IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Two-Factor Authenticated Key Agreement Supporting Unlinkability in 5G-Integrated Wireless Sensor Networks

SOOYEON SHIN (Member, IEEE) AND TAEKYOUNG KWON (Member, IEEE)
Graduate School of Information, Yonsei University, Seoul, 03722, South Korea (e-mail: shinsy80@yonsei.ac.kr, taekyoung@yonsei.ac.kr)

Corresponding author: Taekyoung Kwon (e-mail: taekyoung@yonsei.ac.kr).

This work was partly supported by the National Research Foundation of Korea (NRF-2016-R1C1B2011095) and also partly supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2017-2016-0-00304) supervised by the IITP(Institute for Information & communications Technology Promotion).

ABSTRACT The integration of 5G networks and wireless sensor networks (WSNs) is critical in the new era of the Internet of Things (IoT), for a wide range of applications. However, despite the potential advantages of this integration, there are concerns about unforeseen security threats that may impact our daily lives. Authenticated key agreement is an essential security feature for secure communication between users and IoT devices and for protecting IoT applications from security threats. An IoT notion-based authentication and key agreement scheme was recently proposed for heterogeneous WSNs, claiming to provide user anonymity and mutual authentication, as well as the ability to withstand several types of attacks. In this paper, we examine several security weaknesses of the aforementioned scheme. Next, we design a network architecture suitable for the integration of 5G networks and WSNs. Based on the network architecture, we propose a two-factor authentication and key agreement scheme in 5G-integrated WSNs for the IoT that can resist various attacks, including those identified earlier, and that can preserve security requirements, including unlinkability. Lastly, we evaluate the security and performance of the proposed scheme and compare our scheme with other related schemes.

INDEX TERMS Two-Factor Authentication, Key Agreement, Password, Smart Card, Anonymity, Unlinkability, 5G Network, Wireless Sensor Networks, Internet of Things

I. INTRODUCTION

THE Internet of Things (IoT) is an emerging technology that connects a variety of devices, including smartphones, home appliances, sensors, and other network devices. This new technology can be applied in many application domains, such as smart homes (e.g., security, heating and lighting control), smart cities, healthcare (e.g., remote patient monitoring), and smart manufacturing (e.g., remote monitoring and control of manufacturing system). For the development of IoT applications, establishing an open, standardized network stack with protocols catering to the needs of the constrained devices is essential [1]. Moreover, because the IoT spans such a wide range of application domains, its deployment requires heterogeneous network connectivity [2].

Smartphones have played an important role in early IoT services, communicating using Wi-Fi and cellular network technologies. Cellular networks are considered a potential candidate for providing connectivity to IoT devices, owing to their mobility support, reliability, and ubiquitous deployment [3]. In particular, the fifth-generation networks (5G) currently under development are aiming to provide high speed (1 Gbps), low power, and low latency (1 ms or less). Hence 5G technology will accelerate the deployment of many IoT applications, demanding more ubiquity, more mobility, better performance and speed, and faster response times.

A wireless sensor network (WSN) consists of a large number of wireless, resource-constrained, small sensor nodes, deployed in an area of interest to monitor and collect physical or environmental conditions, such as light, temperature, pressure, motion, sound, or pollutants. WSNs play an important role in the IoT by supporting the sensing and collecting of environmental information. Thus, to successfully provide IoT applications, the integration of 5G networks and WSNs is required.

However, despite of the potential of this integration, it also exposes us further to security threats in our daily lives.

Hence, security and privacy are critical to protecting IoT applications from such attacks. Moreover, the heterogeneity of the networks can have a significant influence on the security of IoT applications [4], where resource-constrained sensor nodes must open a secure communication with more powerful devices. For example, in a smart home, home sensor nodes communicate with the user's smartphone. For secure communications between any parties, and to provide equivalent security levels for communications between diverse devices, optimal cryptography algorithms are essential. Furthermore, IoT devices require high-speed and efficient lightweight security.

In IoT, only legitimate users should be able to access authentic IoT devices (i.e., a gateway or sensor node) and a session key should be established between the user and the IoT device for secure data transmission. Therefore, mutual authentication with key agreement is an important requirement for the IoT. Because the IoT carries data that may contain personal privacy information (i.e., identity and position) and anyone can access another user's device, any information leaks may compromise users' privacy. In 5G-integrated WSNs for the IoT, anonymity is an important security aspect, because it protects the privacy of both users and the IoT devices such as sensor nodes. Anonymity typically refers to the state in which an individual's personal identity or personally identifiable information is not known publicly. The unlinkability of two or more items of interest, from an attacker's perspective, means that within the system, the attacker cannot identify whether these items are related. Pfitzmann and Hansen point out that unlinkability is a sufficient condition of anonymity, but not a necessary condition [5]. However, to remain completely anonymous, most users want strong anonymity [6], which requires unlinkability, where an attacker's examination of the pseudonym holder's message provides no new information about the holder's true name [7]. Thus, in order to properly protect user privacy, both anonymity and unlinkability should be considered.

A. RELATED WORK

In 2006, Wong et al. proposed a lightweight user authentication scheme [8] for WSNs based on XOR and hash operations. However, in 2009, Das showed that the scheme could not withstand a stolen verifier attack and an attack where many users were logged-in with the same ID and, thus, proposed a two-factor-based user authentication scheme [9] to resolve these issues. In his scheme, a password and a smart card are used as two factors to authenticate a user. However, in 2010, a number of researchers [10]–[13] pointed out security problems in Das's scheme, and proposed improvements to overcome. Then, Das et al. [14] in 2012 and Xue et al. [15] in 2013 individually presented user authentication and key agreement schemes for WSNs based on the use of smart cards.

Recently, in 2014, Turkanović et al. proposed a user authentication and key agreement scheme for heterogeneous ad-hoc WSNs, based on the IoT [16]. Their scheme is lightweight because it uses only simple operations, such as XOR and hash function. Through IoT, a random user can connect directly to a single sensor node from the WSN, and negotiate a session key with it without connecting to a gateway node. Unfortunately, the scheme was later proved to be vulnerable to multiple attacks, by Chang et al. [17], Farash et al. [18], Amin and Biswas [19], and Tai et al. [20].

In 2016, Chang et al. pointed out that Turkanović et al.'s scheme is susceptible to an impersonation attack with node capture, a stolen smart card attack, a sensor node spoofing attack, and a stolen verifier attack, as well as failing to ensure backward secrecy [17]. Chang et al. proposed a flexible authentication protocol using a smart card for WSNs that operates in two modes: a lightweight authentication scheme, as an improvement to that of Turkanović et al. scheme, and an advanced protocol based on elliptic curve cryptography (ECC), providing perfect forward secrecy.

At the same time, Farash et al. identified that Turkanović et al.'s scheme cannot resist a stolen smart card attack and a man-in-the-middle attack, and that it does not provide untraceability and forward/backward secrecy [18]. Based on their analysis, they proposed an improved user authentication and key agreement scheme for heterogeneous WSNs. However, Amin et al. found that Farash et al.'s scheme does not withstand a known session-specific temporary information attack, an offline password guessing attack using a stolen smart card attack, a new smart card issue attack, and a user impersonation attack. Furthermore, it does not preserve user anonymity and the secrecy of the secret key of the gateway node [21]. Amin et al. then presented an anonymous-preserving three-factor authenticated key exchange protocol for WSNs, in which a password, a smart card, and biometrics are used as three factors.

In 2016, Amin and Biswas proved that Turkanović et al.'s scheme does not prevent an offline identity-password guessing attack, a smart card theft attack, a user impersonation attack, and a sensor node impersonation attack, as well as providing an inefficient authentication phase [19]. As a solution, they proposed a secure lightweight scheme for user authentication and key agreement in multi-gateway based WSNs.

Most recently, in 2017, Tai et al. also showed that Turkanović et al.'s scheme does not ensure user anonymity, and that a session key established in the scheme can be leaked using compromised sensor nodes. To overcome these security flaws, they proposed an improvement to Turkanović et al.'s scheme. They claimed that their scheme ensures user anonymity and mutual authentication between all parties. However, we have found that Tai et al.'s scheme does not provide mutual authentication and sensor node anonymity and, furthermore that it is susceptible to a sensor node spoofing attack with sensor node capturing, a privileged-insider attack, and a session-specific temporary information attack [22]. We also find additional security weaknesses in the scheme, namely, being susceptible to stolen smart card and offline password guessing attacks and no securing user anonymity.

B. CONTRIBUTION

As shown in the section on related works, the existing studies on user authentication and key agreement for WSNs fail to satisfy desirable security features. In particular, most of the proposed schemes do not provide strong anonymity, referred to as unlinkability. In addition, they focus mainly on WSNs, which means their network architectures are not suitable for 5G-integrated WSNs for the IoT.

- We analyze the security of Tai et al.'s most recent user authentication and key agreement scheme for IoT-based ad hoc heterogeneous WSNs. We show that their scheme is vulnerable to several attacks including stolen smart card, offline password guessing, sensor node spoofing, privileged-insider, and session-specific temporary information attacks. We also show that Tai et al.'s scheme does not preserve user and sensor node anonymity, mutual authentication, and the secrecy of the secret key of the gateway node.
- We design a network architecture suitable for 5G-integrated WSNs for the IoT. Under the new network architecture, we propose a secure two-factor authentication and key agreement scheme that overcomes the aforementioned security weaknesses and preserves all the security features of Tai et al.'s scheme. Moreover, our proposed scheme withstands all known attacks and ensures unlinkability and, thus, strong anonymity.
- Using a security evaluation, we show that our proposed scheme can resist many attacks, including those that would compromise Tai et al.'s scheme. In addition, we compare the security features of our proposed scheme with those of other related schemes.
- Through a performance evaluation, we compare the performance of our proposed scheme with other related schemes in terms of their computational cost, communication cost, and storage cost.

C. ORGANIZATION OF THE PAPER

Section 2 briefly reviews Tai et al.'s scheme, after which we discuss its security weaknesses in Section 3. Section 4 addresses the proposed authentication and key agreement scheme with unlinkability, based on the new network design. The security evaluation of the proposed scheme is discussed in Section 5. Section 6 presents the performance comparison with other related schemes. Finally, we conclude the paper in Section 7.

II. REVIEW OF Tai et al.'s SCHEME

In this section, we briefly review Tai et al.'s scheme [20], which consists of six phases: pre-deployment, registration, login, authentication, password-change, and dynamic node addition. The registration phase is divided further into two sub-phases: user registration and sensor node registration. The notation used in Tai et al.'s scheme is given in Table 1.

