Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 8539674, 11 pages
https://doi.org/10.1155/2018/8539674

Research Article

Cryptanalysis and Security Enhancement of Three Authentication Schemes in Wireless Sensor Networks

Wenting Li[1,2] **Bin Li**[1] **Yiming Zhao**[1] **Ping Wang**[1,3] **and Fushan Wei**[2]

[1]*School of Software and Microelectronics, Peking University, Beijing 100871, China*
[2]*State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China*
[3]*National Engineering Research Center for Software Engineering, Peking University, Beijing 100871, China*

Correspondence should be addressed to Ping Wang; pwang@pku.edu.cn

Received 4 April 2018; Accepted 28 May 2018; Published 5 July 2018

Academic Editor: Joseph Liu

Nowadays wireless sensor networks (WSNs) have drawn great attention from both industrial world and academic community. To facilitate real-time data access for external users from the sensor nodes directly, password-based authentication has become the prevalent authentication mechanism in the past decades. In this work, we investigate three foremost protocols in the area of password-based user authentication scheme for WSNs. Firstly, we analyze an efficient and anonymous protocol and demonstrate that though this protocol is equipped with a formal proof, it actually has several security loopholes been overlooked, such that it cannot resist against smart card loss attack and violate forward secrecy. Secondly, we scrutinize a lightweight protocol and point out that it cannot achieve the claimed security goal of forward secrecy, as well as suffering from user anonymity violation attack and offline password guessing attack. Thirdly, we find that an anonymous scheme fails to preserve two critical properties of forward secrecy and user friendliness. In addition, by adopting the "perfect forward secrecy (PFS)" principle, we provide several effective countermeasures to remedy the identified weaknesses. To test the necessity and effectiveness of our suggestions, we conduct a comparison of 10 representative schemes in terms of the underlying cryptographic primitives used for realizing forward secrecy.

1. Introduction

Currently, wireless sensor networks (WSNs) have become one of the most standard services employed in commercial and industrial applications and proved to be a leading area of research [1–3]. Like many advanced technologies, the original appliance of WSNs can be found in military and heavy industrial applications. In the 1950s, the first modern WSN—the Sound Surveillance System (SOSUS)—is developed by the United States Military and used for detecting Soviet submarines [4]. Nowadays, WSNs thrive in industrial and consumer applications, including machine health monitoring, environmental sensing, natural disaster prevention, and health care monitoring [5–7].

A wireless sensor network generally includes a central gateway node (*GWN*, so-called base station), a large number of circulating, self-directed and low powered devices named sensor nodes, and a set of end users. The *GWN* acts as a bridge between WSNs and the other networks and also a powerful data managing and processing center. Sensor nodes are multifunctional, energy efficient devices and are spatially distributed over the networks for caringly collecting, processing, and transferring data.

In many critical applications, remote users are usually keen on real-time accessing with sensor nodes [8, 9], yet if data queries are carried out by the gateway node, efficiency and accuracy might not be guaranteed over the long transmission path between *GWN* and the sensors. Accordingly, password-based user authentication proves to be a proper solution for this issue as its security, simplicity, and portability [10–12]. That is, users are first authenticated by remote sensor nodes before being permitted to access data.

In 2006, Wong et al. [13] proposed the first password-based authentication scheme for wireless sensor networks that allows legitimate users to query sensitive information at every sensor of the network. However, shortly after this protocol was presented, Tseng et al. [14] and Das et al. [15] pointed out that Wong et al.'s scheme [13] is vulnerable

to replay attack, forgery attack, and node capture attack separately, then an enhanced one based on smart card was firstly proposed by Das et al. [15]. Unfortunately, Khan and Alghathbar [16], Chen et al. [17], and Yeh et al. [18] pointed out some weaknesses in Das et al.'s scheme, such as suffering from impersonation attack, insider attack, and the violation of user anonymity and key agreement. Then some improvements are made in these works.

However, in 2013, Shi and Gong [19] found that Yeh et al.'s scheme [18] cannot achieve mutual authentication and user anonymity, then they proposed an efficient ECC-based authentication scheme for WSNs. At the same time, Khan and Alghathbar's protocol [16] was also proven insecure against insider attack, smart card loss attack and forgery attack by Vaidya et al. [20] and Chen et al.'s scheme [17] was shown as vulnerable to impersonation attack, replay attack and *GWN* by passing attack in [21]. Later, Choi et al. [22] demonstrated that Shi and Gong's scheme [19] is vulnerable to smart card loss attack and an enhanced scheme was given in [22]. Meanwhile, Xue et al. [23] presented a temporal credential-based two-factor (i.e., smart card and password) authentication scheme for WSNs. Although their scheme retains many admirable properties, there are some weaknesses being found by researchers [8, 11, 24], such as offline password guessing, insider, impersonation, and tracking attacks.

