RESEARCH ARTICLE

A Temporal Credential-Based Mutual Authentication with Multiple-Password Scheme for Wireless Sensor Networks

Xin Liu, Ruisheng Zhang*, Qidong Liu

School of information Science & Engineering, Lanzhou University, Lanzhou, China

* zhangrs@lzu.edu.cn

Abstract

Wireless sensor networks (WSNs), which consist of a large number of sensor nodes, have become among the most important technologies in numerous fields, such as environmental monitoring, military surveillance, control systems in nuclear reactors, vehicle safety systems, and medical monitoring. The most serious drawback for the widespread application of WSNs is the lack of security. Given the resource limitation of WSNs, traditional security schemes are unsuitable. Approaches toward withstanding related attacks with small overhead have thus recently been studied by many researchers. Numerous studies have focused on the authentication scheme for WSNs, but most of these works cannot achieve the security performance and overhead perfectly. Nam et al. proposed a two-factor authentication scheme with lightweight sensor computation for WSNs. In this paper, we review this scheme, emphasize its drawbacks, and propose a temporal credential-based mutual authentication with a multiple-password scheme for WSNs. Our scheme uses multiple passwords to achieve three-factor security performance and generate a session key between user and sensor nodes. The security analysis phase shows that our scheme can withstand related attacks, including a lost password threat, and the comparison phase shows that our scheme involves a relatively small overhead. In the comparison of the overhead phase, the result indicates that more than 95% of the overhead is composed of communication and not computation overhead. Therefore, the result motivates us to pay further attention to communication overhead than computation overhead in future research.

OPEN ACCESS

Citation: Liu X, Zhang R, Liu Q (2017) A Temporal Credential-Based Mutual Authentication with Multiple-Password Scheme for Wireless Sensor Networks. PLoS ONE 12(1): e0170657. doi:10.1371/journal.pone.0170657

Editor: Yongtang Shi, Nankai University, CHINA

Received: March 16, 2016

Accepted: January 9, 2017

Published: January 30, 2017

Data Availability Statement: All relevant data are within the paper and its Supporting Information files.

Funding: The authors received no specific funding for this work.

Competing Interests: The authors have declared that no competing interests exist.

Introduction

With the development of microelectronic, computer, and wireless communication techniques, multifunctional sensor nodes with small consumption have rapidly developed [1]. As a result, the Internet of Things has become increasingly popular. Wireless sensor networks (WSNs), which consist of a large number of sensor nodes (SNs), are widely used in various application fields, such as, environmental monitoring, military surveillance, nuclear-reactor control systems, vehicle safety systems, and medical monitoring [2, 3]. Although WSNs perform important functions in numerous application fields, the drawbacks of the network are evident. First, WSNs are often deployed in unattended environments [4] or enemy-controlled environments. Therefore, the networks are easily manipulated. Second, given their characteristics, WSNs consist of numerous resource-constrained nodes. The main limitation points are as follows [5]:

1. Given the low data-transfer rate, the short communication distance, and the harsh environment deployment, the transmission of WSNs is unreliable and has a higher energy costs.
2. Owing to the small size of SNs, each node is supplied with a small battery. WSNs are, however, always deployed in unattended environments or enemy environments; therefore, energy supplementation is impracticable.
3. As SNs use embedded processor and memory, only base computation capacity is available for processing. Therefore, the technology is limited by low computation and storage capacity.

The security of WSNs is related to sensitive data and safety of patients, and it can even escalate to national security. Compared with traditional networks, however, WSNs are vulnerable to various related attacks. Unfortunately, the information transmitted in WSNs is highly important and sensitive, so adversaries $\mathcal{A}$ can destroy WSNs or obtain confidential information from such networks. Therefore, the challenge and priority is to secure the performance of WSNs with small overhead, and this topic has recently been studied by many researchers. Authentication schemes have become the most important concern in the security of WSNs. In the last five years, numerous mutual-authentication and key agreement schemes have been published by researchers around the world and are discussed in the following subsection.

Related Work

The authentication scheme for WSNs has recently been studied by many professors, and several investigations have surveyed the security of WSNs [3, 6–13]. These studies have analyzed the main problems faced by WSN security research and classified authentication schemes into two types: scheme-based asymmetric encryption and scheme-based symmetric encryption. The majority of the schemes aim to achieve improved security performance with small overhead. Nam et al. [14] proposed an anonymous scheme with lightweight computation. The group used elliptic curve cryptography for better security and focused on user anonymity. Watro et al. [15] proposed a security scheme of mutual authentication with RSA cryptosystem and Diffie—Hellman key agreement. Wong et al. [16] proposed another password-based authentication scheme that only uses hash functions. The scheme proposed by Wong et al. is therefore more efficient than Watro et al.'s schemes. However, their scheme is vulnerable to numerous attacks, as proven by M. L. Das et al. [17], who proposed a two-factor scheme with a password and a smart card (SC). Although vulnerable to numerous attacks, the scheme prompted other researchers to improve the two-factor authentication for WSNs. Xue et al. [18] proposed temporal credential authentication for WSNs. This scheme allows the gateway nodes (GW) to issue a temporal credential to users and SNs for mutual authentication. The scheme is efficient because it only uses the hash function and XOR operation. Jiang et al. [19] concluded that Xue et al.'s scheme cannot withstand the privileged insider, weak stolen smart card, identity guessing, and tracking attacks. Then, Jiang et al. proposed a two-factor user authentication scheme with unlinkability for WSNs. Despite presenting an improvement on the weakness of Xue et al.'s approach, Jiang et al.'s scheme is also vulnerable to privileged insider attacks and presents several drawbacks, as proven by A. K. Das [20]. The scheme proposed by A. K. Das used biometrics as the third factor for user authentication and improved the weakness of the scheme by Xue et al. He et al. [21] also found drawbacks in Xue et al.'s scheme. Through their analysis, the team found that Xue et al.'s scheme is vulnerable to offline password guessing, user impersonation, and modification attacks. Thereafter, He et al. proposed a temporal credential authentication with pseudo identity for WSNs. The scheme proposed by Khan and Alghathbar [22] indicated that M. L. Das's scheme cannot withstand bypassing attacks and is vulnerable to privileged insider attacks. Sun et al. [23] concluded that Khan and Alghathbar's scheme is vulnerable to GW impersonation and other related attacks. Sun et al. proposed a scheme to improve the weakness of Khan and Alghathbar's scheme and determined that their scheme had low overhead cost.

Key establishment is the central problem in authentication schemes [24]. Diffie and Hellman proposed the revolutionary introduction of the key establishment protocol [25] and Bellare and Rogaway proposed a model of authentication and key distribution that is widely accepted [26–28]. Choo et al. discovered that all secure key distribution protocols should use partnering definitions based on session identifiers [29] and that session identifiers should also be included within the protocol specification [30]; the secure protocols should construct the session keys using the identities of participants, unique session identifiers and ephemeral-long-term shared secrets [31]; and any entity authentication and key establishment protocol should provide rigorous proof of security based on their meticulous research [32]. They also carefully researched the subtle differences between the well-known models and contributed a better understanding of proof models for key establishment protocols [33]. Based on the careful study, Choo and Hitchcock proposed that the proof models allow different options for the key-sharing requirement in formulation [34]. Numerous researchers have worked on fulfilling this requirement, so listing these works in our paper is unnecessary.

Our Contribution

In this paper, we propose a temporal credential-based mutual authentication with a multiple-password scheme for WSNs. Comparison with other related works shows that our proposed scheme exhibits improved security performance with low overhead. The major contributions are described as follows.

1. We perform user authentication without any GW consumption which presents better efficiency and security performance, as proven by A. K. Das and Amin et al.' s research [20, 35], they bind U with ID_{SC} so that the scheme can reliably withstand D-DOS attacks that are launched by inputting wrong passwords [20] as well as withstand same-login ID attacks [22, 36].

2. We use multiple passwords to authenticate the legality of the user identity. We select all user-inputting passwords, the sequence of passwords, and the number of passwords n as the factors to verify the identity of the user. This innovation not only presents the same security performance as the three-factor authentication based on biometrics but also exhibits a more efficient performance than biometric authentication. This approach overcomes several weaknesses of biometric authentication, which is unsuitable for WSNs. These disadvantages include high noise data rate, false non-match rate, false match rate, intraclass variations, non-universality, spoof attacks [37], high biometric error rate, stolen biometric features attacks [38], and high consumption [20, 39].

3. Through detailed comparison, we found that communication overhead accounts for the majority of the overhead. Most of the related studies, which were concerned only with computation overhead, are not comprehensive. Therefore, more attention should be paid to communication overhead than to computation overhead to evaluate the performance of any scheme in future research.

Notations in This Paper

The notations used in this paper are described as follows.

GW: a gateway node

U: the user

SN: the sensor node

SC: the smart card of U

$\mathcal{A}$: the adversary

ID_U: the identity of U

ID_{GW}: the identity of GW

ID_{SC}: the identity of SC

ID_{SN}: the identity of SN

PW_U: the password of U

n: the number of passwords

k_i, k_{GW}, k_i: the secret number for U,GW,SN respectively

e_i, PK_{GW}, PK_j: the protected information for the secret number of U,GW,SN, respectively

V_i: the verification information of U

DID_{SC}, PID_j: the pseudonym of SC,SN, respectively

T_U, T_{GW}, TS: the current timestamp

RPW_i: the protected information for the multiple password

PTC_i, PTC_j: the protected temporal credential of U, SN, respectively

SK: the session key in the future

σ_U, σ_{GW}: the HMAC output with secret keys k_{UG}, k_{GS}, respectively

(Mac, Ver): a keyed-hashing for message authentication codes

(Enc, Dec): symmetric encryption/decryption functions

$H(\cdot)$: hash function

$||$: bitwise concatenation operation

Review of Nam et al.'s Scheme

In this section, we review Nam et al.'s scheme in detail. The scheme consists of three phases: the registration phase, the login phase, and the authentication and key exchange phase [14]. Nam et al.'s scheme stores an elliptical curve group *G* with generator *P* of prime order *q*; MAC function Σ = (Mac, Ver) [40, 41]; symmetric encryption and decryption functions Δ = (Enc, Dec); and three hash functions, *H*, *J*, and *I* in each entity (we use only *H* to represent the hash function in this paper). After finishing these tasks, GW selects two random numbers, $y \in Z_q^*$ and, $z \in \{0, 1\}^k$, computes $Y = yP$ with $k_{GS} = h(ID_{SN} \| z)$ as the public key and shares a secret key with SN.

Registration phase

A user U registers his identity ID_U and password PW_U through the following steps.

1. A user U registers the identity ID_U and password PW_U and submits ID_U to the GW.
2. GW computes $EID_U = Enc_z(ID_U \| ID_{GW})$ with the key z and sends $\{EID_U, Y, ID_{GW}, G, P, \Sigma, \Delta, H\}$ to U. U stores these messages in the SC.
3. U computes $XEID_U = EID_U \oplus h(ID_U \| PW_U)$ to replace EID_U.