TABLE 1. Notation for Tai et al.'s scheme.

Notation	Description
SC	Smart card
U_i	User
S_j	Sensor node
GWN	Gateway node
ID_i	Identity of U_i
PW_i	Password of U_i
SID_j	Identity of S_j
X_{GWN}, X_U	Secure password keys known only to the GWN
X_{GWN-i}	Shared secure password between GWN and U_i
X_{GWN-j}	Shared secure password between GWN and S_j
K_i	Random number generated by U_i
K_j	Random number generated by S_j
SK	Agreed session key of the user and sensor node
T	Timestamp
ΔT	Time interval for the allowed transmission delay
$h(\cdot)$	Cryptographic one-way hash function
$\|\|$	Concatenation operation
$\oplus$	Bitwise XOR operation

A. PRE-DEPLOYMENT PHASE

A network administrator predefines a pair of an identifier SID_j and a password X_{GWN-j} for each sensor node S_j, where $1 \leq j \leq m$ and m is the number of sensor nodes in the WSN. X_{GWN-j} is generated randomly and stored in S_j's memory. For GWN, the administrator predefines two secure password keys X_{GWM} and X_U, known only to GWN and stored in GWN's memory. In addition, GWN stores SID_j and X_{GWN-j} for each sensor node S_j.

B. USER REGISTRATION PHASE

On demand, a user U_i initiates the user registration phase, after which he/she can access any sensor node.

(1) U_i selects her/his identity ID_i and password PW_i and sends a registration request $\langle ID_i, PW_i \rangle$ to GWN through a secure channel.

(2) GWN randomly selects a password key X_{GWN-i} for U_i and stores it with ID_i in its memory. It then computes $f_i = h(ID_i||X_{GWN}), x_i = h(ID_i||PW_i||X_{GWN-i})$, and $e_i = h(PW_i) \oplus X_U$.

(3) GWN chooses an SC and writes $\langle f_i, x_i, e_i, X_{GWN-i}, h(\cdot) \rangle$ into the SC's memory. Then, GWN sends it to U_i through a secure channel.

C. SENSOR NODE REGISTRATION PHASE

The sensor node registration phase is conducted after the deployment of sensor nodes in the target field.

(1) S_j computes $MP_j = h(SID_j||T_1||X_{GWN-j})$, where T_1 is the S_j's current timestamp, and sends the registration request $\langle SID_j, MP_j, T_1 \rangle$ to GWN.

(2) Upon receiving the registration request, GWN checks $|T_1 - T_C| < \Delta T$, where T_C is the current timestamp of GWN. If this fails, GWN transmits a rejection message to S_j.

(3) Otherwise, GWN searches the corresponding X_{GWN-j} using the received SID_j and computes

$MP_j^* = h(SID_j||T_1||X_{GWN-j})$. GWN then verifies $MP_j^* \overset{?}{=} MP_j$, If this fails, GWN terminates this phase and sends a rejection message to S_j. Otherwise, GWN computes $f_j = h(SID_j||X_{GWN}), x_j = h(T_2||X_{GWN-j}), e_j = f_j \oplus x_j$, and $z_j = h(f_j||e_j||T_2||X_{GWN-j})$, where T_2 is the current timestamp of GWN. GWN sends a response message $\langle e_j, z_j, T_2 \rangle$ to S_j.

(4) On obtaining GWN's response, S_j checks $|T_2 - T_C| < \Delta T$. If this fails, S_j terminates this phase and sends a request to GWN to re-execute the phase. Otherwise, S_j computes $x_j^* = h(T_2||X_{GWN-j}), f_j^* = e_j \oplus x_j^*$, and $z_j^* = h(f_j^*||e_j||T_2||X_{GWN-j})$. S_j then verifies $z_j^* \overset{?}{=} z_j$. If this fails, S_j asks GWN to resend $\langle e_j, z_j \rangle$. If S_j still cannot verify the resent $\langle e_j, z_j \rangle$ successfully, this phase is re-executed immediately. If $z_j^* = z_j$, S_j confirms that $f_j^* = f_j$, and stores f_j^* in its memory.

D. LOGIN PHASE

In order to access information from the WSN, U_i needs to log in.

(1) U_i inserts her/his SC into the card reader and inputs ID_i and PW_i.

(2) SC computes $x_i^* = h(ID_i||PW_i||X_{GWN-i})$ using the inputted ID_i and PW_i and X_{GWN-i} stored in its memory. SC then verifies $x_i^* \overset{?}{=} x_i$. If this fails, this phase is terminated. If U_i inputs the wrong password more than three times, SC is locked immediately. If $x_i^* = x_i$, SC chooses a random number K_i, and computes $MI_i = h(T_1||h(PW_i) \oplus e_i) \oplus ID_i, Z_i = K_i \oplus h(T_1||X_{GWN-i})$, and $N_i = h(MI_i||ID_i||K_i||f_i||T_1||X_{GWN-i})$, where T_1 is U_i's current timestamp.

(3) U_i selects a sensor node S_j, and sends an authentication request $\langle MI_i, Z_i, N_i, T_1 \rangle$ to S_j through an open channel.

E. AUTHENTICATION PHASE

In this phase, U_i and S_j can authenticate each other and negotiate a session key to be shared between them, with the help of GWN.

(1) Upon receiving the authentication request from U_i, S_j checks $|T_1 - T_C| < \Delta T$. If this fails, S_j terminates this phase and sends a rejection message to U_i. Otherwise, S_j selects a random number K_j, and computes $A_j = h(N_i||T_2||X_{GWN-j}) \oplus K_j$ and $B_j = h(A_j||K_j||T_2||f_j)$, where T_2 is the current timestamp of S_j. S_j then sends $\langle MI_i, Z_i, N_i, T_1, SID_j, A_j, B_j, T_2 \rangle$ to GWN via an open channel.

(2) On obtaining $\langle MI_i, Z_i, N_i, T_1, SID_j, A_j, B_j, T_2 \rangle$ from S_j, GWN checks $|T_2 - T_C| < \Delta T$. If this fails, GWN terminates this phase and sends a rejection message to S_j. Otherwise, GWN searches the corresponding X_{GWN-j} using the received SID_j, and computes $K_j^* = h(N_i||T_2||X_{GWN-j}) \oplus A_j, f_j^* = h(SID_j||X_{GWN})$, and $B_j^* = h(A_j||K_j^*||T_2||f_j^*)$. GWN then checks $B_j^* \overset{?}{=} B_j$. If this fails, GWN aborts all further actions and sends a rejection message to S_j. Otherwise, GWN successfully authenticates S_j.

(3) GWN computes $ID_i^* = MI_i \oplus h(T_1||X_U)$ and searches the corresponding X_{GWN-i} using ID_i^*. GWN computes $f_i^* = h(ID_i^*||X_{GWN}), K_i^* = Z_i \oplus h(T_1||X_{GWN-i})$, and $N_i^* = h(MI_i||ID_i^*||K_i^*||f_i^*||T_1||X_{GWN-i})$. GWN then checks $N_i^* \overset{?}{=} N_i$. If this fails, GWN aborts all further actions and sends a rejection message indicating that U_i is illegal to S_j. Otherwise, GWN confirms that U_i and S_j are legal.

(4) GWN then computes $R_i = K_j^* \oplus h(T_3||N_i||f_i^*||X_{GWN-i})$, $R_j = K_i^* \oplus h(T_3||B_j||f_j^*||X_{GWN-j})$, and $F_{ij} = h(T_1||T_2||T_3||R_i||K_i^*||K_j^*)$, where T_3 is the GWN's current timestamp. GWN then sends $\langle R_i, R_j, F_{ij}, T_1, T_2, T_3 \rangle$ to S_j via an open channel.

(5) Upon receiving $\langle R_i, R_j, F_{ij}, T_1, T_2, T_3 \rangle$ from GWN, S_j checks $|T_3 - T_C| < \Delta T$. If this fails, all further actions are aborted and S_j sends a rejection message to GWN and U_i. Otherwise, S_j computes $K_i^* = R_j \oplus h(T_3||B_j||f_j^*||X_{GWN-j})$ and $F_{ij}^* = h(T_1||T_2||T_3||R_i||K_i^*||K_j)$. S_j then checks $F_{ij}^* \overset{?}{=} F_{ij}$. If this fails, S_j asks GNW to resend the message. If S_j still cannot verify the resent message successfully, all further actions are aborted and S_j sends a rejection message to GWN and U_i. Otherwise, if $F_{ij}^* = F_{ij}$, S_j computes the session key $SK = h(K_i^* \oplus K_j)$ shared with U_i and $R_{ij} = h(T_1||T_2||T_3||T_4||K_i^*||K_j||SK)$, where T_4 is S_j's current timestamp, and sends $\langle R_i, R_{ij}, T_1, T_2, T_3, T_4 \rangle$ to U_i via an open channel.

(6) On obtaining $\langle R_i, R_{ij}, T_1, T_2, T_3, T_4 \rangle$ from S_j, U_i checks $|T_4 - T_C| < \Delta T$. If this fails, U_i aborts all further actions and sends a rejection message to S_j. Otherwise, SC computes $K_j^* = R_i \oplus h(T_3||N_i||f_i||X_{GWN-i})$, the session key $SK^* = h(K_i \oplus K_j^*)$ shared with S_j, and $R_{ij}^* = h(T_1||T_2||T_3||T_4||K_i||K_j^*||SK^*)$. It then checks $R_{ij}^* \overset{?}{=} R_{ij}$. If this fails, U_i asks S_j to resend the message $\langle R_i, R_{ij} T_1, T_2, T_3, T_4 \rangle$. If the resent message is still not verified successfully, U_i terminates this phase and sends a rejection message to S_j. Otherwise, if $R_{ij}^* = R_{ij}$, U_i confirms that GWN and S_j are legal, and that the computed SK^* is equal to S_j's SK.

III. SECURITY WEAKNESSES OF Tai et al.'s SCHEME

In this section, we discuss the security weaknesses of Tai et al.'s scheme, and show that an adversary can mount different types of attacks on the scheme.

A. INSECURITY OF THE SECRET KEY OF THE GATEWAY NODE

In Tai et al.'s scheme, an authorized user U_i can extract the hashed value of secret key X_U, because it is easy for U_i to compute $X_U = e_i \oplus h(PW_i)$ using its own password PW_i and the retrieved information e_i from his/her smart card SC_i. Thus, the secret key X_U of the gateway node, which is used for every user, is not secure.

