
Prosanta Gope, Member, IEEE, Ashok Kumar Das, Senior Member, IEEE, Neeraj Kumar, Senior Member, IEEE, and Yongqiang Cheng

Abstract—Industrial Wireless Sensor Network (IWSN) is an emerging class of a generalized Wireless Sensor Network (WSN) having constraints of energy consumption, coverage, connectivity, and security. However, security and privacy is one of the major challenges in IWSN as the nodes are connected to Internet and usually located in an unattended environment with minimum human interventions. In IWSN, there is a fundamental requirement for a user to access the real-time information directly from the designated sensor nodes. This task demands to have a user authentication protocol. To satisfy this requirement, this article proposes a lightweight and privacy-preserving mutual user authentication protocol in which only the user with a trusted device has the right to access the IWSN. Therefore, in the proposed scheme, we considered the physical layer security of the sensor nodes. We show that the proposed scheme ensures security even if a sensor node is captured by an adversary. The proposed protocol uses the lightweight cryptographic primitives, such as one way cryptographic hash function, Physically Unclonable Function (PUF) and bitwise exclusive (XOR) operations. Security and performance analysis shows that the proposed scheme is secure, and is efficient for the resource-constrained sensing devices in IWSN.

Index Terms—Industrial Wireless Sensor Network, Mutual authentication, Key agreement, Physically uncloneable function, Security.

I. INTRODUCTION

The Industrial Wireless Sensor Network (IWSN) value proposition has evolved from simply extending or replacing wired networks to cloud-connected smart object intelligence. Internet Protocol (IP) addressability to the node, reliable mesh networking, field-bus tunneling, proven battery lifetime, and new cloud capabilities are now part of the IWSN landscape. Due to the advancement of the sensing technology, WSNs are becoming important as the Internet provides access to digital information anywhere. Today’s sensor networks can provide remote interaction with the outside physical world. This proliferation of WSNs has enabled several new classes of applications that benefit a large number of applications [1].

Many industrial control systems use WSN in the following applications:

• Environmental sensing: It is one of the basic WSN applications, which is widely used in almost every field of industry. The main objective in the environmental sensing is an efficient information gathering used both for the prevention (real-time or postponed) as well as analysis.
• Condition monitoring: It covers the applications of structural condition monitoring [2], [3], health monitoring in Wireless Body Sensor Network (WBSN) [4] and also machine condition monitoring in an industrial control system.
• Process automation: It provides the information regarding the resources for the production and service provision [5]. In some cases, WSNs can be used for the production performance monitoring, evaluation and improvement.

In IWSNs, the collaborative nature allows many potential advantages over traditional wired industrial monitoring as well as control systems, such as self-organization, flexibility, rapid deployment and inherent intelligent-processing capability [21]. Thus, WSN plays a crucial part in building a highly dependable and self-healing industrial system that can answer to the real-time events in quick time. Hence, it is argued that in order to realize the visualized industrial applications and effective communication protocols, we require the advantages potential gains of WSN [21]. Because of unique characteristics and technical challenges, developing a WSN for industrial applications needs a combination of expertise from various stakeholders (Academia and industry) which are outlined as below [21]:

• The industrial expertise as well as knowledge are needed for application-specific domain.
• The sensor-technology expertise is required to understand various issues related to sensor calibration, transducers as well as clock-drift.
• The Radio Frequency (RF) design and propagation environment expertise is needed to deal with the communication challenges and RF interference issues in industrial environments.
• The networking expertise is also essential in order to understand the hierarchical network architectures, which are required for IWSNs to furnish adaptable and scalable architectures for the heterogeneous applications.

P. Gope is with the Department of Computer Science and Technology, University of Hull, Cottingham Rd, Hull HU6 7RX, United Kingdom (e-mail: prosanta.nitdgp@gmail.com, p.gope@hull.ac.uk).

A. K. Das is with the Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500 032, India (e-mail: iitkgp.akdas@gmail.com, ashok.das@iiit.ac.in).

N. Kumar is with the Department of Computer Science and Engineering, Thapar University, Patiala 147 004, India (e-mail: neeraj.kumar@thapar.edu).

Y. Cheng is with the Department of Computer Science and Technology, University of Hull, Cottingham Rd, Hull HU6 7RX, United Kingdom (e-mail: y.cheng@hull.ac.uk). Corresponding author: P. Gope
Recently, there is a rapid growth of global IWSN market which is mainly attributed to high reliability of wireless technology compared to wired technology. Furthermore, growing trend of smart factories, low cost of wireless sensor nodes and faster deployment are predicted in favor of the WSN market growth. Moreover, growing security concern for automation industry coupled with increasing adoption of sensor networks in order to monitor various processes spurs the demand of IWSNs. On the contrary, availability of multiple wireless communication standards is also expected to have an adverse impact on IWSNs market. The technological advancements in wireless communication and energy consumption without losing accuracy are some of the factors that may disclose new avenues for IWSN market in the near future.

A. Related Work

Since the sensor nodes in IWSN have limited computational and storage capabilities, a lightweight authentication and key agreement protocol is preferred in such a network. In 2006, Wong et al. [5] presented a user authentication scheme for IWSN based on symmetric-key cryptography. In 2007, Tseng et al. [7] showed that the scheme presented by Wong et al. is susceptible to various attacks (e.g., replay and forgery attacks). Independently, Das [8] found that Wong et al.’s scheme is vulnerable to stolen-verifier attacks and then he introduced a new protocol. However, this protocol cannot ensure some of the important security properties, such as mutual authentication and key agreement. Moreover, later studies [9], [10] revealed that the protocol presented in [8] is also vulnerable to insider attacks and impersonation attacks. In 2012, Das et al. [11] and Xue et al. [12] separately proposed two lightweight authentication protocols. However, Turkanovic and Holbl [13] proved that Das et al.’s scheme has some security flaws. In 2013, Li et al. [14] also demonstrated that Xue et al.’s scheme is vulnerable to insider attacks, and they proposed a new scheme to address the vulnerabilities found in Xue et al.’s scheme. In 2014, Turkanovic et al. [15] proposed a new authentication scheme. Chang et al. later pointed out that Turkanovic et al.’s scheme is vulnerable to impersonation attacks. After that, they presented an improved scheme. After thoroughly investigating, we find that the protocol presented in [16] cannot ensure untraceability property, because, in the login message, the parameter \( MI_i \) is fixed for two different sessions. Therefore, an adversary can easily comprehend that both messages belong to the same user. In this way, an adversary can trace the activities of the user. Recently, Gope and Hwang [17] introduced an anonymous mutual authentication protocol for the real-time data access in IWSN. However, their scheme does not support the anonymity of the sensor nodes. The sensors in IWSN are often deployed in the open hostile environment. Hence, there is a possibility of physical and cloning attacks. However, like the other existing protocols in WSN, Gope and Hwang’s scheme cannot ensure the physical security of the sensor nodes either.

B. Motivation

IWSNs offer several benefits in many industrial control systems and various real-time applications. In IWSNs, the sensor devices share the information with each other using a public channel (i.e., the Internet). Therefore, security and privacy of the shared sensing data remains a paramount concern in IWSNs. An intruder (adversary) can gather and aggregate the traffic information in order to make the profile of an industrial plant’s activities (i.e., production status). For sensitive applications in IWSNs (for example, IWSN-based healthcare environment), this type of profile making is dangerous. In addition, if the confidential industrial plant’s information is leaked, the private information can be also exposed to some malicious users. As a result, the security and privacy are important in IWSN-based applications.