Login, authentication, and key exchange phase

In these phases, U, GW, and SN authenticate each other through the following, and the session key SK is generated. The details of these phases are described as follows:

1. U inserts his SC and inputs the identity ID_U and password PW_U. Then, SC retrieves the current timestamp T_U and gets two random numbers $x \in Z_q^*$, $k_{US} \in \{0, 1\}^k$. SC performs a series of calculations as follows. $K_{UG} = xY$, $X = xP$, $k_{UG} = h(T_U \| X \| Y \| K_{UG})$, $EID_U = XEID_U \oplus h(ID_U \| PW_U)$, $C_U = Enc_{k_{UG}}(ID_U \| EID_U \| k_{US})$ and $\sigma_U = Mac_{k_{UG}}(ID_{GW} \| ID_{SN} \| T_U \| C_U)$. Finally, U sends $(T_U, ID_{SN}, X, C_U, \sigma_U)$ to GW.
2. Upon receiving these message, GW checks the freshness of T_U, if T_U is not fresh, GW discards the session. Otherwise, GW checks whether $Ver_{k_{UG}}(ID_{GW} \| ID_{SN} \| T_U \| C_U, \sigma_U)$ is equal to 1, where $k_{UG} = h(T_U \| X \| Y \| K_{UG})$ and $K_{UG} = yX$. If it is not equal, GW discards the session. Otherwise, GW uses the key k_{UG} to decrypt C_U to get ID_U and EID_U. GW uses the key z to decrypt EID_U to get $ID_U^{'}$. Then, GW checks whether ID_U is equal to $ID_U^{'}$. If they are equal, GW computes $C_{GW} = Enc_{k_{GS}}(k_{US})$ and $\sigma_{GW} = Mac_{k_{GS}}(ID_{GW} \| ID_{SN} \| T_{GW} \| T_U \| C_{GW})$, where T_{GW} is the current TS. Finally, GW sends $(ID_{GW}, T_{GW}, T_U, C_{GW}, \sigma_{GW})$ to SN.
3. Upon receiving these messages, SN first checks the freshness of T_{GW}. If T_{GW} is not fresh, SN aborts the session. Otherwise, SN checks whether $Ver_{k_{GS}}(ID_{GW} \| ID_{SN} \| T_{GW} \| T_U \| C_{GW}, \sigma_{GW})$ is equal to 1. If it is not equal, SN aborts the session. Otherwise, SN decrypts C_{GW} with the key k_{GS} to get k_{US}. Then, SN computes $SK = h(k_{US} \| T_U \| ID_{SN})$ and $\rho_{SN} = h(k_{US} \| T_U \| ID_{SN})$. Finally, SN sends ρ_{SN} to the user U.
4. Upon receiving messages, the user checks whether ρ_{SN} is equal to $h(k_{US} \| ID_{SN} \| T_U)$. If they are not equal, U aborts the session. Otherwise, U computes $SK = h(k_{US} \| T_U \| ID_{SN})$ as the SK.

Password update phase

In this phase, Nam et al. have designed an interactive password update phase as follows:

1. U inserts his SC and inputs ID_U, PW_U, and new password $PW_U^{'}$.
2. SC completes a series of calculations with the random $x \in Z_q^*$ and timestamp T_U as follows. $k_{UG} = xY$, $X = xP$, $k_{UG} = h(T_U \| X \| Y \| K_{UG})$, $EID_U = XEID_U \oplus h(ID_U \| PW_U)$ $C_U = Enc_{k_{UG}}(ID_U \| EID_U)$. Then, SC sends (T_U, C_U, X) to the GW.
3. Upon receiving these messages, GW rejects the request if T_U is not fresh. Otherwise, GW computes $k_{UG} = h(T_U \| X \| Y \| K_{UG})$ and $K_{UG} = yX$. Then, GW uses the key k_{UG} to decrypt C_U to get ID_U and EID_U. GW decrypts EID_U with the key z to get another $ID_U^{'}$. GW checks

whether $\mathrm{ID_U}$ is equal to $\mathrm{ID_U'}$. If they are equal, GW computes $\rho_{\mathrm{GW}} = \mathrm{h(k_{UG} \| X \| ID_U \| ID_{GW})}$ and sends ρ_{GW} to SC.

4. SC checks whether ρ_{GW} is equal to $\mathrm{h(k_{UG} \| X \| ID_U \| ID_{GW})}$. If they are not equal, SC aborts the session. Otherwise, SC computes $\mathrm{XEID_U} = \mathrm{EID_U} \oplus \mathrm{h(ID_U \| PW_U')}$ and finishes the password update phase.

Security Analysis of Nam et al.'s Scheme

In this section, we comprehensively analyze the security performance of Nam et al.'s scheme. During the analysis, several weaknesses of the scheme were identified. Nam et al.'s scheme ensures user anonymity and uses the elliptical curve computational Diffie—Hellman (ECCDH) protocol and authenticated key exchange (AKE) to fulfill the security function. However, further analysis shows that the scheme is vulnerable to the following threats.

D-DOS attacks

In the authentication and key exchange phase or password update phase of Nam et al.'s scheme, SC and GW need to execute numerous complex computations to verify the identity of U. To fulfill this task, SC and GW have to execute the hash function three times, encryption once, decryption twice, and MAC calculation and Ver calculation twice. Following several studies [3, 20, 42], we assume that an adversary $\mathcal{A}$ would start a D-DOS attack that is launched by persistently inputting a wrong $\mathrm{ID_U}$ or wrong $\mathrm{PW_U}$. According to Nam et al. [14] and the reference basis that is analyzed in this paper, each verification needs approximately 9.5 hash calculations, wasting 0.00304 s and costing 0.073 mJ of WSNs. $\mathcal{A}$ would not be suspended until the energy of GW is depleted [42].

Based on the preceding discussion, Nam et al.'s scheme is vulnerable to D-DOS attacks, and adversary $\mathcal{A}$ can easily drain the batteries in the login phase.

Online guessing attacks

In the authentication and key exchange phase, we assume that $\mathcal{A}$ eavesdrops on the communication channel [43]. $\mathcal{A}$ can obtain the secret key k_{US} and compute the SK with an online guessing attack through the following steps:

1. $\mathcal{A}$ obtains T_{U}, $\mathrm{ID_{SN}}$, and ρ_{SN} by intercepting channels U → GW, GW → SN, and SN → U.
2. $\mathcal{A}$ guesses the k_{US} from the directory.
3. $\mathcal{A}$ verifies whether $h(\mathrm{k_{US}} \| \mathrm{ID_{SN}} \| \mathrm{T_U})$ is equal to ρ_{SN}. If both numbers are the same, $\mathcal{A}$ obtains $\mathrm{k_{US}}$. Otherwise, $\mathcal{A}$ repeats steps 2 and 3 until the correct $\mathrm{k_{US}}$ is guessed.
4. After obtaining k_{US}, $\mathcal{A}$ computes $\mathrm{SK = h(k_{US} \| T_U \| ID_{SN})}$ to obtain the SK.

According to the preceding discussion, we conclude that $\mathcal{A}$ can obtain the secret key $\mathrm{k_{US}}$ and compute the SK by online guessing attacks. These findings prove that Nam et al.'s scheme is vulnerable to online guessing attacks.

Lost password threat

Numerous approaches, such as the hit library attack and social engineering [44, 45], can be used to obtain user passwords. The lost password threat is currently popular and is a deadly threat to any one-password-based authentication, including WSNs. If the adversary $\mathcal{A}$ obtains

Table 1. The security comparison with other schemes.

	SSCA	NCA	PIA	MA	UA	ONGA	OFPGA	RA	MITMA	LPT	DDA	MSNA	TFS	IOM	IGA	SKA	PUP	DNAP
D.B.He	yes	no	no	yes	yes	no	yes	yes	yes	no	no	n/a	no	yes	yes	yes	no	no
A.K.Das	yes	no	yes	yes	yes	yes	yes	yes	yes	no	yes	no	yes	no	yes	yes	yes	yes
J.H.Nam	no	yes	yes	no	yes	no	no	no	no	no	no	n/a	no	yes	no	yes	yes	no
K.XUE	no	no	no	yes	no	yes	no	yes	yes	no	no	n/a	no	no	no	yes	yes	no
Q.Jiang	no	no	no	no	yes	no	yes	yes	yes	no	no	n/a	no	yes	no	yes	no	no
M.L.Das	no	no	no	yes	no	no	yes	yes	yes	no	no	n/a	no	no	no	no	no	no
Ours	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes

SSCA: Stolen smart card attack; NCA: Nodes captured attack; PIA: Privileged insider attack; MA: mutual authentication; UA: Anonymity; ONGA: Online guessing attack OFPGA: Off-line password guessing attack; RA: Replay attack; MITMA: Man-in-the-middle attack; LPT: Lost password threat; DDA: D-Dos attack; MSNA: Malicious sensor node attacks; TFS: Three-factor security; IOM: Integrity of message; IGA: identity guessing attack; SKA: session key agreement; SKA: session key agreement; PUP: password updated phase; DNAP: dynamic node addition phase

doi:10.1371/journal.pone.0170657.t001

the commonly used passwords of U by other methods, we can see that the authentication scheme encounters a considerable threat.

Replay attacks

In the authentication and key exchange phase, we assume that an adversary $\mathcal{A}$ intercepts the message ρ_{SN}. Then, $\mathcal{A}$ sends ρ_{SN} to U. As U does not check the freshness of T, U cannot realize that $\mathcal{A}$ has already obtained the ρ_{SN}, therefore proving that Nam et al.'s scheme is vulnerable to replay attacks.

Impersonation attacks

In the authentication and key exchange phase, SN authentication verifies whether the identity of GW is invalid. Furthermore, U does not authenticate the validity of SN. $\mathcal{A}$ can start the impersonation attack by forging GW and SN as in the following steps:

1. $\mathcal{A}$ intercepts ID_{GW}, T_{GW}, T_U, C_{GW}, and σ_{GW} from the communication channel GW → SN.
2. $\mathcal{A}$ sends ID_{GW}, T_{GW}, T_U, C_{GW}, and σ_{GW} to SN.
3. $\mathcal{A}$ passes the MAC, and $\mathcal{A}$ is believed to be the real GW.

According to the preceding discussion, as U does not check the freshness of T, we can safely conclude that Nam et al.'s scheme is vulnerable to impersonation attacks. The detailed security analysis is described in Table 1.

Our Proposed Scheme

In this section, we propose a temporal credential-based mutual authentication with multiple-password scheme for WSNs. The temporary SK has many advantages relative to using long-term keys according to Choo's research [46]. Our scheme not only inherits the excellent properties of Nam et al.'s scheme but also improves upon the weaknesses of their scheme. As our scheme uses multiple passwords to replace Tate-pairing computation and the fuzzy extractor function, our scheme can achieve the same security performance with smaller overhead [47].

Unlike Nam et al.'s scheme, our proposed scheme consists of five phases: registration phase, login phase, authentication and key exchange phase, password update phase, and dynamic-node addition phase. These phases are described in detail as follows.

Registration phase

In this phase, we register a legal user, U, and sensor nodes, SN. This concept has already been presented in other studies [18, 21]. The registration phase is executed in a rigorously secure environment prior to the deployment of WSNs. Before registration, GW assigns the unique identities, namely, ID_{SN}, ID_{SC}, and ID_{GW}, to SNs, SC, and the GW respectively. Then, GW randomly generates a secret number, k_{GW}. Finally, the hash function-H(·); message authentication check scheme MAC(·); and Ver(·) are stored in SC, GW, and SN. The registration phase is described in detail as follows.

Registration phase for legal user. In this phase, we register the legal user U through the following steps.

1. U inserts his SC and inputs his multiple-password PW_1, $PW_2 \cdots PW_n$. U generates a random secret number K_i and gets the unique identifier ID_{SC}. U computes $RPW_i = H(ID_{SC} \| PW_1 \| PW_2 \| \cdots \| PW_n \| n \| k_i)$ and retrieves the timestamp TS_1. Finally, U sends (RPW_i, TS_1, ID_{SC}) to GW.

2. Upon receiving the message, GW checks the freshness of TS_1. If TS_1 is not fresh, GW rejects the request. Otherwise, GW gets the unique identifier ID_{GW}. Then, GW computes $TC_i = H(k_{GW} \| ID_{GW} \| ID_{SC})$, $PTC_i = TC_i \oplus RPW_i$, and $PK_{GW} = PTC_i \oplus k_{GW}$. Then, GW retrieves the current timestamp TS_2. Finally, GW stores the tuple (ID_{GW}, ID_{SC}, PK_{GW}) in the verification table and sends (PTC_i, TS_2, ID_{GW}) to U.

3. Upon receiving the message, U checks the freshness of TS_2. If TS_2 is not fresh, U rejects the request. Otherwise, U computes $e_i = k_i \oplus H(n \| PW_1 \| PW_2 \| \cdots \| PW_n)$, $V_i = H(e_i \| RPW_i \| ID_{SC} \| k_i \| n)$. Finally, U stores ($e_i$, V_i, PTC_i, ID_{SC}, ID_{GW}) in the SC.

In this phase, adversary $\mathcal{A}$ cannot restore the sensitive number because of the property of the hash function [48–50] and the confidentiality property of the XOR operation [51–53], as well as the information stored in GW and SC. The random secret numbers k_i and k_{GW} are not stored in GW. This phase is shown in Fig 1.