B. STOLEN SMART CARD AND OFFLINE PASSWORD GUESSING ATTACKS

Although a smart card is usually equipped with tamper resistant hardware, by launching power analysis attacks [23], an adversary can extract all sensitive information stored in its memory. Thus, we assume that if a user's smart card is stolen or lost, an adversary can obtain the information (i.e., $\langle f_i, x_i, e_i,\ X_{GWN-i}, h(\cdot)\rangle$ in Tai et al.'s scheme) from the card.

In Section III-A, we described how an authorized user U_j, who wants to act as an adversary, can know X_U. After extracting X_U from his/her own smart card, and using the smart card stolen from the legal user U_i, adversary U_j can guess PW_i^* and compute $e_i^* = h(PW_i^*) \oplus X_U$. If $e_i^* = e_i$ holds, then the adversary can obtain the actual password. Thus, Tai et al.'s scheme is susceptible to stolen smart card and offline password guessing attacks.

C. INSECURITY OF USER ANONYMITY

As in the case of an offline password guessing attack, if an authorized user U_j, who acts as an adversary, knows X_U, the adversary can compute another legitimate user's identity. The adversary U_j intercepts a legitimate user U_i's login message $\langle MI_i, Z_i, N_i, T_1\rangle$ during protocol execution, where $MI_i = h(T_1||h(PW_i) \oplus e_i) \oplus ID_i$, $Z_i = K_i \oplus h(T_1||X_{GWN-i})$, and $N_i = h(MI_i||ID_i||K_i||f_i||T_1||X_{GWN-i})$. Then, the adversary U_j can easily compute $ID_i' = MI_i \oplus h(T_1||X_U)$, which is the original identity of U_i. Therefore, the user-anonymity property can be broken easily.

D. NO SENSOR NODE ANONYMITY

In the authentication phase, the sensor node S_j sends the request message $\langle MI_i, Z_i, N_i, T_1, SID_j, A_j, B_j, T_2\rangle$ to the gateway node GWN via an insecure channel. Clearly, if an adversary intercepts this request message from the insecure channel, he/she can obtain S_j's identity SID_j. Thus, the anonymity of sensor nodes is not preserved in Tai et al.'s scheme.

E. LACK OF MUTUAL AUTHENTICATION

In a user authentication and key agreement scheme, mutual authentication of all involved parties is essential. Tai et al. stated that their scheme provides mutual authentication between any two of a gateway node, a sensor node, and a user. However, in their scheme, it is not possible for a user to authenticate a sensor node.

In Tai et al.'s scheme, U_i should authenticate the chosen sensor node S_j with the help of GWN. However, in the last step of the authentication phase, S_j delivers only one value R_i received from GWN to U_i, and R_i does not include any information to authenticate S_j. Here, U_i only utilizes this value to extract K_j^* in order to compute SK, which will be shared with S_j in this session. Furthermore, U_i verifies only the session key through $R_{ij}^* \overset{?}{=} R_{ij}$, and does not verify the source authentication of the message $\langle R_i, R_{ij}T_1, T_2, T_3, T_4\rangle$. In other words, U_i does not check whether the message is truly from the selected S_j with SID_j herself/himself during the login phase. Thus, an adversary can launch the sensor node spoofing attack described in the next section, because of the lack of mutual authentication.

F. SENSOR NODE SPOOFING ATTACK WITH SENSOR NODE CAPTURING

An adversary can capture or compromise a sensor node and extract important information stored in its memory because WSNs are installed in unattended or hostile environments. In Tai et al.'s scheme, if an adversary compromises one sensor node, he/she can masquerade any non-compromised and legitimate sensor node to which a user is trying to log in.

Suppose that an adversary compromises a sensor node S_j and obtains SID_j, X_{GWN-j}, and f_j from the compromised S_j. When a user U_i wants to log into the sensor node S_k, the adversary performs the following steps to launch a sensor node spoofing attack:

(1) When U_i sends $\langle MI_i, Z_i, N_i, T_1\rangle$ to S_k, the adversary intercepts that message and randomly selects K_j'. Then, the adversary computes $A_j' = h(N_i||T_2'||X_{GWN-j}) \oplus K_j'$ and $B_j' = h(A_j'||K_j'||T_2'||f_j)$ using S_j's compromised parameters X_{GWN-j} and f_j and the current timestamp T_2'. The adversary sends $\langle MI_i, Z_i, N_i, T_1, SID_j, A_j', B_j', T_2'\rangle$ to GWN.

(2) Upon receiving the above message from S_j, GWN performs the verification process as per step (2) in the authentication phase. Because M_i, Z_i, and N_i are not bound to S_k, GWN cannot identify whether these were actually sent to S_k, and not to S_j. In addition, the adversary used valid parameters of S_j to compute A_j' and B_j' and, thus, GWN trusts that the received message is valid and that is originated from the sensor node S_j, chosen by U_i. GWN then computes R_i, R_j, and F_{ij} and sends $\langle R_i, R_j, F_{ij}, T_1, T_2', T_3\rangle$ to the adversary, who is now impersonating the sensor node S_j.

(3) On receiving $\langle R_i, R_j, F_{ij}, T_1, T_2', T_3\rangle$ from GWN, the adversary obtains K_i^* using the compromised parameters f_j and X_{GWN-j}, and computes $SK' = h(K_i^* \oplus K_j')$ and $R_{ij} = h(T_1||T_2'||T_3||K_i^*||K_j'||SK')$. Finally, the adversary sends $\langle R_i, R_{ij}T_1, T_2', T_3, T_4'\rangle$, where T_4' is the current timestamp of the adversary, to U_i.

(4) After receiving $\langle R_i, R_{ij}T_1, T_2', T_3, T_4\rangle$ from S_j, U_i verifies the timestamp T_4' and obtains $K_j^* = R_i \oplus h(T_3||N_i||f_i||X_{GWN-i})$. U_i then will successfully computes $SK^* = h(K_i||K_j^*)$ and verifies $R_{ij}^* \overset{?}{=} R_{ij}$.

Thus, the adversary has succeeded in masquerading as the sensor node S_k.

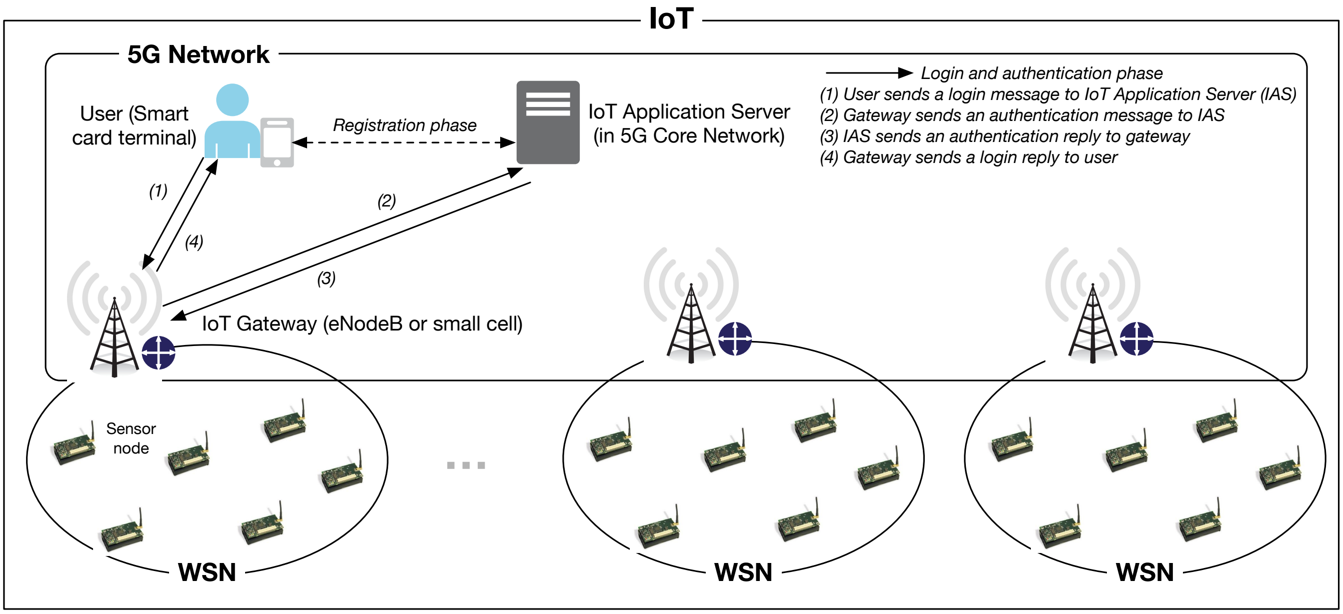

IoT
5G Network
User (Smart card terminal)
Registration phase
IoT Application Server (in 5G Core Network)
Login and authentication phase
(1) User sends a login message to IoT Application Server (IAS)
(2) Gateway sends an authentication message to IAS
(3) IAS sends an authentication reply to gateway
(4) Gateway sends a login reply to user
(1)
(4)
(2)
(3)
IoT Gateway (eNodeB or small cell)
Sensor node
WSN
WSN
WSN

FIGURE 1. Proposed scheme architecture of 5G-integrated WSNs for the IoT.

G. PRIVILEGED-INSIDER ATTACK

In Tai et al.'s scheme, a user U_i sends the plaintext password to GWN in the registration phase. If U_i submits the same password used in other systems to GWN, GWN can use the password to impersonate the victim user when accessing other systems. Thus, Tai et al.'s scheme is susceptible to a privileged-insider attack.

H. SESSION-SPECIFIC TEMPORARY INFORMATION ATTACK

Canetti and Krawczyk [24] introduced a session-specific temporary information attack. This attack implies that if the specific information generated temporarily for a session is leaked, the session key established in the that session is no longer secure.

In Tai et al's scheme, U_i and S_j compute the session key based on the temporary random numbers K_i and K_j generated by U_i and S_j, respectively. If these two temporary numbers K_i and K_j are leaked, then an adversary can compute the session key $SK = h(K_i \oplus K_j)$ established between U_i and S_j. Thus, the security of the session key is compromised in the event of a leakage of session-specific temporary information.

IV. OUR PROPOSED SCHEME

In this section, we propose a two-factor authentication and key agreement scheme in 5G-integrated WSNs for the IoT, that overcomes the aforementioned security weaknesses identified in Tai et al.'s scheme.