In order to access the real-time information directly from some designated sensor nodes in IWSNs, a user first needs to be authenticated by the gateway. Only after mutual authentication among the user and the accessed sensor node with the help of the gateway node, both parties establish a session key between them. Using the established session key, they can communicate securely each other. A user authentication is very much needed in IWSNs to protect the security and privacy, because the private (confidential) industrial plant’s information must be preserved and protected from any adversary. However, designing such a user authentication scheme in IWSNs is a challenging task due to resource-limitations of the sensor nodes and vulnerability of physical capturing of the sensor nodes by an adversary.

C. Research Contributions

This article proposes a lightweight and physically secure anonymous mutual authentication protocol for real-time data access in IWSN. The proposed scheme is based on lightweight cryptographic primitives, such as one-way cryptographic hash function, physically unclonable function (PUF) [18] and bitwise XOR operations, which have limited computational overhead, and hence, it is suitable for the resource-constrained sensing devices in IWSN. The key contributions of this article are summarized as follows:

- A computationally efficient lightweight mutual authentication scheme has been designed that allows only a legitimate user with a trusted device to access the IWSN.
- Physical security of the user’s device as well as the sensor nodes deployed in the open hostile environment is ensured in the proposed scheme.
- In the proposed scheme, we do not require to store any sensitive information, such as secret credentials on the sensing devices.
- The formal security analysis under the widely-used Real-Or-Random (ROR) model [20] ensures the session key (SK) security of the proposed scheme. To further strengthen the security of the proposed scheme, the informal (non-mathematical) security analysis is also carried out.
- A detailed performance analysis and comparison with the existing schemes show that the proposed scheme is suitable for the resource-constrained sensor nodes.
D. Paper Outline

The rest of the paper is constructed as follows. In Section II, we define the system and adversary models applied for the proposed scheme. In addition, we also provide a brief introduction to PUFs in this section. In Section III, the proposed scheme is described in detail. The security and performance analysis of the proposed scheme are discussed in Section IV and Section V respectively. Finally, we conclude the paper in Section VI.

II. Preliminaries

In this section, we first discuss the system model for IWSN and then the adversary model needed for the proposed scheme. Moreover, we briefly discuss the important properties of PUFs.

A. System Model

As shown in Fig. 1, our system model for IWSN consists of five major entities: a set of sensors, a set of base stations, a gateway node, a control and monitoring unit, and a set of users. A user may wish to access the real-time data directly from the sensor nodes. A set of sensors are deployed in a target field (e.g., industrial plant) for environmental and condition monitoring and/or process automation purpose. Sensors collect data from their surrounding environment and periodically transmit them to the gateway (via the nearest base station). After that, the gateway forwards the collected data to the control and monitoring unit through a secure channel. Before receiving any data, the gateway needs to validate the legitimacy of the sensor nodes. In this regard, the gateway also checks whether a sensor node has been physically tampered or not. On the other hand, in many critical applications [3, 7], the users from outside may require to obtain access of the real-time information directly from the sensor nodes instead of the gateway node. An example includes the IWSN-based healthcare environment, where sensors collect the real-time information, such as temperature, blood pressure and pulse rate from a patient’s body. After that, a legitimate medical professional, say a doctor, with a trusted device can get access of these data directly from the sensor nodes. However, before offering any secure direct access of a sensor node, the gateway needs to verify the legitimacy of the user and the gateway also needs to help both the user and the sensor node to establish a session key. So that, they (user and the sensor node) can securely communicate.

B. Adversary Model

We assume that an adversary \( A \) can intercept the transmitted messages communicated over public channel. In addition, \( A \) can alter or delete the message contents transmitted over the insecure public channel as per as the Dolev-Yao (DY) threat model [22]. We further assume that sensors may be deployed out in the open environment and these are not physically protected. Therefore, \( A \) can easily access the sensors and these are subject to physical and cloning attacks. Besides, it is assumed that the user’s device is not physically protected. \( A \) having the access to the user’s trusted device may try to extract information stored in the device. In addition, \( A \) may also attempt to extract sensor data by using an untrusted device. Here, the \( A \)’s objective is to launch an undetectable attack to authenticate itself with the gateway or any one of the registered deployed sensors, which could be dangerous for IWSN applications. For example, if \( A \) can authenticate itself and gain access to a sensor connected to a patient’s pacemaker in the IWSN-based healthcare environment, he/she can cause danger to the life of the patient. Finally, as in [24], the gateway needs to be physically secured by putting it under a locking system inside the IWSN so that the physical capture of the gateway will be much difficult as compared to that for the sensor nodes and the user’s device. Thus, the gateway is considered as a fully trusted node and it will not be compromised by \( A \).

C. Evaluation Criteria

The authors in [34] made a substantial step towards breaking the vicious “break-fix-break-fix” cycle in the existing two-factor authentication research domain for IWSNs. Wang and Wang [35] provided a criteria set, which is originally proposed for a generic client-server architecture. Later, Wang et al. [34] also suggested a comprehensive criteria set of the following independent evaluation metrics for designing a user authentication scheme in IWSNs:

- **No password verifier-table**: Neither the gateway nor the sensor nodes should store the passwords related information in the verifier-table.
- **Password friendly**: A user should be permitted to select his/her password and also to change it freely at any time.
- **No password exposure**: A user’s password should not be extracted or derived by the privileged administrator even if the administrator is treated as a trusted authority in the network.
- **No smart card loss attack**: Having the lost or stolen smart card of a registered authorized user, an adversary should not be able to change the password, recover the password in offline, online or hybrid guessing attacks. In addition, having the extracted information from the lost/stolen smart card, the adversary should not be able to impersonate a victim to login to the system. Hence, it is important that a user authentication should be resilient against the smart card loss attack.
A user authentication scheme in IWSNs should protect various attacks, such as impersonation, offline guessing, replay, man-in-the-middle, parallel, key control, stolen verifier, unknown key share and known key attacks.

**Provision of key agreement**: A registered user and a sensor node should be able to establish a session key among them after their mutual authentication for subsequent communication.

**Sound repairability**: A user authentication scheme should support smart card revocation and dynamic sensor node addition phase after the initial deployment of the sensor nodes in IWSNs.

**No clock synchronization**: A user authentication needs not be affected by clock synchronization and time delay. Hence, the sensor nodes, users and gateway nodes need not be synchronized always in the design of a user authentication scheme in IWSNs.

**Mutual authentication**: A user, the gateway node and a sensor node in IWSNs can authenticate each other during the authentication process.

**Timely typo detection**: In the event of wrong input credentials of a user, such as identity and password by mistakes, he/she will be timely notified.

**Forward secrecy**: A user user authentication scheme designed for IWSNs should provide perfect forward secrecy.

**User anonymity and untraceability**: A user authentication scheme in IWSNs should provide user identity protection as well as untraceability.

Next, devices in IWSN are often deployed in the open and public places, which may cause them to be vulnerable to physical and cloning attacks. Therefore, it is important that any security solution designed for IWSN should not only consider all the aforesaid evaluation metrics but also detect any violations of physical security of the IWSN devices.