Registration for sensor node. In our scheme, each legal SN is required to register in GW so that we can verify the legal SN and add the new SN to WSNs in the future. Before SN registration, the legality of U should be verified. The steps are as follows.

1. SN generates a random secret number k_j and gets the unique identifier ID_{SN}. Then, SN computes $PID_j = H(ID_{SN} \| k_j)$, $PK_j = PID_j \oplus k_j$ and replaces ID_{SN} with PID_j. Finally, SN retrieves timestamp TS_3 and sends (PID_j, TS_3) to GW.

2. Upon receiving the message, GW checks the freshness of TS_3. If TS_3 is not fresh, GW rejects the request. Otherwise, GW computes $TC_j = H(k_{GW} \| PID_j)$, $PTC_j = TC_j \oplus PID_j$. Then, GW retrieves the timestamp TS_4 and stores PID_j. Finally, GW sends (TS_4, PTC_j) to SN.

3. Upon receiving the message, SN checks the freshness of TS_4. If TS_4 is not fresh, GW rejects the request. Otherwise, SN stores (PK_j, PTC_j).

In this phase, different SNs possess different PID_j and PK_j, and the random secret number K_j is not stored in SN. Therefore, our scheme can withstand node capture attacks, as analyzed in the security analysis section. This phase is shown in Fig 2. After finishing the entire registration scheme, GW deletes k_{GW}, SC deletes K_i, and SN deletes K_j before the deployment of WSNs.

Login phase

The login phase procedure is described in detail as follows. If U attempts to login to WSNs and obtains data from SN, the following steps are executed. This phase is shown in Fig 3.

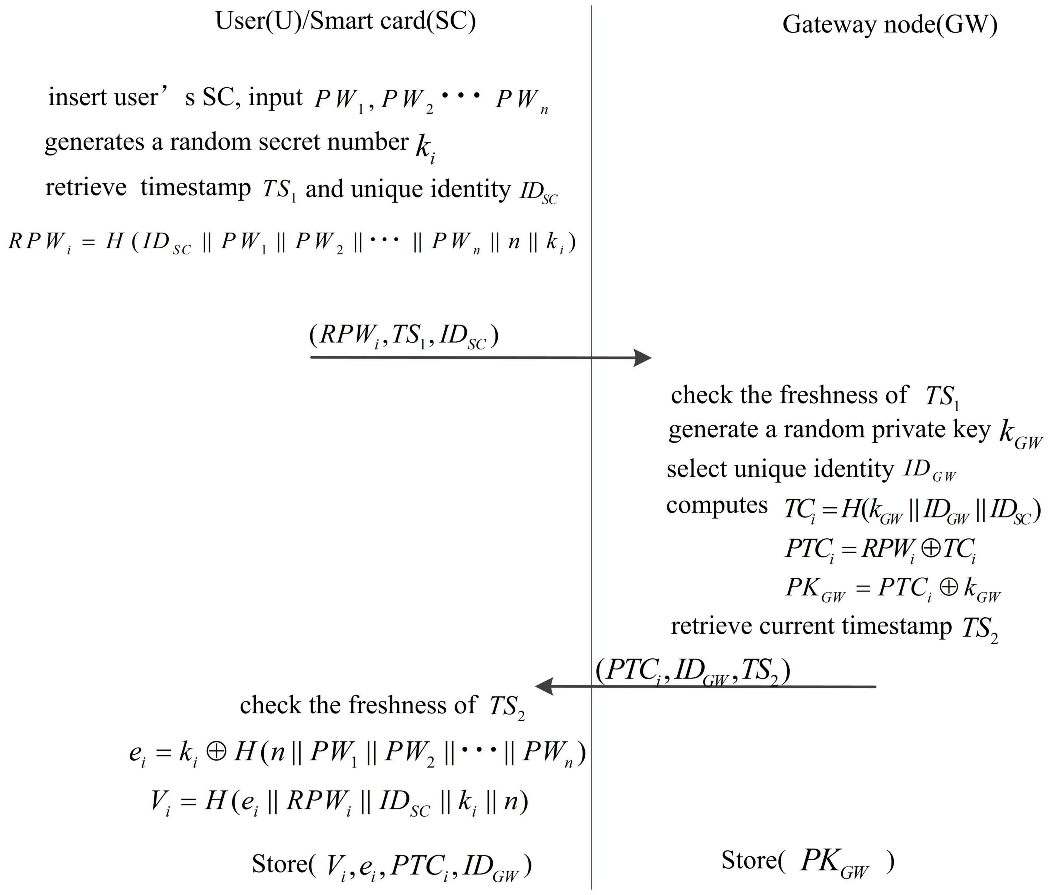

User(U)/Smart card(SC)
Gateway node(GW)
insert user' s SC, input $PW_1, PW_2 \cdots PW_n$
generates a random secret number k_i
retrieve timestamp TS_1 and unique identity ID_{SC}
$RPW_i = H(ID_{SC} \| PW_1 \| PW_2 \| \cdots \| PW_n \| n \| k_i)$
(RPW_i, TS_1, ID_{SC})
check the freshness of TS_1
generate a random private key k_{GW}
select unique identity ID_{GW}
computes $TC_i = H(k_{GW} \| ID_{GW} \| ID_{SC})$
$PTC_i = RPW_i \oplus TC_i$
$PK_{GW} = PTC_i \oplus k_{GW}$
retrieve current timestamp TS_2
(PTC_i, ID_{GW}, TS_2)
check the freshness of TS_2
$e_i = k_i \oplus H(n \| PW_1 \| PW_2 \| \cdots \| PW_n)$
$V_i = H(e_i \| RPW_i \| ID_{SC} \| k_i \| n)$
Store(V_i, e_i, PTC_i, ID_{GW})
Store(PK_{GW})

Fig 1. The registration phase for user of our scheme.

doi:10.1371/journal.pone.0170657.g001

1. U inserts his SC and inputs the registered multiple-password PW_1, $PW_2 \cdots PW_n$.
2. SC gets the unique identifier ID_{SC} and computes $k_i = e_i \oplus H(n \| PW_1 \| PW_2 \| \cdots \| PW_n)$, $RPW_i = H(ID_{SC} \| PW_1 \| PW_2 \| \cdots \| PW_n \| n \| k_i)$.
3. SC checks whether $H(e_i \| RPW_i \| k_i \| n \| ID_{SC})$ is equal to V_i. If it is not equal, SC rejects the request. Otherwise, SC retrieves timestamp TS_1 and computes $TC_i = PTC_i \oplus RPW_i$, $PKS_i = k_i \oplus H(TC_i \| TS_1)$, $C_i = MAC_{k_i}(TC_i \| TS_1 \| RPW_i)$, $DID_{SC} = ID_{SC} \oplus H(TS_1 \| ID_{GW})$.
4. Finally, U sends (PTC_i, C_j, PKS_i, TS_1, DID_{SC}) to GW.

Authentication and key exchange phase

In this phase, we describe the authentication mechanism through U, GW, and SC. The mechanism achieves mutual authentication and generates the SK, for future use. The details are presented as follows.

1. Upon receiving the message, GW checks the freshness of TS_1. If it is not fresh, GW aborts the session. Otherwise, GW retrieves the unique identity ID_{GW} and computes $ID_{SC} = DID_{SC} \oplus H(TS_1 \| ID_{GW})$, GW obtains the PK_{GW} corresponding to ID_{SC} in the verification table. Then, GW computes $k_{GW} = PK_{GW} \oplus PTC_i$, $TC_i = H(k_{GW} \| ID_{GW})$, $RPW_i = PTC_i \oplus TC_i$, and $k_i = PKS_i \oplus H(TC_i \| TS_1)$. GW checks whether $Ver_{ki}(TC_i \| TS_1 \| RPW_i, C_i)$ is equal to

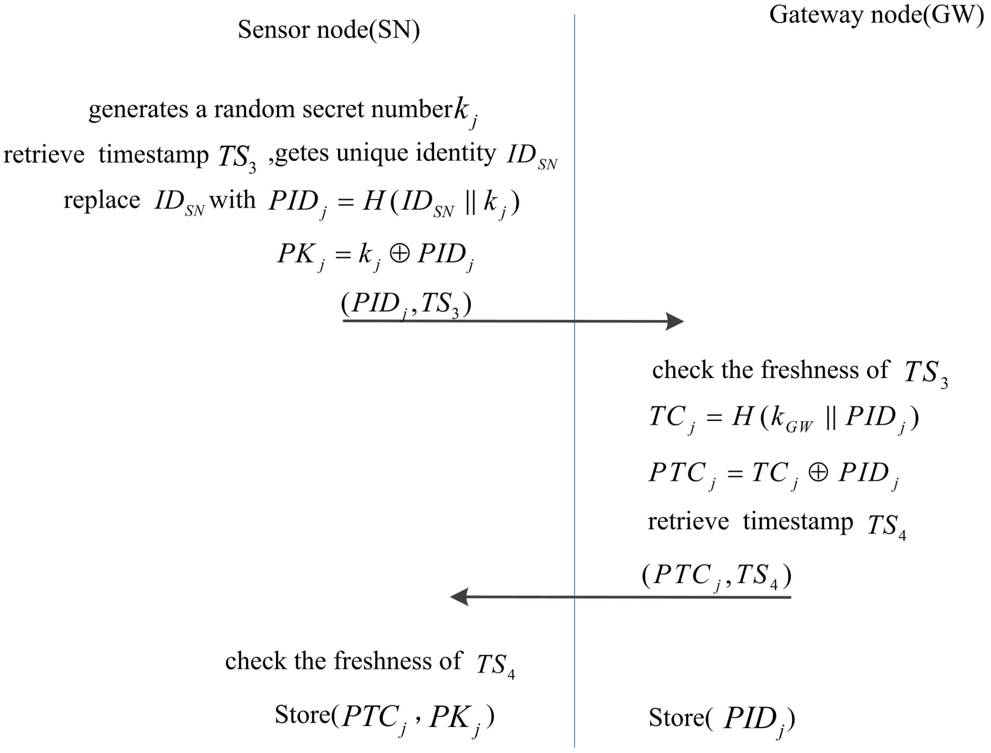

Sensor node(SN)
Gateway node(GW)
generates a random secret number k_j
retrieve timestamp TS_3 ,getes unique identity ID_{SN}
replace ID_{SN} with $PID_j = H(ID_{SN} \| k_j)$
$PK_j = k_j \oplus PID_j$
(PID_j, TS_3)
check the freshness of TS_3
$TC_j = H(k_{GW} \| PID_j)$
$PTC_j = TC_j \oplus PID_j$
retrieve timestamp TS_4
(PTC_j, TS_4)
check the freshness of TS_4
Store(PTC_j , PK_j)
Store(PID_j)

Fig 2. The registration phase of sensor node of our scheme.

doi:10.1371/journal.pone.0170657.g002

1. If it is not equal, GW aborts the session. Otherwise, GW retrieves timestamp TS_2 and computes $TC_j = H(k_{GW} \|PID_j)$, $PKS_{GW} = k_i \oplus H(TC_j \| TS_2)$,$C_{GW} = MAC_{TC_j}(k_i \| TS_2 \| PID_j)$. Finally, GW sends ($PID_j$, C_{GW}, PKS_{GW}, TS_2) to SN.