As mentioned in Section I-B, we design a network architecture suitable for user authentication and key agreement in 5G-integrated WSNs for the IoT. Figure 1 describes the network architecture. The proposed model consists of three types of entities: the user (U_i), IoT gateway (GW_j), and an IoT application server (IAS). After registration and mutual authentication, for IoT services, U_i can obtain real-time data from GW_j via a 5G network. The main tasks of GW_j are to collect real-time data from sensor nodes in the WSN, and to deliver them to the authenticated user via the 5G network. Thus, as an IoT gateway, GW_j can be located in eNodeB or a small cell in the 5G access network. Here, IAS is responsible for providing a registration facility for U_i, as well as proper IoT services, based on the underlying WSNs, to the authenticated user via the 5G network. Thus, IAS can be located in the 5G core network.

Our proposed scheme consists of four phases: system setup, user registration, login and authentication, and password change. We use the additional notation for the proposed scheme listed in Table 2.

TABLE 2. Notation for the proposed scheme.

Notation	Description
IAS	IoT Application Server
GW_j	IoT Gateway
MID_i	Masked identity of U_i
MPW_i	Masked password of U_i
$GWID_j$	Identity of GW_j
X_U	Secret of IAS used for authenticating users
X_{GW}	Secret of IAS used for authenticating gateways
PU_i^k	One-time pseudonym of U_i used in the $k-th$ authentication.
PGW_j^k	One-time pseudonym of GW_j used in the $k-th$ authentication.
r_i^k	Random number generated by U_i for updating PU_i^k
s_i^k	Random number generated by GW_j for updating PGW_j^k

A. SYSTEM SETUP PHASE

Before the deployment of gateways and sensor nodes in a target field, this phase is executed by the IoT application server (IAS) in offline mode. This phase is described below.

(1) IAS selects a master secret X_U for users, which is known only to IAS.

(2) IAS chooses an identity $GWID_j$ and randomly selects the first one-time pseudonym PGW_j^1 for every gateway GW_j, where $1 \leq j \leq m$ and m is the number of gateways.

(3) IAS selects a master secret X_{GW} for gateways, which is known only to IAS, and computes $v_j = h(GWID_j||X_{GW})$ and $w_j^1 = h(PGW_j^1)$, which are different for each gateway.

(4) IAS finally embeds $\langle GWID_j, PGW_j^1, v_j, w_j^1\rangle$ in the memory of GW_j in a secure manner.

B. USER REGISTRATION PHASE

When a new user U_i, where $1 \leq i \leq n$ and n is the number of users, wants to obtain an IoT application service based on WSNs, U_i must first register with the IAS. This phase is described in Figure 2 and below.

(1) The new user U_i chooses the desired identity ID_i and password PW_i, and selects a random number a_i. U_i then computes $MID_i = h(a_i||ID_i)$ and $MPW_i = h(a_i||PW_i)$ and sends the masked identity and password as the registration request, $\langle MID_i, MPW_i\rangle$, to IAS via a secure channel.

(2) After receiving the U_i's registration request, IAS selects the first one-time pseudonym PU_i^1 for U_i and computes $x_i = h(MID_i||MPW_i)$, $y_i = h(MID_i||X_U)$, $d_i = y_i \oplus h(MPW_i||x_i)$, $e_i = PU_i^1 \oplus h(MPW_i||y_i)$, $z_i^1 = h(PU_i^1||X_U)$, and $g_i = z_i^1 \oplus h(MPW_i||x_i||y_i)$. Then, IAS issues a new smart card SC_i for U_i after storing $\{c_i, d_i, e_i, g_i\}$ in the memory of SC_i through a secure channel. Finally, IAS stores $\{MID_i, PU_i^1\}$ in its memory.

(3) Upon receiving the smart card SC_i, U_i computes $b_i = a_i \oplus h(ID_i||PW_i)$ and stores $\{b_i\}$ in SC_i. Finally, SC_i contains $\{c_i, d_i, e_i, g_i, b_i\}$.

C. LOGIN AND AUTHENTICATION PHASE

The login and authentication phase is executed through a public channel whenever U_i wants to gain access to a WSN using his/her ID_i, PW_i, and SC_i. Figure 3 illustrates the login and authentication phase of the proposed scheme. To achieve mutual authentication and session key agreement, this phase executes in several steps as follows.

(1) U_i inserts own SC_i, and inputs identity ID_i and password PW_i into a terminal (i.e., a smart card reader). SC_i computes $a_i = b_i \oplus h(ID_i||PW_i)$, $MID_i = h(a_i||ID_i)$, and $MPW_i = h(a_i||PW_i)$. Then, SC_i computes $x_i^* = h(MID_i||MPW_i)$ and checks whether x_i^* matches with the stored x_i. If it matches, SC_i has ensured that U_i has provided the correct ID_i and PW_i.

(2) SC_i randomly chooses numbers K_i and r_i^1. The random number K_i is used to generate a session key, and r_i^k is used to update the next one-time pseudonym PU_i^{k+1}. SC_i then computes $y_i^* = d_i \oplus h(MPW_i||x_i^*)$, $PU_i^{1*} = e_i \oplus h(MPW_i||y_i^*)$, $z_i^{1*} = g_i \oplus h(MPW_i||x_i^*||y_i^*)$, $M_1 = h(z_i^{1*}||T_1) \oplus MID_i$, $M_2 = K_i \oplus h(y_i^*||T_1)$, $M_3 = r_i^1 \oplus h(y_i^*||z_i^*||T_1)$, and $M_4 = h(M_1||M_2||M_3||K_i||r_i^1||GWID_j||T_1)$, where T_1 is the current timestamp of U_i, and $GWID_j$ is the identity of the gateway GW_j where the user is currently located. SC_i sends a login message $\langle PU_i^{1*}, M_1, M_2, M_3, M_4, T_1\rangle$ to GW_j.

(3) Upon receiving the login message, GW_j first checks whether $|T_1 - T_C| < \Delta T$. If the verification succeeds, GW_j chooses random numbers K_j and s_j^1. The random number K_j is used to generate a session key, and s_j^k is used to update the next one-time pseudonym PGW_j^{k+1}. Using the stored values v_j, w_j^1, and PGW_j^1, GW_j then computes $M_5 = h(w_j||T_2) \oplus GWID_j$, $M_6 = K_j \oplus h(v_j||T_2)$, $M_7 = s_j^1 \oplus h(v_j||w_j||T_2)$, and $M_8 = h(M_5||M_6||M_7||K_j||s_j^1||PU_i^1||T_2)$, where T_2 is the current timestamp of GW_j. In order to authenticate each other, with the help of IAS, GW_j sends an authentication message $PGW_j^1, M_5, M_6, M_7, M_8$, and T_2, including the values received from U_i, $PU_i^{1*}, M_1, M_2, M_3, M_4$, and T_1, to IAS through a public channel.

(4) On receiving the message from GW_j, IAS first checks whether $|T_2 - T_C| < \Delta T$. If the verification does not hold, IAS aborts any further action and sends a rejection message to GW_j. If the verification holds, IAS extracts $GWID_j$ from the database using PGW_j^1 and computes $w_j^{1*} = h(PGW_j^1||X_{GW})$ and $GWID_j^* = M_5 \oplus h(w_j^{1*}||T_2)$. IAS then checks if the retrieved $GWID_j^*$ is equal to the searched $GWID_j$, based on the pseudonym. If the verification does not hold, IAS terminates the scheme because GW_j is not proved to be legitimate. Furthermore, IAS sends a rejection message to U_i and GW_j.

(5) If the above verification holds, IAS has successfully authenticated GW_j and starts with authenticating U_i. First, IAS extracts MID_i from the database using PU_i^1, computes $z_i^{1*} = h(PU_i^1||X_U)$ and $MID_j^* = M_1 \oplus h(z_i^{1*}||T_1)$, and checks if $MID_i^* = MID_i$. If this holds, U_i is authenticated to IAS. Otherwise, IAS aborts the session and sends a rejection message to U_i and GW_j.

(6) After successfully authenticating both GW_j and U_i, IAS derives random values that will be used to generate a session key and to update the one-time pseudonyms. IAS computes $v_j^* = h(GWID_j||X_{GW})$, $K_j^* = M_6 \oplus h(v_j^*||T_2)$,

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/ACCESS.2018.2796539, IEEE Access

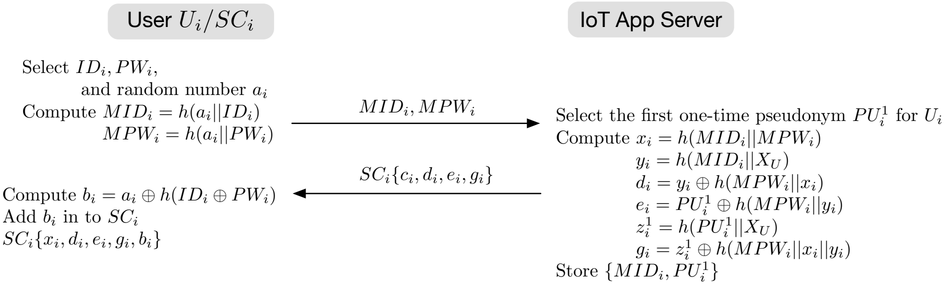

User U_i/SC_i
IoT App Server
Select ID_i, PW_i, and random number a_i
Compute $MID_i = h(a_i||ID_i)$
$MPW_i = h(a_i||PW_i)$
MID_i, MPW_i
Select the first one-time pseudonym PU_i^1 for U_i
Compute $x_i = h(MID_i||MPW_i)$
$y_i = h(MID_i||X_U)$
$d_i = y_i \oplus h(MPW_i||x_i)$
$e_i = PU_i^1 \oplus h(MPW_i||y_i)$
$z_i^1 = h(PU_i^1||X_U)$
$g_i = z_i^1 \oplus h(MPW_i||x_i||y_i)$
Store $\{MID_i, PU_i^1\}$
$SC_i\{c_i, d_i, e_i, g_i\}$
Compute $b_i = a_i \oplus h(ID_i \oplus PW_i)$
Add b_i in to SC_i
$SC_i\{x_i, d_i, e_i, g_i, b_i\}$

FIGURE 2. User registration phase of our proposed scheme.