### D. Physically Uncloneable Function (PUF)

In this section, we present our proposed scheme which consists of four phases: i) user registration phase, ii) sensor node registration phase, iii) authentication phase, and iv) secure periodical data collection phase.

The important notations used to describe the proposed scheme are listed in Table I. In the proposed scheme, a user’s biometric thumb impression and password have been used for local authentication of password and biometric by the user’s device. Nowadays, most of the mobile devices can be equipped with the biometric scanner so that a user’s biometric thumb impression can be imprinted easily in the devices to overcome the cost of sensing infrastructure in our proposed scheme. Hence, the proposed scheme is suitable for use with the critical applications as well as general purpose applications in the IWSN domain.

#### III. THE PROPOSED SCHEME

In this section, we present our proposed scheme which consists of four phases: i) user registration phase, ii) sensor node registration phase, iii) authentication phase, and iv) secure periodical data collection phase.

The important notations used to describe the proposed scheme are listed in Table I. In the proposed scheme, a user’s biometric thumb impression and password have been used for local authentication of password and biometric by the user’s device. Nowadays, most of the mobile devices can be equipped with the biometric scanner so that a user’s biometric thumb impression can be imprinted easily in the devices to overcome the cost of sensing infrastructure in our proposed scheme. Hence, the proposed scheme is suitable for use with the critical applications as well as general purpose applications in the IWSN domain.

#### A. Assumptions for the Proposed Scheme

In our proposed scheme, we make the following assumptions:

- Each user device and the sensor consist of a microcontroller attached to a PUF.
- It is also impossible to tamper with the communication between the microcontroller and its PUF [13], [19].
- User device and sensors have limited resources while the gateway has no such limitations.
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2895030, IEEE Transactions on Industrial Informatics

B. Phase I: User Registration Phase

Assume that a user $U$ wants to obtain the real-time data access in IWSN. Then, he/she needs to register his/her trusted device $D$ in the gateway. As shown in Fig. 2, the procedure of user registration is described as follows.

Step 1: $U$ selects an identity $ID_u$ and transmits $\{ID_u, Reg_{req}\}$ to the gateway through a secure channel, where $Reg_{req}$ denotes the registration request.

Step 2: Upon receiving the registration request message, the gateway generates a random challenge $C_u$ for normal authentication process. To address the issue of desynchronization or Denial-of-Service (DoS) attacks $[17]$, the gateway also generates a set of new challenges $C_u^{syn} = \{c_1, \ldots, c_n\}$ for resynchronization with the user $U$ and sends the information $\{C_u, C_u^{syn}\}$ to $U$ through the secure channel.

Step 3: After receiving $\{C_u, C_u^{syn}\}$, the $U$’s trusted device extracts the PUF outputs $R_u = P_D(C_u)$, $R_u^{syn} = P_D(C_u^{syn})$ and sends $\{R_u, R_u^{syn}\}$ to the gateway.

Step 4: Hereafter, the gateway randomly generates a unique temporary identity $TID_u$ and a set of unlinkable pseudo identities $PID = \{pid_1, pid_2, \ldots, pid_n\}$ and transmits $\{TID_u, PID\}$ to $U$. The gateway stores $\{(ID_u, TID_u, C_u, R_u, (C_u^{syn}, R_u^{syn}), PID\}$ for further communication with $U$.

Step 5: Upon receiving $\{TID_u, PID\}$, $U$ stores them in his/her device. Next, $U$ inputs his/her biometric thumb impression $\beta_u$ into the device. The device extracts $\alpha_u = P_D(\beta_u)$ and then the user $U$ selects a password $psw_u$ and inputs $psw_u$ into the device. The device computes and stores $\delta = h(\alpha_u || psw_u)$ for user verification.

C. Phase II: Sensor Node Registration Phase

While a new sensor node $Sn$ is deployed, it is needed to register $Sn$ in the gateway. The entire registration process of a new sensor node can be discussed as follows.

Step 1: The gateway first generates a challenge $C_{Sn}$ for the interaction with the sensor node $Sn$. Next, to address the issue of desynchronization or DoS attacks, the gateway also generates a set of new challenges $C_{Sn}^{syn} = \{c_1, \ldots, c_n\}$ for resynchronization with $Sn$. If the sensor wants to obtain the real-time data access in IWSN, then, he/she needs to register his/her trusted sensor node $Sn$. However, $Sn$ does not require to store any secret credentials in its memory. The summary of this phase is depicted in Fig. 3.

D. Phase III: Authentication Phase

Assume that a user $U$ wants to obtain real-time data access directly from a particular sensor node in IWSN, then he/she requires to accomplish mutual authentication with the gateway and the desired sensor node. The process of mutual authentication and key agreement is described as follows. The summary of this phase is also provided in Fig. 4.

Step 1: $U$ first inputs his/her biometric thumb impression $\beta_u$ into his/her device. After that, the device extracts the PUF output $\alpha_u = P_D(\beta_u)$ and asks the user $U$ to enter his/her password. $U$ then inputs his/her password $psw_u$ into the device and the device calculates $\delta^* = h(\alpha_u || psw_u)$ and compares the computed $\delta^*$ with the stored $\delta$. If they are not equal, the device terminates the session. Otherwise, the device believes $U$ as a legitimate user. Next, the device generates a nonce $N_u$ and selects the temporary identity of the user $U$ as $TID_u$ and sends the login message $\{TID_u, N_u\}$ through a public channel.

Step 2: After receiving the login message, the gateway first locates the $TID_u$ in its database. The gateway then selects the $CRP$ ($C_u, R_u$) and generates a nonce $N_g$, and calculates $N_g = N_g \oplus R_u$, $V_0 = h(N_g \oplus R_u || N_u)$. Finally, the gateway composes the authentication request message $MA_u$: $\{C_u, N_g^*, V_0\}$ and sends it to the user $U$ through a public channel.

Step 3: Upon receiving the message $MA_u$, the $U$’s trusted device extracts $R_u = P_D(C_u)$ and verifies the parameter $V_0$. If the verification is successful, the device calculates $N_g = N_g^* \oplus R_u$, $C_u^{new} = h(C_u || R_u)$, $R_u^{new} = P_D(C_u^{new})$. Next, the device asks $U$ to input his/her identity $ID_u$ and the identity $ID_{Sn}$ of the accessed sensor node, say $Sn$, that he/she wants to access. The device then computes $R_u = h(ID_u \oplus R_u) \oplus R_u^{new}$, $ID_{Sn} = h(ID_u \oplus N_g) \oplus ID_{Sn}$, $V_1 = h(R_u \oplus R_u \oplus N_g \oplus ID_{Sn}^*)$ and sends the authentication response...
message $M_{A_2}$: \{ $R_u^* $, $ID_{Sn}^*$, $V_1$\} to the gateway through a public channel.