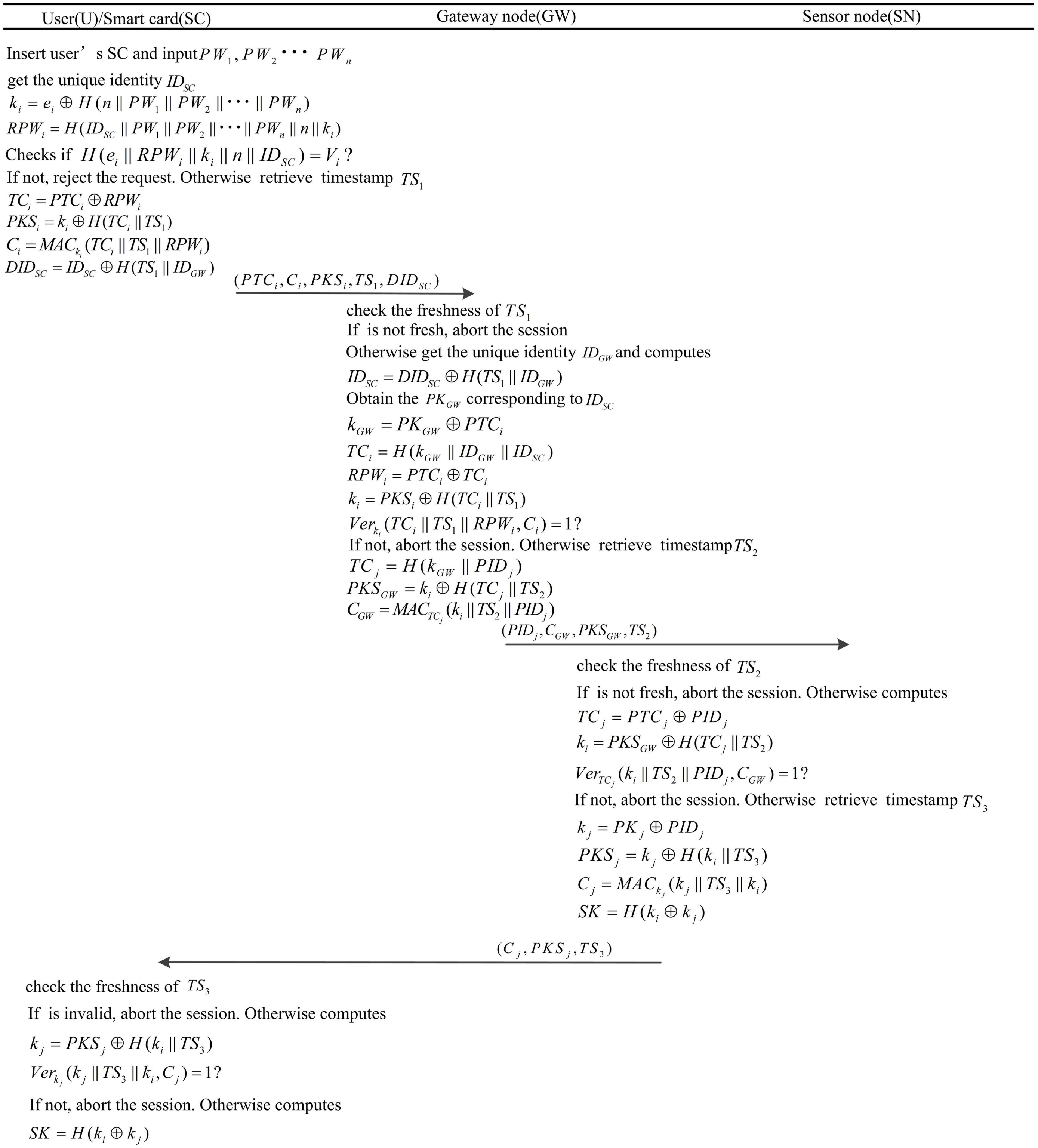
2. Upon receiving the message, SN checks the freshness of TS_2. If it is not fresh, SN aborts the session. Otherwise, SN computes $TC_j = PTC_j \oplus PID_j$, $k_i = PKS_{GW} \oplus H(TC_j \|TS_2)$. Then, SN checks whether $Ver_{TC_j}(k_i \| TS_2 \| PID_j, C_{GW})$ is equal to 1. If it is not equal, SN aborts the session. Otherwise, SN retrieves timestamp TS_3 and computes $k_i = PK_i \oplus PID_j$, $PKS_j = k_j \oplus H(k_i \|TS_3)$,$C_j = MAC_{k_j}(k_j \| TS_3 \| k_i)$ and $SK = H(k_i \oplus k_j)$ as the SK. Finally, SN sends (C_j, PKS_j, TS_3) to U.

3. Upon receiving the message, U checks the freshness of TS_3. If it is not fresh, U aborts the session. Otherwise, the SC of U computes $k_j = PKS_j \oplus H(k_i \|TS_3)$. Then SC checks whether $Ver_{k_j}(k_j \| TS_3 \| k_i, C_j)$ is equal to 1? If it is not equal, SC aborts the session. Otherwise, SC computes $SK = H(k_i \oplus k_j)$ as the SK for the future.

In this phase, our proposed scheme not only achieves mutual authentication and key establishment but also checks the integrity of the message. Each message authentication check function in U, SN, and GW uses different secret encryption keys for secure communication [3]. The detailed security performance of our scheme is discussed in the security analysis section, and the authentication and key exchange phase is shown in Fig 3.

Password updated phase

For security reasons, U needs to change his/her password periodically. In this phase, we propose the password-updating phase to change the password of U and U can change the



User(U)/Smart card(SC) — Gateway node(GW) — Sensor node(SN)

Insert user’ s SC and input $PW_1, PW_2 \cdots PW_n$

get the unique identity ID_{SC}

$k_i = e_i \oplus H(n \| PW_1 \| PW_2 \| \cdots \| PW_n)$

$RPW_i = H(ID_{SC} \| PW_1 \| PW_2 \| \cdots \| PW_n \| n \| k_i)$

Checks if $H(e_i \| RPW_i \| k_i \| n \| ID_{SC}) = V_i$?

If not, reject the request. Otherwise retrieve timestamp TS_1

$TC_i = PTC_i \oplus RPW_i$

$PKS_i = k_i \oplus H(TC_i \| TS_1)$

$C_i = MAC_{k_i}(TC_i \| TS_1 \| RPW_i)$

$DID_{SC} = ID_{SC} \oplus H(TS_1 \| ID_{GW})$

$\xrightarrow{(PTC_i, C_i, PKS_i, TS_1, DID_{SC})}$

check the freshness of TS_1

If is not fresh, abort the session

Otherwise get the unique identity ID_{GW} and computes

$ID_{SC} = DID_{SC} \oplus H(TS_1 \| ID_{GW})$

Obtain the PK_{GW} corresponding to ID_{SC}

$k_{GW} = PK_{GW} \oplus PTC_i$

$TC_i = H(k_{GW} \| ID_{GW} \| ID_{SC})$

$RPW_i = PTC_i \oplus TC_i$

$k_i = PKS_i \oplus H(TC_i \| TS_1)$

$Ver_{k_i}(TC_i \| TS_1 \| RPW_i, C_i) = 1$?

If not, abort the session. Otherwise retrieve timestamp TS_2

$TC_j = H(k_{GW} \| PID_j)$

$PKS_{GW} = k_i \oplus H(TC_j \| TS_2)$

$C_{GW} = MAC_{TC_j}(k_i \| TS_2 \| PID_j)$

$\xrightarrow{(PID_j, C_{GW}, PKS_{GW}, TS_2)}$

check the freshness of TS_2

If is not fresh, abort the session. Otherwise computes

$TC_j = PTC_j \oplus PID_j$

$k_i = PKS_{GW} \oplus H(TC_j \| TS_2)$

$Ver_{TC_j}(k_i \| TS_2 \| PID_j, C_{GW}) = 1$?

If not, abort the session. Otherwise retrieve timestamp TS_3

$k_j = PK_j \oplus PID_j$

$PKS_j = k_j \oplus H(k_i \| TS_3)$

$C_j = MAC_{k_j}(k_j \| TS_3 \| k_i)$

$SK = H(k_i \oplus k_j)$

$\xleftarrow{(C_j, PKS_j, TS_3)}$

check the freshness of TS_3

If is invalid, abort the session. Otherwise computes

$k_j = PKS_j \oplus H(k_i \| TS_3)$

$Ver_{k_j}(k_j \| TS_3 \| k_i, C_j) = 1$?

If not, abort the session. Otherwise computes

$SK = H(k_i \oplus k_j)$

Fig 3. The login, authentication and key exchange phase of our scheme.

doi:10.1371/journal.pone.0170657.g003

sequence of passwords and the number of passwords as the new identity characteristics with minimal consumption. The details of this phase are described as follows.

1. U inserts his SC and inputs the older multiple-password $PW_1, PW_2 \cdots PW_n$.
2. SC gets the unique identifier ID_{SC} and computes $k_i = e_i \oplus H(n \| PW_1 \| PW_2 \| \cdots \| PW_n)$, $RPW_i = H(ID_{SC} \| PW_1 \| PW_2 \| \cdots \| PW_n \| n \| k_i)$.
3. SC checks whether $H(e_i \| RPW_i \| k_i \| n \| ID_{SC})$ is equal to V_i. If it is not equal, SC rejects the request. Otherwise, SC computes $TC_i = PTC_i \oplus RPW_i$. Then, U inputs his new multiple-password $PW_1^{new}, PW_2^{new} \cdots PW_m^{new}$.
4. After inputting the new multiple-password, SC computes $RPW_i^{new} = H(ID_{SC} \| PW_1^{new} \| PW_2^{new} \| \cdots \| PW_m^{new} \| m \| k_i)$, $PTC_i^{new} = TC_i \oplus RPW_i^{new}$, $e_i^{new} = k_i \oplus H(m \| PW_1^{new} \| PW_2^{new} \| \cdots \| PW_m^{new})$, $V_i^{new} = H(e_i^{new} \| RPW_i^{new} \| ID_{SC} \| k_i \| m)$. U sends PTC_i, PTC_i^{new}, and current TS to GW. Finally, SC replaces (e_i, V_i, PTC_i) with$(e_i^{new}, V_i^{new}, PTC_i^{new})$.
5. Upon receiving PTC_i^{new}, GW checks the freshness of TS. If it is not fresh, GW rejects the request. Otherwise, GW computes $k_{GW} = PK_{GW} \oplus PTC_i$, $PK_{GW}^{new} = PTC_i^{new} \oplus k_{GW}$. Then, GW replaces PK_{GW} with PK_{GW}^{new}.

Dynamic node addition phase

New node deployment is inevitable in WSNs because nodes may be lost, exhausted, or destroyed [54]. In this phase, our proposed scheme allows U to add new SN to WSNs after deployment. Our scheme strictly requires that the dynamic node addition phase must be executed by the legal user. Thus, our scheme must initially verify the legality of U. We assume that a new sensor node SN^{new} is going to join the WSNs, and the following steps must be executed.

1. U inserts his SC and inputs the registered multiple-password $PW_1, PW_2 \cdots PW_n$.
2. SC gets the unique identifier ID_{SC} and computes $k_i = e_i \oplus H(n \| PW_1 \| PW_2 \| \cdots \| PW_n)$ and $RPW_i = H(ID_{SC} \| PW_1 \| PW_2 \| \cdots \| PW_n \| n \| k_i)$.
3. SC checks whether $H(e_i \| RPW_i \| k_i \| n \| ID_{SC})$ is equal to V_i. If it is not equal, SC rejects the request. Otherwise, SC sends PTC_i and the current TS to GW.
4. GW checks the freshness of TS. If it is not fresh, GW rejects the request. Otherwise, GW computes $k_{GW} = PK_{GW} \oplus PTC_i$ and assigns the new unique identifier ID_{SC}^{new} to SN^{new} via a secure channel.
5. Finally, SN^{new} executes the registration phase for the sensor node.

Note that in this phase, the dynamic addition phase must be executed by a legal U that is authenticated by SC. This mechanism is able to withstand malicious sensor node attacks.

Security Analysis

In this section, we analyze the security performance of our proposed scheme by both formal and informal analyses. We assume that $\mathcal{A}$ threatens the security of WSNs. Based on the existing defined models of adversary capabilities that are widely accepted [26, 27, 55, 56], and we conclude that $\mathcal{A}$ possesses the following hacking capabilities: (1) intercept the transmitted message via the channel [3, 6]; (2) use power analysis attacks to obtain the information stored in SC [57, 58] and use sensor node capture attack to obtain the information stored in SN [59–61]; (3)use dictionary attacks to guess numbers [43]; (4) posses the right to access the gateway station because he/she is a privileged user [40]; and (5) obtain the used passwords of U through other methods. We assume that sensitive information (PW_1, $PW_2 \cdots PW_n$, n, k_i, k_j, k_{GW}, TC_j, TC_i, SK) is attractive to $\mathcal{A}$. Our goal is to prevent the sensitive information from being extracted by $\mathcal{A}$. Thus we carefully analyzed the security performance of our proposed scheme using BAN-logic [62], which is popularly used to ensure the security of communication and session key agreement. The details of our analysis are described as follows.