$s_j^{1*} = M_7 \oplus h(v_j^*||w_j^{1*}||T_2)$, and $M_8^* = h(M_5||M_6||M_7||K_j^*||s_j^{1*}||PU_i^1||T_2)$, and checks the correctness of the received M_8. If the latter is not valid, IAS aborts the session. Otherwise, IAS computes $y_i^* = h(MID_i||X_U)$, $K_i^* = M_2 \oplus h(y_i^*||T_1)$, $r_i^{1*} = M_3 \oplus h(y_i^*||z_i^{1*}||T_1)$, and $M_4^* = h(M_1||M_2||M_3||K_i^*||r_i^{1*}||GWID_j||T_1)$, and checks the correctness of the received M_4. If the latter is not valid, IAS aborts the session and sends a rejection message to U_i and GW_j. Otherwise, IAS continues to the next step.

(7) IAS computes $NK_i = h(K_i^*||MID_i)$ and $NK_j = h(K_j^*||GWID_j)$, which are used to compute the session key between U_i and GW_j. IAS then computes $PU_i^2 = h(PU_i^1||r_i^{1*})$, $PGW_j^2 = h(PGW_j^1||s_j^{1*})$, $z_i^2 = h(PU_i^2||X_U)$, and $w_j^{2*} = h(PGW_j^2||X_{GW})$ to update the one-time pseudonyms PU_i^2 and PGW_j^2, and their confirmation values z_i^2 and w_j^2 for the next authentication of U_i and GW_j, respectively.

(8) Finally, IAS computes $M_9 = NK_j \oplus h(GWID_j||y_i^*||T_3)$, $M_{10} = z_i^2 \oplus h(y_i^*||z_i^{1*}||T_3)$, $M_{11} = h(M_9||M_{10}||NK_i||NK_j||PU_i^2||T_1||T_2||T_3||y_i^*)$, $M_{12} = NK_i \oplus h(PU_i^1||v_j^*||T_3)$, $M_{13} = w_j^2 \oplus h(v_j^*||w_j^{1*}||T_3)$, and $M_{14} = h(M_{11}||M_{12}||NK_i||NK_j||PGW_j^2||T_1||T_2||T_3||v_i^*)$. Then, it sends the authentication reply $\langle M_9, \cdots, M_{14}, T_1, T_2, T_3\rangle$ to GW_j via a public channel, and updates database PU_i^1 to PU_i^2 for U_i, and PGW_j^1 to PGW_j^2 for GW_j.

(9) Upon receipt of the authentication reply, GW_j first verifies whether $|T_3 - T_C| < \Delta T$. If the verification does not hold, GW_j aborts any further action and sends a rejection message to IAS and U_i. Otherwise, GW_j computes $NK_i^* = M_{12} \oplus h(PU_i^1||v_j||T_3)$, $NK_j = (K_j||GWID_j)$, $PGW_j^2 = h(PGW_j^1||s_j^1)$, $w_j^2 = M_{13} \oplus h(v_j||w_j^1||T_3)$, and $M_{14}^* = h(M_{11}||M_{12}||NK_i^*||NK_j||PGW_j^2||T_1||T_2||T_3||v_j)$, and checks whether the newly computed value M_{14}^* is equal to the received M_{14}. If the verification holds, GW_j believes that IAS and U_i are authentic. Otherwise, GW_j aborts any further action and sends a rejection message to IAS and U_i.

(10) After authenticating both U_i and IAS, GW_j establishes a session key $SK_{U_iGW_j} = h(NK_i^*||NK_j)$, computes $M_{15} = h(PU_i^1||GWID_j||SK_{U_iS_j}^*||T_1||T_2||T_3||T_4)$, and sends the login reply $\langle M_9, M_{10}, M_{11}, M_{15}, T_1, T_2, T_3, T_4\rangle$ to U_i via a public channel. Lastly, GW_j updates its memory PGW_j^1, w_j^1 to PGW_j^2, w_j^2, respectively.

(11) On receiving the login reply from GW_j, U_i checks whether $|T_4 - T_C| < \Delta T$ holds. If this is incorrect, U_i aborts the session and sends a rejection message to GW_j. Otherwise, U_i computes $NK_j^* = M_9 \oplus h(GWID_j||y_i^*||T_3)$, $NK_i = h(K_i||MID_i)$, $PU_i^2 = h(PU_i^1||r_i^1)$, $z_i^{2*} = M_{10} \oplus h(y_i^*||z_i^{1*}||T_3)$, and $M_{11}^* = h(M_9||M_{10}||NK_i||NK_j^*||PU_i^2||T_1||T_2||T_3||y_i^*)$, and checks if $M_{11}^* = M_{11}$. If this fails, U_i aborts the session and sends a rejection message to GW_j. If it matches, IAS is confirmed to be authentic. U_i computes a session key $SK_{U_iGW_j} = h(NK_i \oplus NK_j^*)$ and $M_{15}^* = h(PU_i^1||GWID_j||SK_{U_iGW_j}||T_1||T_2||T_3||T_4)$, and then verifies the legitimacy of GW_j by checking if $M_{15}^* = M_{15}$. If this fails, U_i terminates the session and sends a rejection message to GW_j. If it matches, U_i believes the authenticity of GW_j and updates e_i, g_i of the memory of its own smart card SC_i using PU_i^2, z_i^{2*}, respectively. Finally, U_i successfully ends the login and authentication phase, and both U_i and GW_j can communicate securely using the derived session key $SK_{U_iGW_j}$.

D. PASSWORD CHANGE PHASE

In the proposed scheme, a user can freely change his/her password without the help of an IoT application server. This phase contains the following steps.

(1) U_i inserts his/her smart card SC_i into a terminal, and inputs identity ID_i and his/her old password PW_i^{old}.

(2) SC_i computes $a_i^* = b_i \oplus h(ID_i||PW_i^{old})$, $MID_i^* = h(a_i||ID_i)$, $MPW_i^* = h(a_i||PW_i^{old})$, and $x_i^* = h(MID_i||MPW_i)$. Then, SC_i compares the computed x_i^* with the stored x_i in its memory. If these do not match, this means that U_i has inputted his/her old password PW_i^{old} incorrectly and, hence, SC_i terminates the password change phase immediately. Other-