**Step 4:** After receiving the authentication response message $M_{A_1}$, the gateway verifies the key-hash output $V_1$. If the verification is successful, the gateway derives $ID_{Sn} = h(ID_u \Vert N_g) \oplus ID_{Sn}^*$ and selects the CRP $(C_{Sn}, R_{Sn})$. Next, the gateway generates a random number $n_1$ and the session key $SK$, and then calculates $n_1^* = n_1 \oplus R_{Sn}$, $SK_{Sn}^* = h(R_{Sn} \Vert n_1^*) \oplus SK$, $V_2 = h(n_1^* \Vert R_{Sn} \Vert SK_{Sn} \Vert TID_{A_1})$. Finally, the gateway composes a message $M_{A_3}$: \{ $TID_{u}^*$, $n_1^*$, $SK_{Sn}^*$, $C_{Sn}$, $V_2$\} and sends it to the sensor $Sn$ through a public channel.

**Step 5:** Upon receiving the message $M_{A_3}$, the sensor $Sn$ first extracts the PUF output $R_{Sn} = P_{Sn}(C_{Sn})$ and then verifies the parameter $V_2$. After successful verification, the sensor $Sn$ computes $n_1 = n_1^* \oplus R_{Sn}$, $SK = h(R_{Sn} \Vert n_1) \oplus SK_{Sn}^*$, $C_{Sn}^* = h(C_{Sn} \Vert R_{Sn})$, $R_{Sn}^* = P_{Sn}(C_{Sn}^*)$, $R_{Sn} = h(R_{Sn} \Vert R_{Sn}^*)$, and $V_3 = h(R_{Sn}^* \Vert R_{Sn})$. Next, the sensor composes a message $M_{A_4}$: \{ $R_{Sn}, V_3$\} and sends it to the gateway through a public channel.

**Step 6:** On receiving the message $M_{A_4}$, the gateway verifies the parameter $V_3$. If the verification is successful, the gateway calculates $C_{Sn}^* = h(C_{Sn} \Vert R_{Sn})$, $R_{Sn}^* = h(R_{Sn} \Vert R_{Sn}^*)$, $C_u^* = h(C_u \Vert R_u)$ and $R_u^* = h(ID_u \Vert R_u) \oplus R_u^*$. The gateway also generates a new temporary identity for the user $U$ as $TID_u^{new}$ and computes $SK_u^* = h(ID_u \Vert R_u \Vert N_g \oplus SK \Vert TID_{u}^* = h(ID_u \Vert R_u \Vert TID_{A_1} \oplus TID_u^{new})$ and $V_4 = h(TID_u^{new} \Vert SK_u^*)$. Finally, the gateway composes a message $M_{A_5}$: \{ $TID_u^{new}$, $SK_u^*$, $V_4$\} and sends it to the user $U$ through a public channel. After that, the gateway stores \{ $TID_u^{new}$, $C_{u}^{syn}$, $R_u^{new}$, $(C_{Sn}, R_{Sn})$\} for further interactions with the user $U$ and the sensor node $Sn$.

**Step 7:** Upon receiving the message $M_{A_5}$, the user $U$’s device validates the parameter $V_4$. If the validation is successful, the device computes the session key $SK = h(ID_u \Vert R_u \Vert N_g) \oplus SK_u^*$, $TID_u^{new} = h(ID_u \Vert R_u \Vert TID_u) \oplus TID_u^{new}$ and stores $TID_u^{new}$ for the further communication with the user $U$.

**Remark 1:** It is worth noting that if the gateway cannot find the $TID_u$ in its database, the gateway will ask the device to try with a valid unlinkable pseudo identity $pid_j \in PID$. In that case, the gateway will select one of the unused CRP from \{ $C_{u}^{syn}$, $R_u^{new}$\}. Similarly, if there is any loss of synchronization between the sensor node $Sn$ and the gateway, this can be apprehended if the gateway cannot receive the response message $M_{A_4}$ within the maximum round-trip time period. Then, the gateway will select of the unused CRP from \{ $C_{u}^{syn}$, $C_{Sn}^*$\} for authentication process and after completion of the authentication process, the gateway needs to delete that particular used CRP from its database.
Sn node factor authentication scheme, the temporary identity dispensable to build a two-factor authentication scheme for principle that the public-key techniques are intrinsically in-

\[ N = \text{Sn first extracts the PUF output} \]

Sn verifies the parameter session key \( C \) \( \text{R} \( \text{GW} = \text{Sn} \) \)

establish the session key \( sk \) this session key will use the following mechanism, both the gateway and a sensor node have been physically tampered or not. More-

Remark 2: Wang and Wang [29] analyzed several two-factor authentication schemes. After receiving the response message from the sensor \( Sn \), the gateway first verifies the parameter \( \text{Res}_{\text{GW}} \) \( \text{Sn} \text{new} \) \( \text{x} \) \( \text{Res}_{\text{GW}} \)

Step 1: The gateway generates a nonce \( N_g \) and selects the CRP \( \text{C}_{\text{Sn}}, \text{R}_{\text{Sn}} \). Next, the gateway calculates \( N_g = \text{R}_{\text{Sn}} \oplus N_g \), \( \text{Res}_{\text{GW}} = h(\text{R}_{\text{Sn}} \| N_g^* ) \) and sends the message \( \{ \text{C}_{\text{Sn}}, N_g^*, \text{Res}_{\text{GW}} \} \) to the sensor through a public channel.

Step 2: Upon receiving \( \{ \text{C}_{\text{Sn}}, N_g^*, \text{Res}_{\text{GW}} \} \), the sensor \( Sn \) first extracts the PUF output \( R_{\text{Sn}} = P_{\text{Sn}}(\text{C}_{\text{Sn}}) \) and then verifies the parameter \( \text{Res}_{\text{GW}} \). If the verification is successful, \( Sn \) generates a nonce \( N_x \) and calculates \( N_g = R_{\text{Sn}} \oplus N_x \), \( C_{\text{Sn}} = h(\text{C}_{\text{Sn}} \| N_x ) \), \( R_{\text{Sn}} = P_{\text{Sn}}(C_{\text{Sn}}), x = R_{\text{Sn}} \oplus N_x \), \( N_x^* = N_x \oplus h( R_{\text{Sn}} \| N_x ) \), \( \text{Res}_{\text{Sn}} = h( N_x^* \| R_{\text{Sn}} ) \), and the session key \( sk = h( N_x \| N_x \| R_{\text{Sn}} \) ). Finally, the \( Sn \) composes a message \( \{ N_x^*, x, \text{Res}_{\text{Sn}} \} \) and sends it to the gateway through a public channel.

Step 3: After receiving the response message from the sensor node \( Sn \), the gateway first verifies the parameter \( \text{Res}_{\text{Sn}} \). If the validation is successful, the gateway computes \( N_x = N_x^* \oplus h( R_{\text{Sn}} \| N_x ) \), \( C_{\text{Sn}} = h(\text{C}_{\text{Sn}} \| N_x ) \), \( R_{\text{Sn}} = x \oplus N_x \), and the session key \( sk = h( N_x \| N_x \| R_{\text{Sn}} \) ). Finally, the gateway stores \( \{ C_{\text{Sn}}, R_{\text{Sn}} \} \) for further interaction with the sensor node \( Sn \).

