include cryptography and information security.

CHANGBO GUO received the B.S. degree from the Harbin University of Science and Technology, in 2018. He is currently pursuing the master's degree in computer technology with Xidian University, China. His research interests include cryptography and authentication protocol.

BAOCANG WANG received the B.S. degree in computational mathematics and the M.S. and Ph.D. degrees in cryptology from Xidian University, China, in 2001, 2004, and 2006, respectively. He is currently a Professor and the Ph.D. Supervisor. His research interests include post-quantum cryptography, fully homomorphic cryptography, number theoretic algorithms, and cloud security.