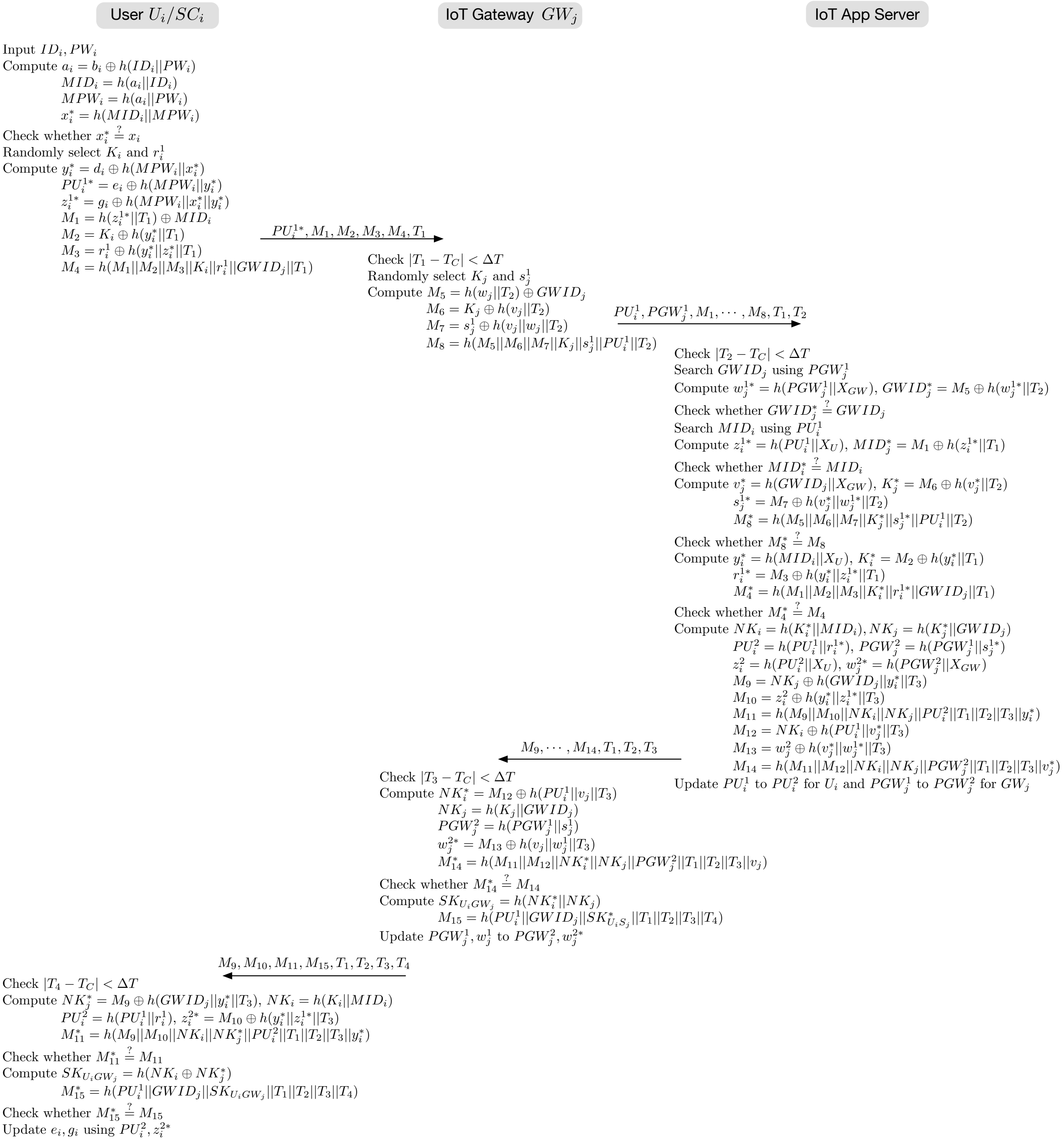

User U_i/SC_i
IoT Gateway GW_j
IoT App Server
Input ID_i, PW_i
Compute $a_i = b_i \oplus h(ID_i||PW_i)$
$MID_i = h(a_i||ID_i)$
$MPW_i = h(a_i||PW_i)$
$x_i^* = h(MID_i||MPW_i)$
Check whether $x_i^* \overset{?}{=} x_i$
Randomly select K_i and r_i^1
Compute $y_i^* = d_i \oplus h(MPW_i||x_i^*)$
$PU_i^{1*} = e_i \oplus h(MPW_i||y_i^*)$
$z_i^{1*} = g_i \oplus h(MPW_i||x_i^*||y_i^*)$
$M_1 = h(z_i^{1*}||T_1) \oplus MID_i$
$M_2 = K_i \oplus h(y_i^*||T_1)$
$M_3 = r_i^1 \oplus h(y_i^*||z_i^*||T_1)$
$M_4 = h(M_1||M_2||M_3||K_i||r_i^1||GWID_j||T_1)$
$PU_i^{1*}, M_1, M_2, M_3, M_4, T_1$
Check $|T_1 - T_C| < \Delta T$
Randomly select K_j and s_j^1
Compute $M_5 = h(w_j||T_2) \oplus GWID_j$
$M_6 = K_j \oplus h(v_j||T_2)$
$M_7 = s_j^1 \oplus h(v_j||w_j||T_2)$
$M_8 = h(M_5||M_6||M_7||K_j||s_j^1||PU_i^1||T_2)$
$PU_i^1, PGW_j^1, M_1, \cdots, M_8, T_1, T_2$
Check $|T_2 - T_C| < \Delta T$
Search $GWID_j$ using PGW_j^1
Compute $w_j^{1*} = h(PGW_j^1||X_{GW})$, $GWID_j^* = M_5 \oplus h(w_j^{1*}||T_2)$
Check whether $GWID_j^* \overset{?}{=} GWID_j$
Search MID_i using PU_i^1
Compute $z_i^{1*} = h(PU_i^1||X_U)$, $MID_j^* = M_1 \oplus h(z_i^{1*}||T_1)$
Check whether $MID_i^* \overset{?}{=} MID_i$
Compute $v_j^* = h(GWID_j||X_{GW})$, $K_j^* = M_6 \oplus h(v_j^*||T_2)$
$s_j^{1*} = M_7 \oplus h(v_j^*||w_j^{1*}||T_2)$
$M_8^* = h(M_5||M_6||M_7||K_j^*||s_j^{1*}||PU_i^1||T_2)$
Check whether $M_8^* \overset{?}{=} M_8$
Compute $y_i^* = h(MID_i||X_U)$, $K_i^* = M_2 \oplus h(y_i^*||T_1)$
$r_i^{1*} = M_3 \oplus h(y_i^*||z_i^{1*}||T_1)$
$M_4^* = h(M_1||M_2||M_3||K_i^*||r_i^{1*}||GWID_j||T_1)$
Check whether $M_4^* \overset{?}{=} M_4$
Compute $NK_i = h(K_i^*||MID_i), NK_j = h(K_j^*||GWID_j)$
$PU_i^2 = h(PU_i^1||r_i^{1*})$, $PGW_j^2 = h(PGW_j^1||s_j^{1*})$
$z_i^2 = h(PU_i^2||X_U)$, $w_j^{2*} = h(PGW_j^2||X_{GW})$
$M_9 = NK_j \oplus h(GWID_j||y_i^*||T_3)$
$M_{10} = z_i^2 \oplus h(y_i^*||z_i^{1*}||T_3)$
$M_{11} = h(M_9||M_{10}||NK_i||NK_j||PU_i^2||T_1||T_2||T_3||y_i^*)$
$M_{12} = NK_i \oplus h(PU_i^1||v_j^*||T_3)$
$M_{13} = w_j^2 \oplus h(v_j^*||w_j^{1*}||T_3)$
$M_{14} = h(M_{11}||M_{12}||NK_i||NK_j||PGW_j^2||T_1||T_2||T_3||v_j^*)$
Update PU_i^1 to PU_i^2 for U_i and PGW_j^1 to PGW_j^2 for GW_j
$M_9, \cdots, M_{14}, T_1, T_2, T_3$
Check $|T_3 - T_C| < \Delta T$
Compute $NK_i^* = M_{12} \oplus h(PU_i^1||v_j||T_3)$
$NK_j = h(K_j||GWID_j)$
$PGW_j^2 = h(PGW_j^1||s_j^1)$
$w_j^{2*} = M_{13} \oplus h(v_j||w_j^1||T_3)$
$M_{14}^* = h(M_{11}||M_{12}||NK_i^*||NK_j||PGW_j^2||T_1||T_2||T_3||v_j)$
Check whether $M_{14}^* \overset{?}{=} M_{14}$
Compute $SK_{U_iGW_j} = h(NK_i^*||NK_j)$
$M_{15} = h(PU_i^1||GWID_j||SK_{U_iS_j}^*||T_1||T_2||T_3||T_4)$
Update PGW_j^1, w_j^1 to PGW_j^2, w_j^{2*}
$M_9, M_{10}, M_{11}, M_{15}, T_1, T_2, T_3, T_4$
Check $|T_4 - T_C| < \Delta T$
Compute $NK_j^* = M_9 \oplus h(GWID_j||y_i^*||T_3)$, $NK_i = h(K_i||MID_i)$
$PU_i^2 = h(PU_i^1||r_i^1)$, $z_i^{2*} = M_{10} \oplus h(y_i^*||z_i^{1*}||T_3)$
$M_{11}^* = h(M_9||M_{10}||NK_i||NK_j^*||PU_i^2||T_1||T_2||T_3||y_i^*)$
Check whether $M_{11}^* \overset{?}{=} M_{11}$
Compute $SK_{U_iGW_j} = h(NK_i \oplus NK_j^*)$
$M_{15}^* = h(PU_i^1||GWID_j||SK_{U_iGW_j}||T_1||T_2||T_3||T_4)$
Check whether $M_{15}^* \overset{?}{=} M_{15}$
Update e_i, g_i using PU_i^2, z_i^{2*}

FIGURE 3. Login and authentication phase of our proposed scheme.

wise, SC_i demands a new password of U_i.

(3) Using the new password PW_i^{new}, SC_i computes the new masked password $MPW_i' = h(a_i^*||PW_i^{new})$. Then, SC_i computes $y_i^* = d_i \oplus h(MPW_i^*||x_i^*)$, $PU_i^{k*} = e_i \oplus h(MPW_i^*||y_i^*)$, and $z_i^{k*} = g_i \oplus h(MPW_i^*||x_i^*||y_i^*)$, where k is an index indicating the next authentication number.

(4) SC_i replaces x_i, d_i, e_i, g_i, and b_i with $x_i' = h(MID_i||MPW_i')$, $d_i' = y_i^* \oplus h(MPW_i'||x_i')$, $e_i = PU_i^{k*} \oplus h(MPW_i'||y_i^*)$, $g_i' = z_i^{k*} \oplus h(MPW_i'||x_i^*||y_i^*)$, and $b_i' = a_i \oplus h(ID_i||PW_i^{new})$, respectively, in its memory.

TABLE 3. Security feature comparison of the proposed scheme with other related schemes.

Features	Proposed scheme	Tai et al. [20]	Chang et al. ($\mathcal{P}_1$) [17]	Farash et al. [18]	Turkanović et al. [16]	Xue et al. [15]	Das et al. [14]
Mutual Authentication	YES	NO	YES	NO	NO	NO	NO
Session Key Agreement	YES	YES	YES	YES	YES	YES	YES
User Anonymity	YES	NO	YES	NO	NO	NO	NO
Unlinkability	YES	NO	NO	NO	NO	NO	NO
Sensor Node Anonymity (GW in our scheme)	YES	NO	NO	YES	NO	NO	NO
Resilience to a							
Offline Password Guessing Attack	YES	NO	YES	NO	NO	NO	NO
Priviledged-Insider Attack	YES	NO	NO	NO	NO	NO	NO
Impersonation Attack	YES	NO	YES	YES	NO	NO	NO
Stolen Verifier Attack	YES	YES	YES	NO	YES	YES	YES
Stolen Smart Card Attack	YES	NO	YES	YES	NO	NO	NO
Session-specific Temporary Information Attack	YES	NO	NO	YES	NO	NO	YES

V. SECURITY EVALUATION OF THE PROPOSED SCHEME

Here, we present a security evaluation of our proposed scheme by showing how it satisfies the security requirements and is secure against various known attacks. We also compare the security of the proposed scheme with other related schemes, in Table 3.

A. MUTUAL AUTHENTICATION

On receiving the authentication message, including the login message of U_i from GW_j, IAS uses the pseudonyms PU_i^1 and PGW_j^1 to search for identities MID_i and $GWID_j$, respectively, in the database. This is because an adversary cannot generate legal $z_i^{1*} = h(PU_i^1||X_U)$ and $w_j^{1*} = h(PGW_j^1||X_{GW})$ without knowing IAS's secret X_U and X_{GW}, even if he/she knows PU_i^1 and PGW_j^1. IAS also retrieves MID_i^* and $GWID_j^*$ from the received messages M_1 and M_5 by computing z_i^{1*} and w_j^{1*}, and verifies the legitimacy of U_i and GW_j using $MID_i^* = MID_i$ and $GWID_j^* = GWID_j$, respectively.

On the other hand, on receiving the authentication reply from IAS, using PU_i^1 in the login message of U_i, GW_j retrieves NK_i^* from the message and computes NK_j and PGW_j^2 itself. Then, GW_j computes $M_{14}^* = h(M_{11}||M_{12}||NK_i^*||NK_j||PGW_j^2||T_1||T_2||T_3||v_j)$ to verify the legitimacy of IAS and U_i using $M_{14}^* = M_{14}$. This is because only a legitimate IAS can retrieve the correct value $v_j = h(GWID_j||X_{GW})$ of GW_j and can compute M_{12} using both v_j and the same pseudonym PU_i^1 of U_i, who requested the login, and provide these values to GW_j.

On receiving the login reply from GW_j, using $GWID_j$ of the gateway requested access in the login message M_4, U_i retrieves NK_j^* from the reply and computes NK_i and PU_i^2 itself. Then, U_i computes $M_{11}^* = h(M_9||M_{10}||NK_i||NK_j^*||PU_i^2||T_1||T_2||T_3||y_i^*)$ to verify the legitimacy of IAS and GW_j using $M_{11}^* = M_{11}$. This is because only a legitimate IAS can retrieve the correct values $y_i^* = h(MID_i||X_U)$ and $z_i^{1*} = h(PU_i^1||X_U)$ of U_i, compute M_{11} using y_i, z_i^1, and the same identity $GWID_j$ of GW_i which the user wants to access, and provide these values to U_i. Therefore, our proposed scheme provides mutual authentication.