Remark 2: Wang and Wang [29] analyzed several two-factor authentication schemes in WSNs and came up with a general principle that the public-key techniques are intrinsically indispensable to build a two-factor authentication scheme for supporting user anonymity property. In the proposed three-factor authentication scheme, the temporary identity \( TID_u \) of a user \( U \) is used instead of using the original identity \( ID_u \) in the communicating messages to provide user anonymity property. Also, the temporary identity is renewed for each session to provide the untraceability property. Thus, considering the resource limitations of sensor nodes in IWSN, the lightweight operations like one-way hash function \( h(\cdot) \) and PUF are used instead of using expensive public-key techniques. Of course, the Elliptic Curve Cryptography (ECC) is feasible for resource-constrained WSNs [30] though it is not as lightweight as one-way hash function and PUF. Hence, to achieve a strong user anonymity property it is required to use public key techniques (for example, ECC) for IWSN [31].

Remark 3: In this article, we have constructed a privacy-preserving scheme by using lightweight hash functions and PUFs, and pre-loading a pool of pseudonym identities. If the pseudonym ID pool is large, the large storage capacity is needed. If the pool is small, the device of a user \( U \) needs to update frequently. Since the device is not resource-limited as compared to a smart card, the approach of pre-loading pseudonym identities in the proposed scheme will not be a limitation.

IV. SECURITY ANALYSIS OF THE PROPOSED SCHEME

In this section, we formally analyze the security of the proposed scheme. In this context, we consider the broadly-accepted Real-Or-Random (ROR) model provided in [20]. Moreover, we also informally (non-mathematically) analyze the security of the proposed scheme.

A. Security Model

We now discuss the ROR model [20] in the following.

Participants: Let \( \Pi_{Sn}^i \) be the \( t \)-th instance of the sensor node \( Sn \), \( \Pi_{U_i}^i \) the \( a \)-th instance of the user \( U_i \) and \( \Pi_{GW}^v \) the instance \( v \) of the gateway GW.

Partnering: \( \Pi_{Sn}^i \), is said to be partner of \( \Pi_{U_i}^j \), when partial transcript of all messages exchanged between the user \( U_i \) and the sensor \( Sn_j \) is unique. The communication for the current session is defined by a session id \( sid \).

Freshness: If the session key \( SK \) between \( U_i \) and \( Sn_j \) is not divulged to an adversary \( A \), the instance \( \Pi_{U_i}^i \) or \( \Pi_{Sn_j}^j \) is said to be fresh.

Adversary: Under the ROR model, \( A \) cannot only read the transmitted messages, but also can modify, delete or change the message contents during the communication. In other words, \( A \) is allowed to have full control over the communication. Moreover, \( A \) will have access to the queries defined below:

- Execute (\( \Pi_{U_i}^i, \Pi_{Sn_j}^j \)): With the help of this query, the transmitted messages between the valid parties \( U_i \) and \( Sn_j \) are intercepted by \( A \). It is modeled as an eavesdropping attack.
- Send(\( \Pi_{U_i}^i, m \)): This query helps a participant instance \( \Pi_{U_i}^i \) to transmit a message \( m \) and also receives a message, which is modeled as an active attack.
- CorruptDevice(\( \Pi_{Sn_j}^j \)): It implements the user’s lost/stolen device attack. Using this query, the secret credentials stored in device are revealed to \( A \).
• CorruptSensor ($\Pi^1$): This query models an attack in which security credentials stored in the sensor node $S_{nj}$ are also compromised.
• Test($\Pi^2$, $\Pi^u$): The semantic security of session key $SK$ between $U_i$ and $S_{nj}$ following the indistinguishability in the ROR model [20] is implemented under this query. At first, an unbiased coin $c$ is flipped prior to beginning of the game and the output is only secret to $A$. This value is later utilized to verify whether the output of the Test query is consistent. If $A$ executes this query and it is found that the session key $SK$ is fresh, $\Pi^f$ or $\Pi^u$ delivers $SK'$ when $c = 1$ or a random number when $c = 0$; otherwise, it delivers $\bot$ (null).

Semantic security of session key. Based on the ROR model, the adversary $A$ has to distinguish between an instance’s actual session key and a random secret key. $A$ can make the Test queries to either $\Pi^f$ or $\Pi^u$, and its output is checked for consistency against the random bit $c$. Once the game is over, $A$ judges a guessed bit $c'$ for winning purpose. $A$ can win the game when $c' = c$. The advantage of $A$ in breaking the semantic security of the proposed authenticated key agreement protocol, say $P$ in time $t$ is denoted and defined by $Adv^{AKE}_P(t) = |2Pr[Succ] - 1|$, where Succ represents an event that $A$ can win the game.

Random Oracle: In this article, the participants and the adversary $A$ have access to a collision resistant one-way cryptographic hash function $h(.)$ and the secure PUF function $PS(.)$, which are further modeled by the random oracles.

B. Formal Security Analysis
To prove the semantic security of the proposed scheme, we first define collision-resistant one-way hash function $h(.)$, the PUF function $PS(.)$ and also the properties of Zipf’s law in passwords [26].

Definition 1 (Collision-resistant one-way hash function): A collision-resistant one-way hash function $h: \{0,1\}^n \rightarrow \{0,1\}^m$ is a deterministic mathematical function that takes a variable length input string and produces a fixed length output string, say $n$ bits. If $Adv_{PS}^{Hash}(t)$ denotes the advantage of an adversary $A$ in finding a hash collision, $Adv_{PS}^{Hash}(t) = Pr[\{i_1, i_2\} \in R_A : i_1 \neq i_2, h(i_1) = h(i_2)]$. An $(\epsilon, rt)$-adversary $A$ attacking the $h(.)$’s collision resistance means that $Adv_{PS}^{Hash}(t) \leq \epsilon$ with at most run time $rt$.

Definition 2 (Secure PUF function): We say PUF is a secure PUF if the following requirement holds. For arbitrary inputs $C_1, C_2 \in \{0,1\}^k$, the variation from the same inputs is at most $d_1$ and the variation from the different outputs is at least $d_2$, where $d_1$ and $d_2$ are security parameters. This implies that for any two PUFs, say $PFS_1(.)$ and $PFS_2(.)$, and for any input $C_1 \in \{0,1\}^k$, $Pr[|HD(PUF_1(C_1), PUF_2(C_2))| > d] = 1 - \epsilon$, where HD denotes the hamming distance and $d$ is the error-tolerance threshold value.

The study conducted in [22] comprises anonymized password-word histograms representing almost 70 million Yahoo! users, mitigating privacy concerns while enabling analysis of dozens of subpopulations based on demographic factors and site usage characteristics. Wang et al. [26] used the Zipf’s law that is a vastly different distribution from the uniform distribution for the user-chosen passwords. Actually the size of password dictionary is much constrained in the sense that the users may not use the whole space of passwords, but rather a small space of the allowed characters space [26]. We use the Zipf’s law in proving the session key security of the proposed scheme in Theorem 1 which is also applied in many recent authentication protocols [22, 23].

Theorem 1: Let $A$ be a polynomial time adversary running in time $t$ against our protocol $P$ and $l$ be the number of bits in the biometric thumb impression $\beta_u$. Then the advantage of $A$ in breaking the semantic security of the proposed scheme for deriving the session key $SK$ is estimated by

$$Adv^{AKE}_P(t) \leq \frac{q_h^2}{|Hash|} + \frac{q_P^2}{|PUF|} + 2\max\left(C' \cdot q_s', q_s \cdot 2^l\right),$$

where $q_h$, $q_P$, $q_s$, $|Hash|$ and $|PUF|$ denote the number of hash queries, the number of PUF queries, the number of Send queries, the range space of $h(.)$ and the range space of $P(.)$, respectively, and $C'$ and $s'$ are the Zipf’s parameters [26].