Formal analysis based on BAN-logic

In this section, we use BAN-logic to analyze the security of our proposed scheme. The notations of BAN-logic are defined as follows, where P denotes the principal as well as, X and Y denote the statements.

$P \mid\equiv X$: P believes X

$P \lhd X$: P sees X

$P \mid \sim X$: P once said X

$P \Rightarrow X$: P has jurisdiction over X

$\#(X)$: X is fresh

(X, Y): The formulae X or Y is one part of the formulae (X, Y)

$< X >_Y$: X combined with Y

$\{X\}_K$: X is encrypted under the key K

$(X)_K$: X is hashed with the key K

$P \overset{K}{\leftrightarrow} Q$: P and Q communicate via shared key K

SK: The session key between U and SN

$P \overset{X}{\Leftrightarrow} Q$: The formulae X is known only to P and Q

Some main logical postulates of the BAN-logic are as follows:

The Message-meaning rule: $\frac{P \mid\equiv P \overset{K}{\leftrightarrow} Q, P \lhd \{X\}_K}{P \mid\equiv Q \mid\sim X}$, $\frac{P \mid\equiv P \overset{X}{\Leftrightarrow} Q,\ P \lhd <X>_Y}{P \mid\equiv Q \mid\sim X}$

The nonce-verification rule: $\frac{P \mid\equiv \#(X), P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}$

The jurisdiction rule: $\frac{P \mid\equiv Q \Rightarrow X, P \mid\equiv Q \mid\equiv X,}{P \mid\equiv X}$

The belief rule: $\frac{P \mid\equiv X,\ P \mid\equiv Y,}{P \mid\equiv (X,Y)}$ $\frac{P \mid\equiv (X,Y)}{P \mid\equiv X}$, $\frac{P \mid\equiv Q \mid\equiv (X,Y)}{P \mid\equiv Q \mid\equiv X}$

The freshness rule: $\frac{P \mid\equiv \#(X)}{P \mid\equiv \#(X,Y)}$

The session key rule: $\frac{P \mid\equiv \#(X), P \mid\equiv Q \mid\equiv X}{P \mid\equiv P \overset{K}{\leftrightarrow} Q}$

In order to prove the security of proposed scheme, the follow goals of BAN-logic must be satisfied.

Goal 1. $U \mid\equiv (U \overset{SK}{\leftrightarrow} SN)$

Goal 2. $U \mid\equiv SN \mid\equiv (U \overset{SK}{\leftrightarrow} SN)$

Goal 3. $\mathrm{SN}| \equiv (\mathrm{U} \overset{\mathrm{SK}}{\leftrightarrow} \mathrm{SN})$

Goal 4. $\mathrm{SN}| \equiv \mathrm{U}| \equiv (\mathrm{U} \overset{\mathrm{SK}}{\leftrightarrow} \mathrm{SN})$

Goal 5. $\mathrm{U}| \equiv (\mathrm{U} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{SN})$

Goal 6. $\mathrm{U} | \equiv \mathrm{SN}| \equiv (\mathrm{U} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{SN})$

Goal 7. $\mathrm{SN}| \equiv (\mathrm{U} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{SN})$

Goal 8. $\mathrm{SN} | \equiv \mathrm{U}| \equiv (\mathrm{U} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{SN})$

First, the initial status of our scheme is made according to the following assumptions:

A_1: $\mathrm{U} | \equiv \#(\mathrm{TS_1})$

A_2: $\mathrm{U} | \equiv \#(\mathrm{TS_3})$

A_3: $\mathrm{U} | \equiv \mathrm{U} \xleftrightarrow{\mathrm{TC_i}} \mathrm{GW}$

A_4: $\mathrm{U} | \equiv \mathrm{U} \xleftrightarrow{\mathrm{RPW_i}} \mathrm{GW}$

A_5: $\mathrm{U} | \equiv \mathrm{SN} \Rightarrow \mathrm{U} \overset{\mathrm{SK}}{\leftrightarrow} \mathrm{SN}$

A_6: $\mathrm{U} | \equiv \mathrm{SN} \Rightarrow \mathrm{U} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{SN}$

A_7: $\mathrm{U} | \equiv \mathrm{SN} \Rightarrow \mathrm{U} \overset{\mathrm{k_j}}{\leftrightarrow} \mathrm{SN}$

A_8: $\mathrm{GW} | \equiv \#(\mathrm{TS_1})$

A_9: $\mathrm{GW} | \equiv \#(\mathrm{TS_2})$

A_{10}: $\mathrm{GW} | \equiv \mathrm{U} \xleftrightarrow{\mathrm{TC_i}} \mathrm{GW}$

A_{11}: $\mathrm{GW} | \equiv \mathrm{U} \xleftrightarrow{\mathrm{RPW_i}} \mathrm{GW}$

A_{12}: $\mathrm{GW} | \equiv \mathrm{SN} \xleftrightarrow{\mathrm{TC_j}} \mathrm{GW}$

A_{13}: $\mathrm{GW} | \equiv \mathrm{SN} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{GW}$

A_{14}: $\mathrm{SN} | \equiv \#(\mathrm{TS_2})$

A_{15}: $\mathrm{SN} | \equiv \#(\mathrm{TS_3})$

A_{16}: $\mathrm{SN} | \equiv \mathrm{SN} \xleftrightarrow{\mathrm{TC_j}} \mathrm{GW}$

A_{17}: $\mathrm{SN} | \equiv \mathrm{SN} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{GW}$

A_{18}: $\mathrm{SN} | \equiv \mathrm{GW} \Rightarrow \mathrm{U} \overset{\mathrm{SK}}{\leftrightarrow} \mathrm{SN}$

A_{19}: $\mathrm{SN} | \equiv \mathrm{GW} \Rightarrow \mathrm{U} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{SN}$

A_{20}: $\mathrm{SN} | \equiv \mathrm{U} \Rightarrow \mathrm{U} \overset{\mathrm{SK}}{\leftrightarrow} \mathrm{SN}$

A_{21}: $\mathrm{SN} | \equiv \mathrm{U} \Rightarrow \mathrm{U} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{SN}$

Second, our scheme is transformed to the idealized form.

M_1: $\mathrm{U} \rightarrow \mathrm{GW} : (\mathrm{TS_1}, \mathrm{U} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{SN})_{\mathrm{TC_i}}$

M_2: $\mathrm{GW} \rightarrow \mathrm{SN} : (\mathrm{TS_2}, \mathrm{U} | \equiv \mathrm{U} \overset{\mathrm{k_i}}{\leftrightarrow} \mathrm{SN})_{\mathrm{TC_j}}$

M_3: $SN \rightarrow U : (TS_3, U \overset{k_i}{\leftrightarrow} SN, U \overset{k_j}{\leftrightarrow} SN)_{k_i}$

Third, the idealized form of our scheme is analyzed based on BAN-logic and the assumptions. The main steps are described as follows:

By M_1 and the seeing rule, we get:

S_1: $GW \triangleleft (TS_1, U \overset{k_i}{\leftrightarrow} SN)_{TC_i}$

By A_{10}, S_1 and the message-meaning rule, we get:

S_2: $GW \mid\equiv U \mid\sim (TS_1, U \overset{k_i}{\leftrightarrow} SN)$

By A_8, S_2, freshness rule and nonce-verification, we get:

S_3: $GW \mid\equiv U \mid\equiv U \overset{k_i}{\leftrightarrow} SN$

By M_2 and the seeing rule, we get:

S_4: $SN \triangleleft (TS_2, U \mid\equiv U \overset{k_i}{\leftrightarrow} SN)_{TC_j}$

By A_{16}, S_4 and the message-meaning rule, we get:

S_5: $SN \mid\equiv GW \mid\sim (TS_1, U \mid\equiv U \overset{k_i}{\leftrightarrow} SN)$

By A_{14}, S_5, freshness rule and nonce-verification, we get:

S_6: $SN \mid\equiv GW \mid\equiv (U \mid\equiv U \overset{k_i}{\leftrightarrow} SN)$

By A_{19}, S_6 and the jurisdiction rule, we get:

S_7: $SN \mid\equiv U \mid\equiv U \overset{k_i}{\leftrightarrow} SN$ (Goal 8)

By A_{21}, S_7 and the jurisdiction rule, we get:

S_8: $SN \mid\equiv U \overset{k_i}{\leftrightarrow} SN$ (Goal 7)

By S_7 and session key rule which k_i is the necessary parameters of SK, we get:

S_9: $SN \mid\equiv U \mid\equiv (U \overset{SK}{\leftrightarrow} SN)$ (Goal 4)

By A_{20}, S_9 and the jurisdiction rule, we get:

S_{10}: $SN \mid\equiv U \overset{SK}{\leftrightarrow} SN$ (Goal 3)

By M_3 and the seeing rule, we get:

S_{11}: $U \triangleleft (TS_3, U \overset{k_i}{\leftrightarrow} SN, U \overset{k_j}{\leftrightarrow} SN)_{k_i}$

By A_{10}, S_{11} and the message-meaning rule, we get:

S_{12}: $U \mid\equiv SN \mid\sim (TS_3, U \overset{k_i}{\leftrightarrow} SN, U \overset{k_j}{\leftrightarrow} SN)$

By A_{15}, S_{12} freshness rule and nonce-verification, we get:

S_{13}: $U \mid\equiv SN \mid\equiv (U \overset{k_i}{\leftrightarrow} SN, U \overset{k_j}{\leftrightarrow} SN)$

By S_{13} and the belief rule, we get:

S_{14}: $U \mid\equiv SN \mid\equiv U \overset{k_i}{\leftrightarrow} SN$ (Goal 6)

S_{15}: $U \mid\equiv SN \mid\equiv U \overset{k_j}{\leftrightarrow} SN$

By A_6, S_{14} and the jurisdiction rule, we get:

S_{16}: $U \mid\equiv U \overset{k_i}{\leftrightarrow} SN$ (Goal 5)

By S_{15} and the session key rule which k_j is the necessary parameters of SK, we get:

S_{17}: $U \mid\equiv SN \mid\equiv (U \overset{SK}{\leftrightarrow} SN)$ (Goal 2)

By A_5, S_{17} and the jurisdiction rule, we get:

S_{18}: $U \mid\equiv U \overset{SK}{\leftrightarrow} SN$ (Goal 1)

From the above discussion, our scheme satisfies (Goal 1), (Goal 2), (Goal 3), (Goal 4), (Goal 5), (Goal 6), (Goal 7) and (Goal 8). Therefore, U, GW and SN perform the mutual authentication and session key exchange securely.

Informal analysis

In this section, we prove our scheme could withstand other attacks. The detailed analysis is described as follows.

Stolen smart card attacks. We know that $\mathcal{A}$ could use a power analysis attack to extract the information stored in the SC. We assume that $\mathcal{A}$ obtains information (e_i, V_i, PTC_i, ID_{SC}). These messages are operated after a one-way hash function. The multiple passwords and the secret number K_i from the SC are impossible to obtain. Because $\mathcal{A}$ meets the property of the one-way hash function [48–50], our scheme can withstand the stolen SC attacks.

Nodes captured attacks. After WSNs are deployed in the target field, $\mathcal{A}$ can easily capture a legitimate sensor node [59–61]. Although there are some important studies that focus on the key revocation protocols [63, 64], we believe the confidentiality of stored key/data is as important as key revocation. We assume that $\mathcal{A}$ could obtain (PTC_j, PK_j) from SN. Owing to the properties of the one-way hash function and XOR operation [51–53], the secret number k_j or TC_j are impossible to obtain from SN. Given that ID_{SN} is replaced with PID_j in the registration phase, $\mathcal{A}$ cannot extract ID_{SN}. The secret number, k_j, is impossible to guess because of the two unknown numbers. To obtain TC_j, $\mathcal{A}$ can compute $TC_j = PTC_j \oplus PID_j$. However, PID_j is not stored in SN. Therefore, $\mathcal{A}$ cannot obtain TC_j. According to the preceding discussion, we can conclude that our proposed scheme can withstand the nodes captured attack.