Quite recently, Li et al. [25] analyzed the security of Jiang et al.'s scheme [11] (an improvement based on [23]) and showed that their scheme suffers from user friendliness issue, desynchronization problem, and is inapplicable for WSN environments. Then a new scheme was proposed in [25]; however, in this paper, we reveal that Li et al.'s scheme [25] still fails to eliminate the security pitfalls of smart card loss attack and the violation of forward secrecy. At the same time, we find that the newly proposed schemes by Amin et al. [8] and Wu et al. [9] are prone to the same security defects with Li et al.'s scheme [25].

From the above analysis, it can be seen that many of the previous protocols are not much satisfactory. On the one hand, this is because the lack of necessary principles. Some principles that have been proven are still ignored in the design of the protocol, such as user anonymity principle [7] and perfect forward secrecy principle [26]. On the other hand, the protocol designers usually do not follow unified evaluation criteria, and they tend to emphasize the advantages of their new designed protocol, but ignore its inadequacies. Besides reporting the security flaws in [8, 9, 25], we also provide effective countermeasures and refinements to overcome these pitfalls, accordingly, examine the necessary of our suggestions.

Contributions. In this work, we mainly review and analyze three state-of-the-art authentication protocols proposed by Li et al. [25], Amin et al. [8], and Wu et al. [9]. And reveal that all these three schemes suffer from smart card loss issue and cannot achieve forward secrecy. Then we suggest several possible countermeasures to overcome these pitfalls. We also provide a comparison of 10 representative schemes for wireless sensor networks which emphatically considered how and with what technology did they realize forward secrecy. This illustrates the necessity and effectiveness of our suggestions and provides a better understanding of the exiting schemes.

Table 1: Notations and abbreviations.

Symbol	Description
U_i	i^{th} user
S_j	j^{th} remote sensor node
GWN	gateway node
$\mathscr{A}$	malicious attacker
ID_i, SID_j	identity of U_i and S_j
PW_i	password of U_i
x, K	the secret key of gateway node GWN
SC	a smart card
$\oplus$	the XOR operation
$\|$	the concatenation operation
$h(\cdot)$	one-way hash function
$\longrightarrow$	the public channel
$\Longrightarrow$	the secure channel

Organization. The remainder of this paper is organized as follows. Section 2 reviews and demonstrates the pitfalls of Li et al.'s scheme. Section 3 cryptanalyzes Amin et al.'s protocol with proper countermeasures over discovered flaws. Section 4 describes the weaknesses of Wu et al.'s protocol and compares 10 representative schemes. The conclusion is made in Section 5.

2. Cryptanalysis of Li et al.'s Scheme

Earlier in 2018, Li et al. [25] presented a three-factor anonymous and efficient authentication scheme for wireless sensor networks. Although their scheme has many attractive properties, such as the provision of user anonymity and local password change, it still fails to attain many of the claimed goals. In this section, we will demonstrate that though Li et al. try to settle the user friendliness issue of Jiang et al.'s scheme [11], their solution leads to offline dictionary attack. And we also observe that Li et al.'s scheme cannot preserve forward secrecy, which is the most crucial goal for WSNs.

2.1. Review of Li et al.'s Scheme. In this subsection, we briefly revisit Li et al.'s scheme [25]. For ease of description, some intuitive notations and abbreviates are listed in Table 1 and will appear throughout this paper. Their scheme includes three main phases: registration, login and authentication, and password change. We will follow their presentations as close as possible.

2.1.1. Registration Phase. Before the registration phase of Li et al.'s [25] scheme, the gateway node defines a finite cyclic group $\mathbb{G} = < P >$ of order a large prime number n. This group could be an elliptic curve group, or it could be a prime order subgroup of $\mathbb{Z}_p^*$. Then GWN chooses two random numbers x,

K as its master secret key and computes $X = xP$ as its public key. Ultimately, *GWN* publishes $\{E(F_p), G, P, X\}$ and stores x, K securely.

Sensor Registration Phase. *GWN* chooses an identity SID_j and computes the secret key $K_S = h(SID_j \parallel K)$ for each sensor node. Then, *GWN* embeds $\{SID_j, K_S\}$ in the memory of S_j and deploys it in the particular area.