Proof 1: We follow the similar proof as presented in [22, 23]. A sequence of five games, denoted by $G_i$, where $i = [0,4]$, are defined for proving the session key security of the proposed scheme. These games are essentials where $Succ_i$ denotes the event wherein the adversary $A$ succeeds in guessing the bit $c$ in game $G_i$. The detailed description of each game is given below.

Game $G_0$: It is considered as an actual attack by $A$ against the proposed authentication key exchange (AKE) scheme $P$ in the ROR model. Since the bit $c$ needs to be chosen at the start of $G_0$, it is clear that

$$Adv^{AKE}_P(t) = |2Pr[Succ_0] - 1|.$$ (1)

Game $G_1$: This game is modeled as an eavesdropping attack in which $A$ intercepts the transmitted messages $M_{A_1}$:

$$\{TID_u, N_u, M_{A_2} : \{C_u, N_u, V_0, M_{A_3} : \{R_u^*, ID_{Su}^*, V_1\}, M_{A_4} : \{TID_u, N_u, SK_{Su}^*, V_2\}, M_{A_5} : \{R_{Su}^*, V_3\}, M_{A_6} : \{TID_u^*, SK_{Su}^*, V_4\}\}$$

during the authentication phase. Under this game, $A$ invokes $Execute(\Pi^f, \Pi^u)$ query. After that $A$ makes the Test query to verify whether it is the real session key $SK$ or a random number. In our proposed scheme, $SK$ is computed as $SK = h(ID_u || R_u || N_u) \oplus SK_{Su}^*$, $h(R_{su} || N_u) \oplus SK_{Su}^*$. In this regard, the computation of $SK$ demands the exposure of the secret credentials $ID_u, R_u$ and $R_{su}$, and these credentials are unknown to $A$, where only a legitimate user’s device and the intended legitimate sensor node can compute the desired response $R_u$ or $R_{su}$. Therefore, $A$’s probability in winning $G_1$ by eavesdropping on the exchanged messages in not increased. It then follows that

$$Pr[Succ_1] = Pr[Succ_0].$$ (2)

Game $G_2$: The difference between this game and the previous game $G_1$ is that the simulations of the $Send$ and $Hash$ queries are included in $G_2$. Therefore, it can be treated as an active attack where $A$ may try to fool a legitimate entity to accept a message modified by $A$. Since all messages $M_{A_1}, \ldots, M_{A_6}$ are constructed using the random secrets $R_u$ and/or $R_{su}$,
and no hash collision happens when $A$ makes Send query with help of $h(.)$ query (see Definition 1). According to birthday paradox, we have the following relationship:

$$|Pr[Succ_2] - Pr[Succ_1]| \leq q_s^2/(2|Hash|).$$

(3)

Game $G_3$: The difference between $G_2$ and $G_3$ is that simulations of the Send and PUF queries are included in $G_3$. Therefore, in a similar argument posed in $G_2$, due to the secure PUF function (see Definition 2) we have the following relationship:

$$|Pr[Succ_3] - Pr[Succ_2]| \leq q_s^2/(2|PUF|).$$

(4)

Game $G_4$: In this final game, the simulation CorruptDevice and CorruptSensor are included. In this context, $A$ can obtain the information $\{TID_u, PID, \delta\}$ stored in the device of the user $U$. But $A$ cannot obtain any information from CorruptSensor since the sensor nodes do not store any secret credentials in our proposed scheme. $A$ uses both password $psw_u$ and the biometric thumb impression $\beta_u \in \{0,1\}^l$. Due to use of PUF, the probability of guessing the thumb impression $\beta_u$ is $\frac{1}{2^l}$. $A$ can try to guess low-entropy passwords using the Zipf’s law on passwords [26]. If we only consider the trawling guessing attacks, the actually the advantage of $A$ will be over 0.5 when $q_s = 10^7$ or $10^8$ [26, 32]. When we further consider the targeted guessing attacks (in which $A$ can make use of the target user’s personal information), the advantage of $A$ will be over 0.5 when $q_s \leq 10^6$ [33].

We also impose a restriction on the limited number of wrong password inputs in the system. Since the games $G_3$ and $G_4$ are identical in the absence of the guessing attacks, it follows that

$$|Pr[Succ_4] - Pr[Succ_3]| \leq \max\left(C' \cdot q_s^2, \frac{q_s}{2^l}\right).$$

(5)

Next, since all queries are made by the $A$, the last resource for winning the game is random guessing the bit after invoking the Test query. Therefore, we have

$$Pr[Succ_4] = \frac{1}{2}.$$  

(6)

(1), (2) and (6) give the following relationship:

$$\frac{1}{2} \cdot Adv^{AKE}_p(t) = |Pr[Succ_0] - \frac{1}{2}| = |Pr[Succ_1] - \frac{1}{2}| = |Pr[Succ_1] - Pr[Succ_4]|.$$  

(7)

Applying the triangular inequality, and (3), (4) and (5), we have the following result:

$$|Pr[Succ_1] - Pr[Succ_4]| \leq |Pr[Succ_1] - Pr[Succ_3]| + |Pr[Succ_3] - Pr[Succ_4]| \leq |Pr[Succ_1] - Pr[Succ_2]| + |Pr[Succ_2] - Pr[Succ_3]| + |Pr[Succ_3] - Pr[Succ_4]| \leq q_s^2/(2|Hash|) + q_s^2/(2|PUF|) + \max\left(C' \cdot q_s^2, \frac{q_s}{2^l}\right).$$

(8)

Finally, solving (7) and (8), we obtain the required result:

$$Adv^{AKE}_p(t) \leq \frac{q_s^2}{|Hash|} + \frac{q_s^2}{|PUF|} + 2\max\left(C' \cdot q_s^2, \frac{q_s}{2^l}\right).$$

C. Informal Security Analysis

In this section, we also informally (non-mathematically) analyze the security of the proposed scheme for the following security features and attacks.

1) Attainment of Mutual Authentication: In the authentication phase of the proposed scheme, the user authenticates the gateway by using key hash output $V_0$ and the gateway also authenticates the user by checking whether $h(R^u_n||R_n||ID_{Sn})$ matches with the received $V_1$. However, without knowing $R_u$ it will be computationally infeasible for an adversary $A$ to forge the authentication message of a legitimate user or gateway. Similarly, the sensor authenticates the gateway by checking whether $h(n_t||R_{Sn}||SK^s_{Sn}||TID_u)$ matches with the received $V_2$, and the gateway also authenticates the sensor by verifying whether $h(R_{Sn}^{\ast}||R_{Sn}||ID_{Sn})$ matches with the received $V_3$. On the other hand, in the physical-temper checking and secure periodical data collection phase of the proposed scheme, the sensor node authenticates the gateway by checking whether $h(R_{Sn}||N^s_{t})$ matches with the received $R_{Sn}^{GW}$, and the gateway also authenticates the sensor node by verifying whether $h(N^t_{Sn}||R_{Sn})$ matches with the received $R_{Sn}^{GW}$. In this way, the proposed scheme ensures mutual authentication property among the communicating parties.