Privileged insider attacks. We assume that the adversary $\mathcal{A}$ is a privileged insider of WSNs. Therefore, $\mathcal{A}$ can access GW to obtain others' sensitive information. In our scheme, GW does not store the passwords of U and other sensitive information. Therefore, $\mathcal{A}$ cannot extract the passwords of U. We assume that $\mathcal{A}$ can obtain (PK_{GW}, PID_j) from GW. Given the properties of the one-way hash function and XOR operation, deriving k_{GW} and TC_i is an almost impossible task for $\mathcal{A}$. We assume that $\mathcal{A}$ intends to compute $k_{GW} = PTC_i \oplus PK_{GW}$. However, since PTC_i is stored in the SC of U, $\mathcal{A}$ cannot obtain k_{GW}. The preceding discussion shows that our proposed scheme can withstand privileged insider attacks.

Impersonation attacks/ mutual authentication. The adversary $\mathcal{A}$ can impersonate the GW to send/receive the message or install any program to take over the entire network [65]. In our scheme, each receiver must authenticate the identity of the sender by MAC and Ver functions with the sender's own secret key. GW verifies the identity of U by computing $Ver_{k_i}(TC_i \parallel TS_1 \parallel RPW_i, C_i) = 1?$ with K_i. SN verifies the identity of GW by $Ver_{TC_j}(k_i \parallel TS_2 \parallel PID_j, C_{GW}) = 1?$ with TC_j. U verifies the identity of SN by $Ver_{k_j}(k_j \parallel TS_3 \parallel k_i, C_j) = 1?$ with K_j. $\mathcal{A}$ cannot impersonate any legitimate entity without knowing the secret numbers, such as K_i, TC_j, and k_j. Accordingly our proposed scheme can withstand an impersonation attack and achieve mutual authentication.

User anonymity. According to Choo et al.' research [66], there is a mechanical approach to derive identity-based schemes from existing Diffie-Hellman-based schemes. After a careful study of this work, our scheme is designed to withstand this method for protecting user's anonymity. In the login phase, our proposed scheme uses ID_{SC} as the only identity of U. However, a serious problem with user privacy exists. User anonymity is necessary to resist tracing attacks. Our scheme hides ID_{SC} in $RPW_i = H(ID_{SC} \parallel PW_1 \parallel PW_2 \parallel \cdots \parallel PW_n \parallel n \parallel k_i)$, $V_i = H(e_i \parallel RPW_i \parallel ID_{SC} \parallel k_i \parallel n)$ and $DID_{SC} = ID_{SC} \oplus H(TS_1 \parallel ID_{GW})$. The transmitted pseudo identity DID_{SC} is the dynamic name. Given the hash function property, $\mathcal{A}$ cannot extract ID_{SC} without ID_{GW}. Consequently, our scheme achieves the goal of anonymity and can withstand tracing attacks.

Online guessing attacks. In our scheme, the registration phase is executed strictly in a secure environment before deployment. We assume that $\mathcal{A}$ intercepts message transmission in the channel during the login, authentication and key exchange, password updating, and dynamic-node addition phases. $\mathcal{A}$ can obtain the messages (PTC_i, C_i, PKS_i, TS_1), (PID_j, C_{GW}, PKS_{GW}, TS_2), and (C_j, PKS_j, TS_3), (PTC_i PTC_i^{new}). Notably, the intercepted message, excluding the TS, is entirely encrypted by hash function and XOR operation. In addition, each hash function includes a minimum of two unknown numbers. Therefore, $\mathcal{A}$ cannot use online guessing attacks to guess the inputs of the hash function. In C_{GW} and Ver_{TC_j} calculation, although only one unknown input is in the function, $\mathcal{A}$ cannot guess the inputs from the dictionary without the secret key, TC_j. Therefore, our scheme can resist online guessing attacks.

Offline password guessing attacks. Offline password guessing attacks have always been a major security concern in designing password-based schemes. There are some outstanding studies trying to solve this problem, and our scheme strictly observes the rules that are described in Nam et al.'s research [67]. In this attack analysis section, $\mathcal{A}$ can use the power analysis attack to extract the information stored in the SC. Therefore, $\mathcal{A}$ obtains (e_i, V_i, PTC_i, ID_{SC}) from SC. All messages extracted by $\mathcal{A}$ are operated by hash function and XOR operation. Therefore, $\mathcal{A}$ cannot derive the sensitive information from these messages. Each message includes a minimum of two unknown inputs, as well as multiple passwords encrypted by the hash function. Therefore, $\mathcal{A}$ cannot use offline password-guessing attacks to derive the multiple passwords and the number of passwords n from the SC.

Replay attacks. We assume that $\mathcal{A}$ intercepts the messages transmitted in the communication channel and replays these messages to the receiver without any modification. A replay attack cannot work in our scheme because each entity initially checks the freshness of the TS. If the TS is not fresh, then the receiver rejects the request. Therefore, our scheme can resist replay attacks.

Man-in-the-middle attacks. Choo et al. proposed that the unknown key share attack (man-in-the-middle attack) is the most fatal security problem for any protocol [68]. We assume that $\mathcal{A}$ intercepts the messages transmitted in the communication channel and replays these messages to the receiver with a particular modification of the message. The purpose of this action of $\mathcal{A}$ is to make the receiver believe that $\mathcal{A}$ is the legitimate sender. $\mathcal{A}$ can intercept the transmitted messages via the channel. To pass authentication, $\mathcal{A}$ must compute C_i, C_{GW}, and C_j and $\mathcal{A}$ is unable to obtain (TC_i, RPW_i, k_i, TC_j, k_j) without knowing the secret number or the temporal credential of each entity. Therefore, $\mathcal{A}$ cannot obtain the right (C_i, C_{GW}, C_j) and pass authentication. Therefore, our scheme can resist man-in-the-middle attacks.

Lost password threat. According to other studies [69–71], passwords are currently not safe and are therefore vulnerable to any identity authentication. $\mathcal{A}$ can obtain the used passwords of U through numerous methods. For example, $\mathcal{A}$ can obtain user passwords from a low-security level database or by using social engineer [44, 45]. Then, $\mathcal{A}$ can use these lost passwords to pass the authentication of WSNs with the stolen SC. Once the password is lost, the scheme for WSNs encounters a considerable threat. In our scheme, multiple passwords are used to replace the unique password, which means that the legitimate user needs to input several passwords at will. The passwords, their sequence, and their number are used as key factors to authenticate the user's identity. Although $\mathcal{A}$ obtains the used passwords, he/she does not know other security factors, such as the sequence of passwords, their combination, and their number. In other schemes, if $\mathcal{A}$ obtains m passwords of the user, the probability of obtaining the correct password is described as follows:

$$P_{\mathrm{one}} = \frac{1}{m} \times P_{\mathrm{h}},$$

where we assume that the probability of using the old password is P_{h}. In our scheme, U adopts n passwords as login passwords. The probability of obtaining the correct password is

$$P_{\mathrm{multiple}} = \frac{1}{A_{\mathrm{m}}^{\mathrm{n}}} \times \frac{1}{m} \times P_{\mathrm{h}}.$$

If the lost passwords do not consist of all the multiple passwords, the probability is smaller than P_{multiple}. According to the preceding discussion, P_{multiple} is smaller than P_{one}, and $\mathcal{A}$ cannot obtain the correct multiple passwords. Therefore, our scheme can prevent the lost password threat.

D-DOS attacks. Because of the energy limitation of WSNs, D-DOS attack is one of the most detrimental threats to WSNs [3, 42, 59], this attack includes the hello flood, inputting the wrong password, and resource depletion attacks. The goal of these attacks is to deplete the resource, especially the energy of WSNs. Numerous related schemes verify the user identity in GW with several complex computations, including numerous hash functions and other operations. This authentication method costs considerable energy of WSNs if $\mathcal{A}$ starts a D-DOS attack, which is launched by persistently inputting wrong passwords persistently. Our scheme verifies the user identity by the SC without any consumption of GW. This idea can cut the spare overhead off and can validly resist the D-DOS attacks that are launched by inputting wrong passwords in the login phase.

Malicious sensor-node attack. In the dynamic-node addition phase, U can add his/her new SNs to the WSNs. If the $\mathrm{SN}^{\mathrm{new}}$ is the malicious sensor node that is employed by $\mathcal{A}$, then $\mathrm{SN}^{\mathrm{new}}$ can obtain information from other legitimate SNs and start malicious sensor-node attacks on WSNs, including Sybil, wormhole, sink hole, rushing, routing loop, and other types of attacks [1, 40]. To protect WSNs from malicious sensor-node attacks, our scheme requires the procedure of the dynamic node addition phase to be executed under the legitimate user. If someone wants to add any new SN to the WSN, the validity of the user identity must be verified. If the identity is not legitimate, the request is rejected. Therefore, our scheme can withstand malicious sensor-node attacks.

Three-factor security. Numerous related schemes adopting three security factors [20, 72, 73] usually adopt SC, password, and biometric characteristics as authenticating factors. However, biometrics present several drawbacks that are unsuitable for WSNs. Therefore, our scheme uses multiple passwords to replace the biometric characteristic. Several passwords, their sequence, and the number of passwords are used as the most important factors for verification.

Integrity of message. In our scheme, the MAC and Ver functions are used to achieve the goal of confidentiality and integrity, which are the most important properties of security [74, 75]. Upon receiving messages, the receiver verifies whether the output of the Ver function is equal to 1. If it is not equal, the receiver aborts the session and rejects the request from the sender. Therefore, if $\mathcal{A}$ modifies the message and sends it to the next entity, then the message is denied. Therefore, our scheme checks the integrity of the message.

Security performance comparison

In this section, we compare our proposed scheme with other schemes from the security aspect. The comparison shows that our scheme exhibits superior security performance to other schemes. The detailed comparison is presented in Table 1. Yes and No in this table denote that the scheme could withstand the attack or could not withstand the attack, respectively, and n/a denotes the scheme is not applicable in this comparison. The abbreviations below Table 1 denote the compared security properties [76].

Performance Analysis

In this section, we compare our proposed scheme with other schemes that are listed in Table 1. As introduced in other studies [6, 72], the overhead of several base operations, such as XOR operation, TS, and random number generation are ignored. These types of operations entail approximately no cost in comparison with the one-way hash computation and other complex computations. We believe that the communication overhead and storage overhead are of equal importance to the computational overhead. As introduced in Amin et al.'s research [76], the communication and storage overheads are analyzed in detail. Therefore, we analyze our scheme in three terms.

Reference basis

In this section, we enumerate the reference basis of WSN performance that is adopted in this paper. As described in several studies [14, 17–21, 23, 35, 73, 77–83], all protocols are compared by the number of main computations. To show the result intuitively, we unified the hash function to represent all protocol overheads. The basis of comparison is described as follows:

1. According to Nam et al.'s research [14] and Crypto++ 5.6.0 benchmarks, we know that SHA-1 takes 11.4 cycles per byte, HMAC takes 11.9 cycles per byte, and AES takes 16.9 cycles per byte under Windows Vista and Intel Core 2. Therefore, one HMAC is equal to 1.04 hash functions and one AES is almost equal to 1.5 hash functions.
2. As introduced in other studies [72, 84], one asymmetric encryption/decryption is equal to 100 symmetric encryptions/decryptions. In addition, a symmetric encryption/decryption is at least 60 times faster than a one-exponential operation.
3. According to other studies [20, 39, 72], the time to execute a fuzzy extractor is the same as for an elliptic curve point multiplication. The time for a one-way hashing operation is 0.00032 s, for a symmetric encryption/decryption operation is 0.0056 s, for a modular exponentiation operation is 0.0192 s, and for an elliptic curve point relative multiplication operation or a fuzzy extractor is 0.0171 s.
4. According to Ma's study [85], we assume one WSN that adopts MICA2 and, integrates an 8 bit 8 MHz ATmega128L processor with the voltage is 3 V, the computational electric current is 8 mA, the received electric current is 10 mA, the transmitted electric current is 27 mA, and the transmission rate is 12.4 kb/s. Therefore, the executed 0.00032 s computation needs 3 V × 8 mA × 0.00032 s = 0.00768 mJ.
5. In agreement to [6, 20], we assume that the hash output is 160 bits [86], one prime factor is 160 bits minimum, the elliptical curve output is 320 bits, and the secret parameter is at least 160 bits [87]. The TS has 32 bits; expiration time for TE, is 32 bits; the user identity ID, pseudo ID, and random nonce are 160 bits; sensor node identity ID_{SN}, GW ID_{GW}, and

Table 2. The comparison with main computations.

	cycles per byte	hash function	the time(s)	consumption(mJ)
T_H	11.4	1 T_H	0.00032	0.00768
T_A	1140	about 150 T_H	0.048	1.152
T_E	16.9	about 1.5 T_H	0.00048	0.01152
T_M	11.9	about 1 T_H	0.00032	0.00768
T_{ME}	684	about 60 T_H	0.0192	0.4608
T_{Ex}	1026	about 90 T_H	0.336	8.064
T_{EC}/T_F	609	about 53 T_H	0.0171	0.4104

doi:10.1371/journal.pone.0170657.t002

pseudo $\mathrm{ID}_{\mathrm{SN}}$ are 16 bits; encryption/decryption output is 128 bits; MAC output is 128 bits; and key setup is 128 bits.