B. SECURE SESSION KEY AGREEMENT AND RESILIENCE TO A SESSION-SPECIFIC TEMPORARY INFORMATION ATTACK

Secure session key agreement is essential to providing confidentiality of future communication between a user and a gateway. In the proposed scheme, at the end of the authentication phase, U_i and GW_j agree on the session key $SK_{U_iGW_j} = h(NK_i||NK_j)$. On receiving the authentication and login replies from IAS, GW_j and U_i retrieve $NK_i = h(K_i||MID_i)$ and $NK_j = h(K_j||GWID_j)$, respectively, from the replies. Then, GW_j and U_i individually compute NK_j and NK_i using randomly selected values K_j and K_i and their identities $GWID_j$ and MID_i, respectively. After mutual authentication, they compute $SK_{U_iGW_j}$. Both randomly selected values K_i and K_j, from U_i and GW_j, respectively, are always masked by the secret values y_i and z_i. Even if an adversary knows K_i and K_j, he/she cannot compute $SK_{U_iGW_j}$ without knowing U_i's masked identity MID_i and the identity of GW_j. In addition to, the adversary cannot retrieve NK_j, NK_i from the login and authentication replies $M_9 = NK_j \oplus h(GWID_j||y_i^*||T_3)$ and $M_{12} = NK_i \oplus h(PU_i^1||v_j^*||T_3)$ without knowing y_i and z_i, respectively. As a result, our proposed scheme achieves secure key agreement, and a leakage of the session-specific temporary information K_i and K_j does not affect the security of the established session key.

C. ANONYMITY WITH UNLINKABILITY

From the registration phase, user U_i always uses the masked identity $MID_i = h(a_i||ID_i)$ instead of the real identity ID_i. In the login authentication phase, U_i and GW_j hide MID_i and $GWID_j$ by computing masked versions $M_1 = h(z_i^{1*}||T_1) \oplus MID_i$ and $M_5 = h(w_j||T_2) \oplus GWID_j$, respectively. Because all messages in the login and authentication phase are transmitted via a public channel, an adversary could simply eavesdrop on the channel. If an adversary eavesdrops on the communication between all parties in the login

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/ACCESS.2018.2796539, IEEE Access

and authentication phase, he/she cannot detect the identities MID_i and GW_j from the intercepted messages.

To enable IAS to identify each U_i and GW_j, the proposed scheme utilizes the one-time pseudonyms PU_i^k and PGW_j^k, which are different for each login and authentication session. During the k-th login and authentication, these pseudonyms PU_i^k and PGW_j^k are updated individually for the $k+1$-th login and authentication using random numbers r_i^k and s_j^k selected by U_i and GW_j, respectively. IAS is also able to update the pseudonyms using r_i^{*k} and s_j^{*k}, retrieved from the received message. Then, U_i and GW_j can verify that the updated pseudonyms of IAS are properly synchronized using $M_{11}^* = M_{11}$ and $M_{14}^* = M_{14}$, respectively. Moreover, all other messages are also different for each login and authentication session due to the use of current timestamps. Thus, an adversary cannot identify users between different login and authentication sessions by capturing all messages of those sessions. In addition to, an adversary cannot determine which gateway is involved in different login and authentication sessions. In conclusion, our proposed scheme achieves user and gateway anonymity with unlinkability.

D. RESILIENCE TO STOLEN SMART CARD, OFFLINE IDENTITY GUESSING, AND OFFLINE PASSWORD GUESSING ATTACKS

During the execution of the proposed scheme, a user's identity ID_i and password PW_i are protected by a random value a_i and the non-invertible cryptographic one-way hash function. Thus, an adversary cannot extract the user's identity and password. However, the adversary may attempt to extract the stored information of U_i and guess ID_i and PW_i, based on the extracted information.

Suppose that an adversary steals the smart card of a legal user U_i. By launching power analysis attacks [23], the adversary can then extract the stored information $\{x_i, d_i, e_i, g_i, b_i\}$ in the smart card SC_i of the user U_i, where $x_i = h(MID_i||MPW_i)$, $d_i = y_i \oplus h(MPW_i||x_i)$, $e_i = PU_i^1 \oplus h(MPW_i||y_i)$, $g_i = z_i^1 \oplus h(MPW_i||x_i||y_i)$, and $b_i = a_i \oplus h(ID_i \oplus PW_i)$. Because both ID_i and PW_i in x_i are well protected by the non-invertible cryptographic one-way hash function, these are unknown to the adversary. If the adversary tries to guess either an identity or password, he/she has to guess two parameters at the same time, which is infeasible in polynomial time. Furthermore, except of b_i, all other values are computed using the masked identity MID_i and password MPW_i with a random value a_i, instead of ID_i and PW. Therefore, the proposed scheme is secure against stolen smart card, offline identity guessing, and offline password guessing attacks.

E. RESILIENCE TO A PRIVILEGED-INSIDER ATTACK

A strong password policy and a multi-factor authentication system can make it difficult for a user to remember passwords on multiple accounts [25]. Thus, it is common practice for users to reuse passwords on multiple accounts [26], [27]. In such situations, a privileged-insider, such as the system administrator or IoT application server in the proposed scheme, can misuse or disclose the user's passwords, resulting in a user impersonation on other application systems. A priviledged-insider attack can occur when a user sends her/his password to the system administrator in plaintext form [28].

During the registration phase of the proposed scheme, U_i submits the masked password MPW_i instead of the plaintext password PW_i to IAS via a secure channel, where $MPW_i = h(a_i||PW_i)$. The privileged-insider IAS of our scheme cannot extract the original password PW_i from MPW_i owing to the non-invertible cryptographic one-way hash function. Hence, the insider cannot use the user's password to access other systems. Therefore, the proposed scheme can withstand a privileged-insider attack.

F. RESILIENCE TO A STOLEN VERIFIER ATTACK

In general, the system administrator or IoT application server stores some information related to users for use during the authentication phase. This information may be stolen by an adversary to launch attacks, including a user impersonation attack. In our scheme, IAS does not maintain any user-specific information (i.e., ID_i and PW_i), other than the masked identity MID_i and one-time pseudonym PU_i^k. Thus, the proposed scheme is safe against a stolen verifier attack.

G. RESILIENCE TO AN IMPERSONATION ATTACK

Suppose an adversary obtains a legitimate user U_i's smart card SC_i, extracts the stored data $\{x_i, d_i, e_i, g_i, b_i\}$, and intercepts all messages from the previous authentication session. In order to impersonate the user, the adversary should produce a legal login message $\langle PU_i^{1*}, M_1, M_2, M_3, M_4, T_1 \rangle$. The adversary must possess the values $\{MID_i, PU_i^k, x_i, y_i, z_i\}$ to produce the legal message. In particular, to prove the legitimacy of U_i, $M_1 = h(z_i^{1*}||T_1) \oplus MID_i$ and $M_2 = K_i \oplus h(y_i^*||T_1)$ are important. To compute M_1 and M_2, the adversary needs to compute the values $z_i^{1*} = g_i \oplus h(MPW_i||x_i^*||y_i^*)$ and $y_i^* = d_i \oplus h(MPW_i||x_i^*)$, as well as $x_i^* = h(MID_i||MPW_i)$. However, without either U_i's password PW_i or the smart card SC_i, the adversary cannot compute these values. Thus, the proposed scheme is able to resist a user impersonation attack.

VI. PERFORMANCE EVALUATION OF THE PROPOSED SCHEME

We evaluate the performance of the proposed scheme and compare it with other related schemes in terms of various features, such as the computational cost, communication cost, and storage cost.

A. COMPUTATIONAL COST ANALYSIS

Our proposed scheme only uses a hash function and XOR operation, which are lightweight compared with other operations, such as symmetric-key encryption/decryption and

TABLE 4. Computational cost comparison of the proposed scheme with other related schemes.

Scheme	Proposed scheme	Tai et al. [20]	Chang et al. ($\mathcal{P}_1$) [17]	Farash et al. [18]	Turkanović et al. [16]	Xue et al. [15]	Das et al. [14]
User	$18T_H$	$4T_H$	$7T_H$	$11T_H$	$7T_H$	$7T_H$	$4T_H$+$1T_{E/D}$
Sensor node (GW in our scheme)	$11T_H$	$5T_H$	$5T_H$	$7T_H$	$5T_H$	$6T_H$	$3T_H$+$1T_{E/D}$
Gateway node (IAS in our scheme)	$24T_H$	$10T_H$	$8T_H$	$14T_H$	$7T_H$	$13T_H$	$3T_H$+$3T_{E/D}$
Total computation complexity	$53T_H$	$19T_H$	$20T_H$	$32T_H$	$19T_H$	$26T_H$	$10T_H$+$5T_{E/D}$
Total running time	0.0212 ms	0.0038 ms	0.008 ms	0.0128 ms	0.0038 ms	0.0104 ms	0.6555 ms

T_h: Time complexity of computing the one-way hash function; $T_{E/D}$: Time complexity of computing the symmetric encryption/decryption.

TABLE 5. Communication cost comparison of the proposed scheme with other related schemes.

Scheme	Proposed scheme	Tai et al. [20]	Chang et al. ($\mathcal{P}_1$) [17]	Farash et al. [18]	Turkanović et al. [16]	Xue et al. [15]	Das et al. [14]
User	6 (8)	4 (6)	4 (3)	4 (5)	5 (6)	6 (4)	7 (-)
Sensor node (GW in our scheme)	20 (15)	14 (1)	9 (8)	13 (9)	8 (11)	- (9)	
Gateway node (IAS in our scheme)	9 (12)	6 (8)	4 (6)	5 (8)	6 (8)	5 (10)	9 (7)
Total number of sent values	35	24	17	22	19	19	15
Total length of sent values	520 bytes	384 bytes	272 bytes	352 bytes	304 bytes	304 bytes	240 bytes

x (y): the number of values contained in the sent messages (the number of values contained in the received messages)

TABLE 6. Storage cost comparison of the proposed scheme with other related schemes.

Scheme	Proposed scheme	Tai et al. [20]	Chang et al. ($\mathcal{P}_1$) [17]	Farash et al. [18]	Turkanović et al. [16]	Xue et al. [15]	Das et al. [14]
User (SC)	768 bits	640 bits	512 bits	640 bits	768 bits	640 bits	768+(256×CH*) bits
Sensor node (GW in our scheme)	640 bits	512 bits	384 bits	640 bits	640 bits	512 bits	384 bits

CH*: the total number of all cluster heads in the WSN

public-key cryptographic functions. We assume that the running time of symmetric-key encryption/decryption is $T_{e/d} \approx 0.1303$ ms and the running time of the hash function is $T_h \approx 0.0004$ ms, based on the experimental results of [29].