2) Attainment of User Anonymity with Untraceability: In order to accomplish user anonymity with untraceability, the proposed scheme employs temporary identity $TID_u$ as an identifier in the transmitted message instead of the user’s real identity $ID_u$. Therefore, except the gateway, no one can identify the user $U$. Moreover, since the temporary identity ($TID_u$) of the user $U$ is randomly generated and changes after completing of each session. Therefore, it is computationally infeasible for an adversary to revive the user’s real identity from the transmitted messages. On the other hand, in case loss of synchronization or DoS attacks, the user $U$ needs to use one of the used pseudo identity $pid_j \in PID$, and after that both the gateway and the user need to delete that pseudo identity. In this way, the proposed scheme ensures user anonymity along with untraceability properties.

3) Attainment of Sensor Anonymity: In the proposed scheme, when a user $U$ needs to obtain data from a particular sensor $Sn$, $U$ encodes the real identity of the sensor $Sn$ as
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2895030, IEEE Transactions on Industrial Informatics

Table II

<table>
<thead>
<tr>
<th>Scheme</th>
<th>$I_1$</th>
<th>$I_2$</th>
<th>$I_3$</th>
<th>$I_4$</th>
<th>$I_5$</th>
<th>$I_6$</th>
<th>$I_7$</th>
<th>$I_8$</th>
<th>$I_9$</th>
<th>$I_{10}$</th>
<th>$I_{11}$</th>
<th>$I_{12}$</th>
<th>$I_{13}$</th>
<th>$I_{14}$</th>
<th>$I_{15}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Das et al. [11]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Turkanovic et al. [15]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>x</td>
<td>✓</td>
<td>x</td>
<td>x</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Chang-Le [16]</td>
<td>x</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gope-Hwang [17]</td>
<td>✓</td>
<td>x</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Our</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

Note: $I_1$: user anonymity with untraceability; $I_2$: sensor anonymity; $I_3$: forward secrecy; $I_4$: replay attack; $I_5$: loss of device attacks; $I_6$: physical attacks; $I_7$: man-in-the-middle attack; $I_8$: no password verifier-table; $I_9$: password friendly; $I_{10}$: no password exposure; $I_{11}$: provision of key agreement; $I_{12}$: sound repairability; $I_{13}$: no clock synchronization; $I_{14}$: mutual authentication; $I_{15}$: timely typo detection.

✓: a scheme is secure against an attack or preserves a feature; x: a scheme is insecure against an attack or it does not preserve a feature.

Table III

<table>
<thead>
<tr>
<th>Scheme</th>
<th>User Device</th>
<th>Gateway</th>
<th>Sensor Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Das et al. [11]</td>
<td>$11N_H$</td>
<td>$11N_H$</td>
<td>$6N_H$</td>
</tr>
<tr>
<td>Turkanovic et al. [15]</td>
<td>$8N_H$</td>
<td>$8N_H$</td>
<td>$6N_H$</td>
</tr>
<tr>
<td>Chang-Le [16]</td>
<td>$2N_{Exp}+7N_H$</td>
<td>$9N_H$</td>
<td>$2N_{Exp}+5N_H$</td>
</tr>
<tr>
<td>Gope-Hwang [17]</td>
<td>$3N_P+6N_H$</td>
<td>$9N_H$</td>
<td>$2N_P+4N_H$</td>
</tr>
</tbody>
</table>

$ID^*_Sn = h(ID_u∥N_g)\oplus ID_Sn$ and sends $ID^*_Sn$ to the gateway. Except the gateway, no one can decode $ID_Sn$ from the $ID^*_Sn$. Moreover, the random number $N_g$ changes in each session. Therefore, $h(ID_u∥N_g)$ is used as an effective one-time pad to encode $ID_Sn$ due to collision resistant property of one-way hash function $h(\cdot)$ (see Definition 1). Hence, no adversary can differentiate $ID^*_Sn = h(ID_u∥N_g)\oplus ID_Sn$ from a randomly chosen string. As a result, the proposed scheme preserves the sensor anonymity too.

4) Attainment of Forward Secrecy: In the proposed scheme, suppose an adversary $A$ has obtained the PUF responses $R_u$ and $R_{Sn}$. However, $A$ still cannot revive the session key $SK$. After each successful session, the gateway also updates its database with the new PUF responses $R^u_{new}$ and $R^u_{new}$, which can not be obtained by $A$ (see Definition 3). Hence, the proposed scheme ensures forward secrecy property.

5) Protection Against Physical Attacks: Suppose an adversary wants to perform physical tampering on the user’s trusted device or a sensor node for his/her own profit. However, any such attempt to tamper with the PUF will change the behavior of the device and it will render the PUF useless. Due to this, during the execution of the Phases IV and V in the proposed scheme, the PUFs will not be able to produce the desired outputs $R_u = P_D(C_u)$ and $R_{Sn} = P_{Sn}(C_{Sn})$. Therefore, the gateway can comprehend such attempt of tempering, and accordingly, it will take necessary action. Also, PUFs are safe against cloning and a PUF cannot be recreated [19]. Therefore, the proposed scheme is resilient against cloning attack.

6) Resistance to Loss of Device Attacks: In the proposed scheme, only a legitimate user with his/her trusted device has the right to access the IWSN. Suppose the user’s trusted device is lost or stolen. Now, an adversary $A$ may attempt to impersonate as the legitimate user to get access to the IWSN. In this regard, when the device will ask $A$ to input his/her biometric thumb impression, $A$ cannot input the same biometric thumb impression $\beta_u$ into the device. Moreover, $A$ does not know the password $psw_u$. Eventually, the device cannot compute the same value of $\delta$ as stored in its internal memory and it will not be able to incorporate with $A$ to impersonate as a legal user. Next, assume that $A$ will try to perform side channel attacks in order to obtain all the information stored in the device. However, as we mentioned before, any such attempt will change the PUF behavior [18] and it will render the PUF useless. Therefore, the PUF will not be able to produce the desired output $R_u = P_D(C_u)$, which is essential to convince the gateway and get access of the IWSN. It then follows that the proposed scheme is resilient against lost/stolen device attacks.