User Registration Phase. When a user U_i aims to acquire the sensitive information of remote sensor nodes, the following procedure is carried out by U_i firstly.

(1) U_i chooses an identity ID_i, a password PW_i, and a nonce a_i and calculates $RPW_i = h(PW_i \parallel a_i)$. Then U_i imprints his/her biometric b_i on a specific device.

(2) $U_i \Longrightarrow GWN$: $\{ID_i, RPW_i, b_i\}$.

(3) Once obtaining U_i's registration request, *GWN* generates a random codeword $c_i \in C$ and computes $F(c_i, b_i) = (\alpha, \delta)$, where $\alpha = h(c_i)$ and $\delta = c_i \oplus b_i$. *GWN* further computes $A_i = h(ID_i \parallel RPW_i \parallel c_i)$ and $B_i = h(ID_i \parallel K) \oplus h(RPW_i \parallel c_i)$ and keeps $\{\alpha, \delta, A_i, B_i, X, f(\cdot)\}$ into a new smart card *SC*. At the same time, *GWN* stores ID_i in its database.

(4) $GWN \Longrightarrow U_i : \{\alpha, \delta, A_i, B_i, X, f(\cdot)\}$.

(5) When receiving the smart card, U_i stores a_i into it.

2.1.2. Login and Authentication Phase. In this phase, the following steps are performed by U_i, S_j, and *GWN* as well as negotiating a session key.

(1) U_i inserts *SC* into a card reader and inputs b_i^* on a specific device. Then *SC* computes $c_i^* = f(\delta \oplus b_i^*) = f(c_i \oplus (b_i \oplus b_i^*))$ and checks whether $h(c_i^*) \stackrel{?}{=} \alpha = h(c_i)$. If not, *SC* terminates the session. Otherwise, *SC* asks U_i to input ID_i and PW_i and computes $A_i^* = h(ID_i \parallel h(PW_i \parallel a_i) \parallel c_i^*)$. Then *SC* checks whether $A_i^* \stackrel{?}{=} A_i$. If it does not hold, *SC* rejects the session. Otherwise, *SC* chooses two random numbers r_i and s and then calculates $M_1 = B_i \oplus h(h(PW_i \parallel a_i) \parallel c_i^*)$, $M_2 = sP$, $M_3 = sX = sxP$, $M_4 = ID_i \oplus M_3$, $M_5 = M_1 \oplus r_i$, $M_6 = h(ID_i \parallel r_i) \oplus SID_j$, and $M_7 = h(M_1 \parallel SID_j \parallel M_3 \parallel r_i)$.

(2) $U_i \longrightarrow GWN$: $\{M_2, M_4, M_5, M_6, M_7\}$.

(3) Upon receiving the login request, *GWN* computes $M_3 = xM_2 = xsP$ and $ID_i = M_4 \oplus M_3$ and verifies if ID_i is in the database. If not, the request is aborted. Otherwise, *GWN* computes $M_1 = h(ID_i \parallel K)$, $r_i = M_5 \oplus M_1$, $SID_j = M_6 \oplus h(ID_i \parallel r_i)$, and $M_7^* = h(M_1 \parallel SID_j \parallel M_3 \parallel r_i)$ and checks whether $M_7^* \stackrel{?}{=} M_7$. If it does not hold, *GWN* terminates the session. Otherwise, *GWN* selects a random number r_g and computes $K_S = h(SID_j \parallel K)$, $M_8 = ID_i \oplus K_S$, $M_9 = r_g \oplus h(ID_i \parallel K_S)$, $M_{10} = r_g \oplus r_i$, and $M_{11} = h(ID_i \parallel SID_j \parallel K_S \parallel r_i \parallel r_g)$.

(4) $GWN \longrightarrow S_j : \{M_8, M_9, M_{10}, M_{11}\}$.

(5) When receiving the message, S_j computes $ID_i = M_8 \oplus K_S$, $r_g = h(ID_i \parallel K_S) \oplus M_9$, $r_i = r_g \oplus M_{10}$, and $M_{11}^* = h(ID_i \parallel SID_j \parallel K_S \parallel r_i \parallel r_g)$ and checks whether $M_{11}^* \stackrel{?}{=} M_{11}$. If not, the session is rejected. Otherwise, S_j selects a random number r_j and computes $M_{12} = r_j \oplus K_S$, $SK_j = h(ID_i \parallel SID_j \parallel r_i \parallel r_g \parallel r_j)$, and $M_{13} = h(K_S \parallel SK_j \parallel r_j)$.