In Table 4, we summarize the computational cost (computation complexity) and running time of the proposed scheme and of existing schemes in [14]–[18], [20] for the user, gateway node (IoT application server in the proposed scheme), and sensor node (gateway in the proposed scheme). With the exception of Das et al.'s scheme, our proposed scheme has a higher computational cost and running time than those of the other schemes. With only 0.084 ms to 0.174 ms added to the running time, our scheme provides all the security features, including unlinkability, and is resilient to various known attacks, as shown in Table 3. In addition to, all entities in our proposed scheme are more powerful devices than the sensor nodes in other schemes, because the proposed scheme has a different network model to the other schemes.

B. COMMUNICATION COST ANALYSIS

The communication costs of U_i, GW_j, and IAS of our scheme and other schemes in [14]–[18], [20] are given in Table 5. We assume that the lengths of the identity, password, random number, and output of the hash function are each 128 bits (16 bytes). In the propose scheme, U_i transmits 96 bytes, GW_j transmits 320 bytes, and IAS transmits 144 bytes. Therefore, the total transmission costs of U_i, GW_j, and IAS are 520 bytes. The communication costs of U_i and IAS do not differ greatly from the communication costs in other related schemes, whereas the communication cost of GW_j is relatively high compared to the communication cost of the sensor node in other schemes. However, 320 bytes is not a large value for GW_j because the gateways in the proposed scheme have sufficient resources, unlike sensor nodes.

C. STORAGE COST ANALYSIS

Here, we analyze the storage cost in terms of memory capacity of SC and the sensor node or gateway node in our scheme. In other words, we calculate the total length of the parameters, including the hash function $h(\cdot)$, in bits, that a smart card and a sensor or gateway node need to store in their memory. For convenience, we assume that all parameters and the hash function are 128 bits in length.

In Table 6, we present a smart card storage cost comparison of the proposed scheme and other related existing schemes in [14]–[18], [20]. In Das et al.'s scheme [14], a smart card saves the identities and keys for all cluster heads, where CH* denotes the number of cluster heads. The storage cost of the proposed scheme for the smart card and the sensor node is almost equal to that of other schemes, while providing more security features and being resilient to more attacks.

VII. CONCLUSION

In this paper, we reviewed Tai et al.'s scheme and demonstrated that it is vulnerable to a stolen smart card attack, offline password guessing attack, sensor node spoofing attack, privileged-insider attack, and session-specific temporary information attack. We further showed that Tai et al.'s scheme does not preserve user and sensor node anonymity, mutual authentication, and the secrecy of the secret key of the gateway node. We have designed a network architecture suitable for 5G-integrated WSNs for the IoT. Based on this network architecture, we have proposed a secure two-factor authentication and key agreement scheme with unlinkability. We evaluated the security of the proposed scheme and compared it with other related schemes. The results show that the proposed scheme is secure against various known attacks, and that it satisfies all security features, including unlinkability, required for secure user authentication and key agreement. We also evaluated the performance of the proposed scheme in terms of its computational cost, communication cost, and storage cost, which we then compared with those of other related schemes. The evaluation results of security and performance show that our scheme provides better safety without significantly different performance from other schemes, and performance results are expected to improve because the gateway performs better than the sensor node in 5G-integrated WSNs for the IoT.

In the future work, we will measure the performance of the proposed scheme by implementing and conducting experiments using actual devices on 5G-integrated WSNs for the IoT (e.g., smart phones and sensor motes) and, will improve the proposed scheme based on the experimental results.

REFERENCES

[1] S. Thombre, R. U. Islam, K. Andersson, and M. S. Hossain, "IP based wireless sensor networks: Performance analysis using simulations and experiments," Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), vol. 7, no. 3, pp. 53–76, September 2016.

[2] S. K. Lee, M. Bae, and H. Kim, "Future of IoT networks: A survey," Applied Sciences, vol. 7, no. 10, 2017.

[3] H. Shariatmadari, R. Ratasuk, S. Iraji, A. Laya, T. Taleb, R. Jäntti, and A. Ghosh, "Machine-type communications: current status and future perspectives toward 5g systems," IEEE Communications Magazine, vol. 53, no. 9, pp. 10–17, September 2015.

[4] R. Roman, P. Najera, and J. Lopez, "Securing the internet of things," Computer, vol. 44, no. 9, pp. 51–58, Sept 2011.

[5] A. Pfitzmann and M. Köhntopp, Anonymity, Unobservability, and Pseudonymity — A Proposal for Terminology. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 1–9.

[6] T. Kwon, "Privacy preservation with x.509 standard certificates," Information Sciences, vol. 181, no. 13, pp. 2906 – 2921, 2011.

[7] D. G. Post, "Pooling intellectual capital: Thoughts on anonymity, pseudonymity, and limited liability in cyberspace," University of Chicago Legal Forum, vol. 1996, Iss. 1, Article 5., 1996.

[8] K. H. M. Wong, Y. Zheng, J. Cao, and S. Wang, "A dynamic user authentication scheme for wireless sensor networks," in IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'06), vol. 1, June 2006.

[9] M. L. Das, "Two-factor user authentication in wireless sensor networks," IEEE Transactions on Wireless Communications, vol. 8, no. 3, pp. 1086–1090, March 2009.

[10] 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, March 2010.

[11] T.-H. Chen and W.-K. Shih, "A robust mutual authentication protocol for wireless sensor networks," ETRI Journal, vol. 32, no. 5, pp. 704–712, 2010.

[12] H. F. Huang, Y. F. Chang, and C. H. Liu, "Enhancement of two-factor user authentication in wireless sensor networks," in Proc. of the 6th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP'10), Darmstadt, Germany. IEEE, October 2010, pp. 27–30.

[13] D. He, Y. Gao, S. Chan, C. Chen, and J. Bu, "An enhanced two-factor user authentication scheme in wireless sensor networks," Ad Hoc & Sensor Wireless Networks, vol. 10, pp. 361 – 371, 2010.

[14] A. K. Das, P. Sharma, S. Chatterjee, and J. K. Sing, "A dynamic password-based user authentication scheme for hierarchical wireless sensor networks," Journal of Network and Computer Applications, vol. 35, no. 5, pp. 1646 – 1656, 2012.

[15] K. Xue, C. Ma, P. Hong, and R. Ding, "A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks," Journal of Network and Computer Applications, vol. 36, no. 1, pp. 316 – 323, January 2013.

[16] 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 Networks, vol. 20, pp. 96 – 112, September 2014.

[17] C. C. Chang and H. D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," IEEE Transactions on Wireless Communications, vol. 15, no. 1, pp. 357–366, January 2016.

[18] 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 Networks, vol. 36, pp. 152 – 176, January 2016.

[19] R. Amin and G. Biswas, "A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks," Ad Hoc Networks, vol. 36, pp. 58 – 80, January 2016.

[20] W.-L. Tai, Y.-F. Chang, and W.-H. Li, "An IoT notion–based authentication and key agreement scheme ensuring user anonymity for heterogeneous ad hoc wireless sensor networks," Journal of Information Security and Applications, vol. 34, pp. 133 – 141, June 2017.

[21] R. Amin, S. H. Islam, G. Biswas, M. K. Khan, L. Leng, and N. Kumar, "Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks," Computer Networks, vol. 101, no. Supplement C, pp. 42 – 62, 2016.

[22] S. Shin and T. Kwon, "Cryptanalysis of the iot notion-based authentication and key agreement scheme for wireless sensor networks," Research Briefs on Information & Communication Technology Evolution (ReBICTE), vol. 3, no. 12, November 2017.

[23] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," IEEE Transactions on Computers, vol. 51, no. 5, pp. 541–552, May 2002.

[24] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in Proc. of the International Conference on the Theory and Application of Cryptographic Techniques Innsbruck (Eurocrypt'01), Austria, LNCS, vol. 2045. Springer-Verlag, May 2001, pp. 453–474.

[25] A. Kim, G. Han, and S.-H. Seo, "Secure and usable bio-passwords based on confidence interval," Journal of Internet Services and Information Security (JISIS), vol. 7, no. 1, pp. 14–27, February 2017.

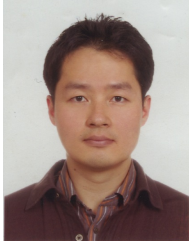
[26] E. Stobert and R. Biddle, "The password life cycle: User behaviour in managing passwords," in The Proc. of the Symposium On Usable Privacy and Security (SOUPS'14), Menlo Park, CA. USENIX Association, July 2014, pp. 243–255.

[27] A. Das, J. Bonneau, M. Caesar, N. Borisov, and X. Wang, "The tangled web of password reuse," in Proc. of the Network and Distributed System Security Symposium (NDSS'14), vol. 14. Internet Society, 2014, pp. 23–26.

[28] S. Kumari, M. K. Khan, and M. Atiquzzaman, "User authentication schemes for wireless sensor networks: A review," Ad Hoc Networks, vol. 27, pp. 159 – 194, April 2015.

[29] L. Xu and F. Wu, "Cryptanalysis and improvement of a user authentication scheme preserving uniqueness and anonymity for connected health care," Journal of Medical Systems, vol. 39, no. 2, p. 10, Jan 2015.

SOOYEON SHIN received her B.S., M.S., and Ph.D. degrees in computer science and engineering from Sejong University, Seoul, Korea, in 2004, 2006, and 2012, respectively. From 2012 to 2013, she was a postdoctoral researcher at Sejong University. In 2013, she joined Yonsei University, Seoul, Korea, to continue her postdoctoral research. Her current research interests include cryptographic protocol, network security, usable security, and human-computer interaction.



TAEKYOUNG KWON received his B.S., M.S., and Ph.D. degrees in computer science from Yonsei University, Seoul, Korea, in 1992, 1995, and 1999, respectively. He is currently a professor of information security at Yonsei University, Seoul, Korea. From 1999 to 2000, he was a postdoctoral researcher at the University of California, Berkeley, USA. From 2001 to 2013, he was a professor of computer engineering at Sejong University, Seoul, Korea. His research interests include authentication, cryptographic protocol, software security, usable security, and human-computer interaction.

• • •