V. PERFORMANCE ANALYSIS AND COMPARISON

In this section, we evaluate the performance of the proposed scheme in terms of security features, computation, communication, and storage costs. To manifest the advantages of the proposed scheme, we first compare our scheme with some of the recently proposed schemes, such as the schemes of Das et al. [11], Turkanovic et al. [15], Chang and Le [16], and Gope and Hwang [17]. From Table II, we can see that Das and Le’s scheme [16] cannot ensure user anonymity property (as discussed in Section 4-A). In Gope and Hwang’s scheme [17], the anonymity of the sensor nodes has not been considered. It is also worth noticing that Das et al.’s scheme [11] does not support the features $I_5$, $I_6$, $I_9$ and $I_{12}$-I$15$. Turkanovic et al.’s scheme [15] does not support the features $I_2$, $I_7$, $I_{12}$ and $I_{13}$. However, all the existing protocols in WSNs can ensure the physical security of the sensor nodes and user device. On the other hand, only the proposed scheme designed in this article can prevent several imperative attacks and fulfill the desirable security features.
Next, we compare the computation cost of the proposed scheme with the prior related schemes \([11], [15], [16], [17]\). Table [IV] shows the number of hash operations \(N_H\), modular exponentiation operations \(N_{Exp}\) and PUF operations \(N_P\) required by the proposed scheme and other schemes \([11], [15], [16], [17]\). The results in this table show that the computation cost of the proposed scheme is quite similar to that in the scheme \([17]\). To evaluate the performance of the proposed scheme in terms of the computation costs at the user and gateway, we first simulate the cryptographic operations used in the proposed scheme and other schemes \([11], [15], [16], [17]\) on an Ubuntu 12.04 virtual machine with an Intel Core i5-4300 dual-core 2.60 GHz CPU (operating as the gateway). To simulate the user’s device, we use a single core 798 MHz CPU and 256 MB of RAM. The simulation uses JCE library to evaluate the execution time of several cryptographic operations used in the proposed scheme and other schemes \([11], [15], [16], [17]\). Here, for hash operation we consider the SHA-256 \([25]\). In that case, each hash operation at the user device and the gateway takes 1.37 ms and 0.68 ms, respectively. Besides, for the PUF operation we consider the simulation of an 128-bit arbirter PUF circuit on the MSP430 micro-controller machine with 798 MHz CPU, where each PUF operation takes 0.43 ms. From Table [V] we can see that Chang-Le’s scheme \([16]\) takes more computation cost than other schemes, where each modular exponential operation takes 16.84 ms. On the other hand, Table [V] shows that although the communication cost of the proposed scheme is little higher than some of the schemes, the computation cost of the proposed scheme at the user and gateway is lower than the other schemes. Next, we consider the computation and the communication cost at the sensor node in the proposed scheme and other existing schemes \([11], [15], [16], [17]\). In this regard, we simulate the cryptographic operations used in the proposed scheme and other existing schemes \([11], [15], [16], [17]\) on a modular sensor board MSB-430 with the T1 MSP430 micro-controller and the temperature sensor- TMP36. Here, for the PUF operation we consider the simulation of an 128-bit arbirter PUF circuit on the MSP430 micro-controller machine. Based on the simulation results, the execution time of a hash operation (SHA-256), modular exponentiation operation, PUF operation are 1.37 ms, 16.84 ms and 0.43 ms, respectively. Table [V] shows that the proposed scheme takes only 6.34 ms to execute \(2N_P + 4N_H\) operations, which is significantly less than that for the scheme \([16]\) and other schemes. From Table [V] we can also see that in the proposed scheme a sensor node needs to bear less communication cost as compared to other schemes. Furthermore, since in the proposed scheme the sensor node does not require to store any secret credentials in its memory, the storage cost of the proposed scheme is nil for the sensor node point of view. Conclusively, from Tables [IV] and [V] we can argue that the performance of the proposed scheme is better than other schemes \([11], [15], [16], [17]\), and hence, it is more suitable for designing any secure IWSN as compared to other schemes.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Computation Cost (User)</th>
<th>Computation Cost (Gateway)</th>
<th>Communication Cost (User)</th>
<th>Communication Cost (Gateway)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Das et al. [11]</td>
<td>15.07 ms</td>
<td>7.48 ms</td>
<td>84 bytes</td>
<td>120 bytes</td>
</tr>
<tr>
<td>Turkcanovic et al.[15]</td>
<td>10.96 ms</td>
<td>11.96 ms</td>
<td>80 bytes</td>
<td>136 bytes</td>
</tr>
<tr>
<td>Chang-Le [16]</td>
<td>25.31 ms</td>
<td>12.64 ms</td>
<td>112 bytes</td>
<td>174 bytes</td>
</tr>
<tr>
<td>Gope-Hwang [17]</td>
<td>10.96 ms</td>
<td>6.12 ms</td>
<td>88 bytes</td>
<td>152 bytes</td>
</tr>
<tr>
<td>Our</td>
<td>10.8 ms</td>
<td>6.12 ms</td>
<td>88 bytes</td>
<td>160 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Computation Cost (in ms)</th>
<th>Communication Cost (in bytes)</th>
<th>Storage Cost (in bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Das et al. [11]</td>
<td>8.22</td>
<td>35</td>
<td>256</td>
</tr>
<tr>
<td>Turkcanovic et al.[15]</td>
<td>8.22</td>
<td>35</td>
<td>256</td>
</tr>
<tr>
<td>Chang-Le [16]</td>
<td>40.53</td>
<td>51</td>
<td>378</td>
</tr>
<tr>
<td>Gope-Hwang [17]</td>
<td>8.22</td>
<td>35</td>
<td>128</td>
</tr>
<tr>
<td>Our</td>
<td>6.34</td>
<td>32</td>
<td>Nil</td>
</tr>
</tbody>
</table>

VI. CONCLUSION

This article proposes a lightweight and physically secure mutual authentication protocol for IWSN. In the proposed protocol, we used lightweight cryptographic primitives, such as one-way hash function, physically uncloneable function and bitwise XOR operations. In the comparative summary, we demonstrated that the proposed scheme ensures several imperative security features and incurs lower computation, communication, and storage costs. Hence, the proposed scheme is more suitable for IWSN security in comparison to the other existing schemes.

REFERENCES


Prosanta Gope (M’18) received the PhD degree in Computer Science and Information Engineering from National Cheng Kung University (NCKU), Tainan, Taiwan, in 2015. He is currently working as a Lecturer in the Department of Computer Science (Cyber Security) at the University of Hull, United Kingdom. Prior to this, Dr. Gope was working as a Research Fellow in the Department of Computer Science at National University of Singapore (NUS). His research interests include lightweight authentication, authenticated encryption, access control system, security in mobile communication and cloud computing, lightweight security solutions for smart grid and hardware security of the IoT devices. He has authored over 50 peer-reviewed articles in several reputable international journals and conferences, and has four filed patents. He received the Distinguished Ph.D. Scholar Award in 2014 from the National Cheng Kung University, Tainan, Taiwan. He currently serves as an Associate Editor of the IEEE SENSORS JOURNAL, the SECURITY AND COMMUNICATION NETWORKS and the MOBILE INFORMATION SYSTEMS JOURNAL.

Ashok Kumar Das (M’17–SM’18) received the Ph.D. degree in computer science and engineering, the M.Tech. degree in computer science and data processing, and the M.Sc. degree in mathematics from IIT Kharagpur, India. He is currently an Associate Professor with the Center for Security, Theory and Applications, IIIT Hyderabad, India. His research interests include cryptography and network security. He has authored over 180 papers in international journals and conferences, including more than 155 reputed journal papers. He was a recipient of the Silver Medal from IIT Kharagpur.

Neeraj Kumar (M’16–SM’17) received the Ph.D. degree in computer science and engineering from Shri Mata Vaishno Devi University, Katra (J&amp;K), India, in 2009. He was a Post-Doctoral Research Fellow at Coventry University, Coventry, U.K. He is currently an Associate Professor with the Department of Computer Science and Engineering, Thapar University, Patiala, India. He has authored more than 200 technical research papers in the above areas.

Yongqiang Cheng received the Ph.D. degree from School of Engineering, Design and Technology, the University of Bradford, UK. He is currently a senior lecturer with the department of computer science and technology at the University of Hull, UK. His research interest includes digital healthcare technologies, embedded systems, control theory and applications, AI and data mining.