(6) $S_j \longrightarrow GWN$: $\{M_{12}, M_{13}\}$.

(7) After getting the response message, *GWN* computes $r_j = M_{12} \oplus K_S$, $SK = h(ID_i \parallel SID_j \parallel r_i \parallel r_g \parallel r_j)$, and $M_{13}^* = h(K_S \parallel SK \parallel r_j)$ and checks whether $M_{13}^* \stackrel{?}{=} M_{13}$. If not, *GWN* aborts the session. Otherwise, *GWN* calculates $M_{14} = M_1 \oplus r_g$, $M_{15} = r_i \oplus r_j$, and $M_{16} = h(ID_i \parallel SK \parallel r_g \parallel r_j)$.

(8) $GWN \longrightarrow U_i : \{M_{14}, M_{15}, M_{16}\}$.

(9) When U_i receiving the response message, *SC* computes $r_g = M_{14} \oplus M_1$, $r_j = M_{15} \oplus r_i$, $SK_i = h(ID_i \parallel SID_j \parallel r_i \parallel r_g \parallel r_j)$, and $M_{16}^* = h(ID_i \parallel SK_i \parallel r_g \parallel r_j)$ and checks whether $M_{16}^* \stackrel{?}{=} M_{16}$. If it does not hold, U_i terminates the connection. Otherwise, U_i and S_j establish a connection with a session key.

2.2. Cryptanalysis of Li et al.'s Scheme. A concrete and concise adversarial model is essential for a good design of user authentication scheme in wireless sensor networks. Though lacking of specification in Li et al.'s scheme [25], the following assumptions about the adversary's capabilities are implicitly made in [25]:

(1) Two communication channels exist: one is a secure, or a private channel which is mainly used for registration; another is a public channel which acts on login and authentication phases. As in the conventional authentication protocols, the adversary $\mathscr{A}$ is modeled to have full control of the public channel; i.e., $\mathscr{A}$ can eavesdrop, intercept, and modify and redirect any transmitted messages between the communication parties [3, 6].

(2) The user-memorable identities and passwords are of low entropy and can be offline enumerated by $\mathscr{A}$ at the same time within polynomial time.

(3) When considering truly multifactor authentication (i.e., the scheme is secure even if one or more factors are cracked [10]), it is rational to assume that $\mathscr{A}$ may (i) learn a victim's password such as phishing or shoulder surfing attacks, (ii) extract the secret parameters in the lost smart card by side-channel attack, or (iii) obtain a victim's biometric information via malicious device, but cannot achieve all. Otherwise, it is a trivial case.

(4) To delineate the critical feature of forward secrecy, $\mathscr{A}$ is allowed to corrupt any valid entities to obtain its longterm secret key(s). In addition, previous session key(s) may be revealed by $\mathscr{A}$ as a possible reason of improper erasure [10, 27].

It is worth noting that the above adversarial model, following the existing works in [3, 6, 7, 10, 28], is one of the few ones that apply to multifactor authentication in WSNs. For the sake of user friendliness, many protocols allow their users to select his/her identity *ID* and password *PW*. However, the user usually chooses easy-to-remember identity (e.g., email, phone number) and password, which are of low entropy ($|\mathscr{D}_{id}| \leq |\mathscr{D}_{pw}| \leq 10^6$ [29, 30]) and can be offline enumerated by $\mathscr{A}$ within polynomial time. Besides, assumption (3) specifies truly three-factor security and assumption (4) is used to capture the crucial notion of forward secrecy when *GWN* or any sensor node S_j is corrupted. In the following sections, our analysis will take account of these four assumptions.

2.2.1. Smart Card Loss Attack. In [25], Li et al. pointed out that Jiang et al.'s scheme [11] lacks timely detection mechanism, which means once a user inputs wrong identity or password unintentionally, the system will remain executing the following login and authentication phases. Undoubtedly, this interaction process will bring extra cost. In reality, it is a common accident as users usually involve in countless applications and manage various pairs of identity and password [7]. To solve this problem, Li et al.'s scheme [25] inserts a verification item $A_i = h(ID_i \parallel h(PW_i \parallel a_i) \parallel c_i)$ in the smart card for the purpose of providing timely detection and performing password change without any interaction with the *GWN*. However, their modification goes back to the "security-usability" balance problem proposed by Huang et al. [12]; that is, it realizes local password change but brings offline dictionary attack. We illustrate this attack as below.

Step 1. $\mathscr{A}$ chooses a pair (ID_i^*, PW_i^*) from $\mathscr{D}_{id} \times \mathscr{D}_{pw}$, where $\mathscr{D}_{id}$ represents the identity space and $\mathscr{D}_{pw}$ represents the password space.