Therefore, we can conclude all main computations in several aspects. The overhead of these main computations is described in Table 2.

The notations in this section are as follows:

T_H: hash function operation;T_A: asymmetric encryption/decryption; T_E: symmetric encryption/decryption;T_M: MAC generation/verification;T_{ME}: modular exponentiation operation; T_{Ex}: one-exponential operation; T_{EC}: elliptic curve point multiplication;T_F: fuzzy extractor.

Comparison with other schemes. In this section, we compare our proposed scheme with the schemes proposed by Nam et al. [14], A. K. Das [20], He et al. [21], Jiang et al. [19], M. L. Das [17], and Xue et al. [18] in terms of computational, communication, and storage overheads. Comparison details are described as follows.

Computational overhead. In this section, we compare the computational overhead of all schemes in several aspects. The details of the comparison of computational overhead are shown in Table 3. Notation: the numbers shown in Table 3 is a rough number that retains three decimal places.

Communication overhead. As introduced by the study [6], the transmission overhead is considerably larger than the computational overhead. The proportion of all overheads is listed as follows: 71% data transmission, 20% MAC transmission, 7% nonce transmission (for freshness), and 2% MAC and encryption computation. Therefore, analyzing the communication overhead is crucial. We assume that the receiving electric current of WSNs is

Table 3. Comparison of computational overhead.

Phase	Login		Authentication and key agreement			Total	hash	time(s)	enegy (mJ)
	U	GW	U	GW	SN				
Nam et al.	$2T_{EC}+2T_H+1T_E+1T_M$	$1T_{EC}+1T_H+2T_E+1T_M$	$2T_H$	$1T_E+1T_M$	$1T_M+2T_H+1T_E$	$3T_{EC}+7T_H+5T_E+4T_M$	$177.5T_H$	0.0568	1.363
A.K.Das	$1T_F+3T_H$	0	$6T_H$	$11T_H$	$5T_H$	$1T_F+25T_H$	$78T_H$	0.0251	0.602
He et al.	$5T_H$	$4T_H$	$3T_H$	$5T_H$	$6T_H$	$23T_H$	$23T_H$	0.00736	0.177
Jiang et al.	$3T_H$	$2T_H$	$4T_H$	$7T_H$	$5T_H$	$21T_H$	$21T_H$	0.00672	0.161
M.L.Das	$4T_H$	0	0	$4T_H$	$1T_H$	$9T_H$	$9T_H$	0.00288	0.054
XUE etal.	$2T_H$	0	$8T_H$	$11T_H$	$6T_H$	$27T_H$	$27T_H$	0.00864	0.207
Ours	$3T_H$	0	$4T_H+2T_M$	$5T_H+2T_M$	$3T_H+2T_M$	$15T_H+6T_M$	$21T_H$	0.00672	0.161

doi:10.1371/journal.pone.0170657.t003

Table 4. Comparison of communication overhead.

schemes	Total bits	Rough consumption(mJ)			
		U	GW	SN	total
Nam et al.	1264	4.463	4.645	2.206	11.314
A.K.Das	1952	4.7	9.132	3.643	17.475
He et al.	1744	5.489	6.093	4.03	15.612
Jiang et al.	1920	4.622	8.923	3.643	17.188
M.L.Das	704	2.299	3.151	0.852	6.302
XUE et al.	1744	5.489	6.093	4.03	15.612
Ours	1440	4.955	4.683	3.251	12.899

doi:10.1371/journal.pone.0170657.t004

10 mA, the transmitting electric current is 27 mA, and the rate of transmission is 12.4 kb/s. According to Ma's study [85], we assume that 1-byte transmission consumption is 3 V × 27 mA × 8 b/12400 b/s = 0.052mJ and a received byte consumption is 3 V × 10 mA × 8 b/12400 b/s = 0.019 mJ.

The details of the communication overhead of all schemes are presented in Table 4. The hello and successful signals are ignored. Notation: the number shown in Table 4 is a rough number that retains three decimal places.

Storage overhead analysis. In this section, we compare the size of stored messages with other schemes. According to the reference basis, we compute the size of stored messages in U, GW, and SN, respectively. The detailed comparison of storage overhead is presented in Table 5.

Comparison of total overhead. In this section, we compare the total overhead of schemes, including communication and computation overheads. We compare the overhead of each entity in Table 6 and compute the total overhead of all schemes. The result shows that the communication consumption is markedly larger than the computation consumption and the percentage is almost above 95% of the total overhead, and the result is the same as that in Perrig et al.'s study [6] and in common agreement with other research. Future security schemes developed will be compared based on computation overhead and communication overhead. Owing to the property of WSNs [85], the gateway station presents larger energy, higher computation performance, and larger storage performance than SN. If we want to improve the overhead of the research scheme, the most important point is improving the communication overhead of SN instead of computational overhead. Notation: the number shown in Table 6 is a rough number that retains three decimal places. The notations in this section are denoted as follows: CC: communication costs; PC: computation costs; Tot: total overhead %: the communication costs' percentage of total overhead.

Table 5. Comparison of storage overhead.

schemes	The storage overhead			
	U/SC	GW	SN	Total (bit)
Nam et al.	P, $XEID_U$, Y, ID_{GW}	y, EID_U, Y, ID_{GW}, k_{GS}	ID_{SN}, k_{GS}	1648
A.K.Das	$r_i^*, f_i, e_i, TID_i, TE_i, PTC_i$	TID_i, X_S, K_{GWN-S}, TE_i, ID_i, ID_{SN}	TC_j, ID_{SN}	3424
He et al.	r_i, PID_i, TE_i, PTC_i	SID_j, $H(PW_j)$, $K_{GWN-S,}$ K_{GWN-U}	TC_j, ID_{SN}	1184
Jiang et al.	r, TID_i, $TE_{i,}$ PTC_i	TID_i, TE_i, ID_i, K_{GWN-S}, ID_{SN}	TC_j, ID_{SN}	2080
M.L.Das	ID_i, $H(PW_i)$, N_i, x_a	ID_i, K, N_i, x_a, S_N	S_N, x_a	1472
XUE et al.	ID_i, H(H(PW_i)), TE_i, PTC_i	$K_{GWN-S,}$ K_{GWN-U}, ID_{SN}	TC_j, ID_{SN}	1024
Ours	PTC_i, V_i, e_i, ID_{GW},	PK_{GW}, PID_j, ID_{GW}, ID_{SC}	PTC_J, PK_j	1328

doi:10.1371/journal.pone.0170657.t005

Table 6. Comparison of total consumption.

schemes	Rough total consumption(mJ)										
	U			GW			SN			total	%
	CC	PC	Tot	CC	PC	Tot	CC	PC	Tot		
Nam et al.	4.463	0.864	5.327	4.645	0.465	5.11	2.206	0.035	2.241	12.678	89.24
A.K.Das	4.7	0.476	5.176	9.132	0.084	9.216	3.643	0.038	3.681	18.073	96.69
He et al.	5.489	0.061	5.55	6.093	0.069	6.162	4.03	0.046	4.076	15.788	98.89
Jiang et al.	4.622	0.054	4.676	8.923	0.069	8.992	3.643	0.038	3.681	17.349	99.07
M.L.Das	2.299	0.03	2.329	3.151	0.03	3.181	0.852	0.008	0.86	6.37	98.93
XUE et al.	5.489	0.077	5.566	6.093	0.084	6.177	4.03	0.046	4.076	15.819	98.69
Ours	4.955	0.069	5.024	4.883	0.054	4.937	3.251	0.038	3.289	13.25	98.41

doi:10.1371/journal.pone.0170657.t006

Conclusion

In this paper, we designed a temporal credential-based mutual authentication with a multiple-password scheme for WSNs. Through comparison with other schemes, we have proven that our scheme exhibits better security performance than the other schemes. Moreover, our scheme can withstand related attacks, including the lost password threat. The discussion in this paper proves that our scheme entails relatively small consumption. The analysis shows that the communication consumption's percentage of total overhead is almost above 95% and it is markedly larger than the computational consumption. Therefore, we will compare future security schemes based on computational overhead and communication overhead.

Supporting Information

S1 Table. The security comparison with other schemes. This table illustrates the security comparison with other schemes. The comparison show that our scheme has better security performance than others.
(DOCX)

S2 Table. The comparison with main computations. This table illustrates the main computations in the authentication scheme for wireless sensor networks.
(DOCX)

S3 Table. Comparison of computational overhead. This table illustrates the computational overhead comparison with other schemes. The comparison shows that our scheme has better performance than others in computational overhead.
(DOCX)

S4 Table. Comparison of communication overhead. This table illustrates the communication overhead comparison with other schemes. The comparison shows that our scheme has better performance than others in communication overhead.
(DOCX)

S5 Table. Comparison of storage overhead. This table illustrates the storage overhead comparison with other schemes. The comparison shows thatour scheme has better performance than others in storage overhead.
(DOCX)

S6 Table. Comparison of total consumption. This table illustrates the comparison with other schemes. The detailed comparison shows that the communication overhead accounts for the

majority of total overhead.
(DOCX)

Acknowledgments

This research has no sponsors of any kind and we would like to thank the kind colleagues from the Laboratory of Information Security at Lanzhou University for their assistance with this paper. Finally, we sincerely thank the anonymous referees for their constructive feedback.

Author Contributions

Conceptualization: XL RZ.

Data curation: XL.

Formal analysis: XL.

Investigation: QL.

Methodology: XL.

Writing – original draft: XL.

Writing – review & editing: RZ QL.

References

1. Yang G, Chen W, Cao X. The security of Wireless sensor networks: Sciences Press; 2010.
2. Liu X, Shen Y, Li S, Chen F, editors. A fingerprint-based user authentication protocol with one-time password for wireless sensor networks. Sensor Network Security Technology and Privacy Communication System (SNS & PCS), 2013 International Conference on; 2013: IEEE.
3. Nguyen KT, Laurent M, Oualha N. Survey on secure communication protocols for the Internet of Things. Ad Hoc Networks. 2015.
4. Chong C-Y, Kumar SP. Sensor networks: evolution, opportunities, and challenges. Proceedings of the IEEE. 2003; 91(8):1247–56.
5. Zhang N. Research on Wireless sensor network security technology: Southwest Jiaotong University Press; 2010.
6. Perrig A, Szewczyk R, Tygar JD, Wen V, Culler DE. SPINS: Security protocols for sensor networks. Wireless networks. 2002; 8(5):521–34.
7. Camaraa C, Peris-Lopeza P, Tapiadora JE. Security and Privacy Issues in Implantable Medical Devices: A Comprehensive Survey.
8. Miorandi D, Sicari S, De Pellegrini F, Chlamtac I. Internet of things: Vision, applications and research challenges. Ad Hoc Networks. 2012; 10(7):1497–516.
9. Atzori L, Iera A, Morabito G. The internet of things: A survey. Computer networks. 2010; 54(15):2787–805.
10. Kumar JS, Patel DR. A survey on Internet of Things: security and privacy issues. International Journal of Computer Applications. 2014; 90(11).
11. Roman R, Alcaraz C, Lopez J, Sklavos N. Key management systems for sensor networks in the context of the Internet of Things. Computers & Electrical Engineering. 2011; 37(2):147–59.
12. Wang Y, Attebury G, Ramamurthy B. A survey of security issues in wireless sensor networks. 2006.
13. Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. Communications magazine, IEEE. 2002; 40(8):102–14.
14. Nam J, Choo K-KR, Han S, Kim M, Paik J, Won D. Efficient and Anonymous Two-Factor User Authentication in Wireless Sensor Networks: Achieving User Anonymity with Lightweight Sensor Computation. 2015.

15. Watro R, Kong D, Cuti S-f, Gardiner C, Lynn C, Kruus P, editors. TinyPK: securing sensor networks with public key technology. Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks; 2004: ACM.

16. Wong KH, Zheng Y, Cao J, Wang S, editors. A dynamic user authentication scheme for wireless sensor networks. Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006 IEEE International Conference on; 2006: IEEE.

17. Das ML. Two-factor user authentication in wireless sensor networks. Wireless Communications, IEEE Transactions on. 2009; 8(3):1086–90.

18. Xue K, Ma C, Hong P, Ding R. A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks. Journal of Network and Computer Applications. 2013; 36 (1):316–23.

19. Jiang Q, Ma J, Lu X, Tian Y. An efficient two-factor user authentication scheme with unlinkability for wireless sensor networks. Peer-to-Peer Networking and Applications. 2014:1–12.

20. Das AK. A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks. Peer-to-Peer Networking and Applications. 2014:1–22.

21. He D, Kumar N, Chilamkurti N. A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks. Information Sciences. 2015.

22. Khan MK, Alghathbar K. Cryptanalysis and security improvements of 'two-factor user authentication in wireless sensor networks'. Sensors. 2010; 10(3):2450–9. doi: 10.3390/s100302450 PMID: 22294935

23. Sun D-Z, Li J-X, Feng Z-Y, Cao Z-F, Xu G-Q. On the security and improvement of a two-factor user authentication scheme in wireless sensor networks. Personal and ubiquitous computing. 2013; 17 (5):895–905.

24. Choo KKR. Secure Key Establishment. Advances in Information Security. 2008; 41.

25. Diffie W, Hellman M. New directions in cryptography. IEEE Transactions on Information Theory. 1976; 22(6):644–54.

26. Bellare M, Rogaway P, editors. Entity Authentication and Key Distribution. International Cryptology Conference on Advances in Cryptology; 1993.

27. Bellare M, Rogaway P, editors. Provably Secure Session Key Distribution—The Three Party Case. Proceedings of the twenty-seventh annual ACM symposium on Theory of computing; 1995.

28. Bellare M, Pointcheval D, Rogaway P. Authenticated Key Exchange Secure against Dictionary Attacks: Springer Berlin Heidelberg; 2012. 139–55 p.

29. Choo KKR, Boyd C, Hitchcock Y, Maitland G. On Session Identifiers in Provably Secure Protocols 2004. 351–66 p.

30. Choo KKR. A Proof of Revised Yahalom Protocol in the Bellare and Rogaway (1993) Model1. Computer Journal. 1993; 773(5):110–25.

31. Choo KKR, Boyd C, Hitchcock Y. On Session Key Construction in Provably-Secure Key Establishment Protocols: Springer Berlin Heidelberg; 2005. 116–31 p.

32. Choo KKR, Boyd C, Hitchcock Y. The importance of proofs of security for key establishment protocols ☆: Formal analysis of Jan—Chen, Yang—Shen—Shieh, Kim—Huh—Hwang—Lee, Lin—Sun—Hwang, and Yeh—Sun protocols. Computer Communications. 2006; 29(15):2788–97.

33. Choo KKR, Boyd CA, Hitchcock Y. Examining Indistinguishability-Based Proof Models for Key Establishment Protocols: Springer Berlin Heidelberg; 2005. 585–604 p.

34. Choo KKR, Hitchcock Y. Security Requirements for Key Establishment Proof Models: Revisiting Bellare—Rogaway and Jeong—Katz—Lee Protocols: Springer Berlin Heidelberg; 2005. 429–42 p.

35. Amin R, Islam SH, Biswas G, Khan MK, Li X. Cryptanalysis and enhancement of anonymity preserving remote user mutual authentication and session key agreement scheme for e-health care systems. Journal of medical systems. 2015; 39(11):1–21.

36. Das ML, Saxena A, Gulati VP, Phatak DB. A novel remote user authentication scheme using bilinear pairings. Computers & Security. 2006; 25(3):184–9.

37. Delac K, Grgic M, editors. A survey of biometric recognition methods. Electronics in Marine, 2004 Proceedings Elmar 2004 46th International Symposium; 2004: IEEE.

38. Gorman LO. Comparing passwords, tokens, and biometrics for user authentication. Proceedings of the IEEE. 2003; 91(12):2021–40.

39. He D, Kumar N, Lee J-H, Sherratt R. Enhanced three-factor security protocol for consumer USB mass storage devices. Consumer Electronics, IEEE Transactions on. 2014; 60(1):30–7.

40. Cayirci E, Rong C. Security in wireless ad hoc and sensor networks: John Wiley & Sons; 2008.

41. Krawczyk H, Canetti R, Bellare M. HMAC: Keyed-hashing for message authentication. 1997.

42. Raymond DR, Midkiff SF. Denial-of-service in wireless sensor networks: Attacks and defenses. Pervasive Computing, IEEE. 2008; 7(1):74–81.

43. Goodrich MT, Tamassia R. Introduction to computer security: Pearson; 2011.

44. Mitnick K. Ghost in the Wires: My Adventures as the World's Most Wanted Hacker: Little, Brown; 2011.

45. Mitnick KD, Simon WL. The art of deception: Controlling the human element of security: John Wiley & Sons; 2011.

46. Choo KKR. On the Security Analysis of Lee, Hwang & Lee (2004) and Song & Kim (2000) Key Exchange / Agreement Protocols. Informatica. 2006; 17(4):467–80.

47. jia C. Wireless sensor network security research [D]: Zhejiang University; 2008.

48. Bakhtiari S, Safavi-Naini R, Pieprzyk J. Cryptographic hash functions: A survey. Centre for Computer Security Research, Department of Computer Science, University of Wollongong, Australie. 1995.

49. Damgård IB, editor A design principle for hash functions. Advances in Cryptology—CRYPTO'89 Proceedings; 1990: Springer.

50. Rogaway P, Shrimpton T, editors. Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. Fast Software Encryption; 2004: Springer.

51. Maymounkov P, Mazieres D. Kademlia: A peer-to-peer information system based on the xor metric. Peer-to-Peer Systems: Springer; 2002. p. 53–65.

52. Yang C-N, Wang D-S. Property analysis of XOR-based visual cryptography. Circuits and Systems for Video Technology, IEEE Transactions on. 2014; 24(2):189–97.

53. Javidi B, Bernard L, Towghi N. Noise performance of double-phase encryption compared to XOR encryption. Optical Engineering. 1999; 38(1):9–19.

54. Zeng P, Choo KKR, Sun DZ. On the security of an enhanced novel access control protocol for wireless sensor networks. IEEE Transactions on Consumer Electronics. 2010; 56(2):566–9.

55. Bellare M, Pointcheval D, Rogaway P. Authenticated Key Exchange Secure against Dictionary Attacks: Springer Berlin Heidelberg; 2000. 139–55 p.

56. Choo KKR, Boyd CA, Hitchcock Y, Maitland GM. Complementing Computational Protocol Analysis with Formal Specifications. Ifip Advances in Information & Communication Technology. 2004; 173:129–44.

57. Messerges TS, Dabbish E, Sloan RH. Examining smart-card security under the threat of power analysis attacks. Computers, IEEE Transactions on. 2002; 51(5):541–52.

58. Kocher P, Jaffe J, Jun B, editors. Differential power analysis. Advances in Cryptology—CRYPTO'99; 1999: Springer.

59. Newsome J, Shi E, Song D, Perrig A, editors. The sybil attack in sensor networks: analysis & defenses. Proceedings of the 3rd international symposium on Information processing in sensor networks; 2004: ACM.

60. Eschenauer L, Gligor VD, editors. A key-management scheme for distributed sensor networks. Proceedings of the 9th ACM conference on Computer and communications security; 2002: ACM.

61. Perrig A, Stankovic J, Wagner D. Security in wireless sensor networks. Communications of the ACM. 2004; 47(6):53–7.

62. Burrows M, Abadi M, Needham RM, editors. A logic of authentication. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences; 1989: The Royal Society.

63. Ge M, Choo KKR, Wu H, Yu Y. Survey on key revocation mechanisms in wireless sensor networks. Journal of Network & Computer Applications. 2016; 63(C):24–38.

64. Ge M, Choo KKR, editors. A Novel Hybrid Key Revocation Scheme for Wireless Sensor Networks. International Conference on Network and System Security, Nss; 2014.

65. Zeng P, Cao Z, Choo KKR, Wang S. Security weakness in a dynamic program update protocol for wireless sensor networks. IEEE Communications Letters. 2009; 13(6):426–8.

66. Choo KKR, Nam J, Won D. A mechanical approach to derive identity-based protocols from Diffie—Hellman-based protocols. Information Sciences. 2014; 281:182–200.

67. Nam J, Choo KKR, Paik J, Won D. Cryptanalysis of Server-Aided Password-Based Authenticated Key Exchange Protocols. International Journal of Security & Its Applications. 2013; 7(2):47–58.

68. Choo KKR, Boyd C, Hitchcock Y. Errors in Computational Complexity Proofs for Protocols: Springer Berlin Heidelberg; 2005. 624–43 p.

69. Matthews T. Passwords are not enough. Computer Fraud & Security. 2012; 2012(5):18–20.

70. Morris R, Thompson K. K.: Password security: A case history. Communications of the Acm. 1979; 22 (11):594–7.

71. Bonneau J, Herley C, Oorschot PCV, Stajano F, editors. The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes. 2012 IEEE Symposium on Security and Privacy; 2012.

72. Lee C-C, Chen C-T, Wu P-H, Chen T-Y. Three-factor control protocol based on elliptic curve cryptosystem for universal serial bus mass storage devices. Computers & Digital Techniques, IET. 2013; 7 (1):48–56.

73. Li C-T, Hwang M-S. An efficient biometrics-based remote user authentication scheme using smart cards. Journal of Network and computer applications. 2010; 33(1):1–5.

74. Subashini S, Kavitha V. A survey on security issues in service delivery models of cloud computing. Journal of network and computer applications. 2011; 34(1):1–11.

75. Padmavathi DG, Shanmugapriya M. A survey of attacks, security mechanisms and challenges in wireless sensor networks. arXiv preprint arXiv:09090576. 2009.

76. Amin R, Biswas G. A secure light weight scheme for user authentication and key agreement in multigateway based wireless sensor networks. Ad Hoc Networks. 2016; 36:58–80.

77. Wang D, Wang N, Wang P, Qing S. Preserving privacy for free: Efficient and provably secure two-factor authentication scheme with user anonymity. Information Sciences. 2015.

78. Zhou J, Cao Z, Dong X, Xiong N, Vasilakos AV. 4S: A secure and privacy-preserving key management scheme for cloud-assisted wireless body area network in m-healthcare social networks. Information Sciences. 2015; 314:255–76.

79. Zhao Z. An efficient anonymous authentication scheme for wireless body area networks using elliptic curve cryptosystem. Journal of medical systems. 2014; 38(2):1–7.

80. Yuan J-J. An enhanced two-factor user authentication in wireless sensor networks. Telecommunication Systems. 2014; 55(1):105–13.

81. Delgado-Mohatar O, Fúster-Sabater A, Sierra JM. A light-weight authentication scheme for wireless sensor networks. Ad Hoc Networks. 2011; 9(5):727–35.

82. Chatterjee K, De A, Gupta D. A Secure and Efficient Authentication Protocol in Wireless Sensor Network. Wireless Personal Communications. 2015; 81(1):17–37.

83. Wang D, Wang P. Understanding security failures of two-factor authentication schemes for real-time applications in hierarchical wireless sensor networks. Ad Hoc Networks. 2014; 20:1–15.

84. Schneier B. Applied cryptography: protocols, algorithms, and source code in C: john wiley & sons; 2007.

85. Ma C. Key Management for Heterogeneous Sensor Networks: National Defense Industry Press; 2012. 206–9 p.

86. PUB F. Secure hash standard. Public Law. 1995;100:235.

87. Brouwer AE, Pellikaan R, Verheul ER. Doing more with fewer bits. Advances in Cryptology-ASIACRYPT'99: Springer; 1999. p. 321–32.