Step 2. $\mathscr{A}$ computes $A_i^* = h(ID_i^* \parallel h(PW_i^* \parallel a_i) \parallel c_i)$, where a_i is extracted from the victim's smart card and c_i can be obtained by computing $c_i = f(\delta \oplus b_i)$ with the help of malicious device.

Step 3. $\mathscr{A}$ verifies the correctness of (ID_i^*, PW_i^*) pair by checking whether the computed A_i^* equals the extracted A_i.

Step 4. $\mathscr{A}$ repeats the above Steps 1 ~ 3 until the right values are found.

Besides the previous reasonable assumption (3), it should be pointed out that, in the registration phase of Li et al.'s scheme [25], U_i imprints his/her biometric information b_i on a specific device and simply submits the plain-text b_i to *GWN*. Then, *GWN* employs the fuzzy commitment technology [31] and the generated α to compute δ. In such situation, if a privileged insider, e.g., the administrator, has learned the user's biometric information, she is able to complete the above offline guessing attack. Of course, she is able to impersonate the victim to login other applications as biometric characteristics cannot be easily changed.

For another, in order to realize user friendliness, most password-based authentication schemes (e.g., [8, 9, 11]) allow users to choose his/her own *ID* and *PW*, and Li et al.'s scheme is no exception. However, users usually tend to choose easy-to-remember and thus of low entropy identities and passwords, so that it is reasonable to make the assumption (2) that $\mathscr{A}$ can offline enumerate all the (ID, PW) pairs within polynomial time. The running time of the above attack procedure is $\mathcal{O}(2T_H \times |\mathscr{D}_{id}| \times |\mathscr{D}_{pw}|)$, where $|\mathscr{D}_{id}|$ denotes the number of identities, $|\mathscr{D}_{pw}|$ denotes the number of passwords, and T_H is the running time for Hash operation. Since $|\mathscr{D}_{id}|$ and $|\mathscr{D}_{pw}|$ are very limited in practice (e.g., $|\mathscr{D}_{id}| \leq |\mathscr{D}_{pw}| \leq 10^6$ [29, 30]), our above attack is meaningful and poses a real challenge to user authentication protocols for wireless sensor networks.

2.2.2. The Violation of Forward Secrecy. WSNs are generally deployed in security-critical applications, such as battlefield surveillance and health care monitoring [7, 27, 32, 33]. The sensor nodes at risk had been driven: on one hand, due to the unattended environments and low-cost considerations, it is easier for an adversary $\mathscr{A}$ to focus on sensors access to breakthrough success; on the other hand, sensors often perform extremely sensitive tasks and thus, they preserve sensitive information and exhibit greater attack surface. Consequently, sensor nodes are more vulnerable to serious attacks, so that an admired authentication scheme for WSNs ought to be guaranteed against node capture attack.

Unfortunately, Li et al.'s scheme [25] cannot resist against this severe node capture attack. Let us consider the following scenarios. In case a sensor node S_k has been compromised by an adversary $\mathscr{A}$ and the stored secret key K_S can be extracted. This assumption is sound as made in assumption (4) and it is also implicitly described in Li et al.'s scheme [25]. With the extracted K_S, $\mathscr{A}$ can successfully obtain the previous session key between S_k and any user U_i, as follows.

Step 1. Eavesdrop and intercept the message $\{M_8, M_9, M_{10}, M_{11}\}$ sending from *GWN* to S_k.

Step 2. Compute $ID_i = M_8 \oplus K_S$, $r_g = h(ID_i \parallel K_S) \oplus M_9$, and $r_i = r_g \oplus M_{10}$.

Step 3. Intercept the message $\{M_{12}, M_{13}\}$ sending from S_k to *GWN*.

Step 4. Compute $r_k = M_{12} \oplus K_S$.

Step 5. Intercept the login message $\{M_2, M_4, M_5, M_6, M_7\}$ sending from U_i to *GWN*.

Step 6. Compute $SID_k = M_6 \oplus h(ID_i \parallel r_i)$.

Step 7. Compute the previous session key as $SK = h(ID_i \parallel SID_k \parallel r_i \parallel r_g \parallel r_k)$.

There are some points to be noted regarding the aforementioned attack. Firstly, the reason why we add Steps 5 and 6 is that these two steps are conducive to check the parameters though $\mathscr{A}$ has already known SID_k. Then, it is not hard to see that $\mathscr{A}$ only needs to eavesdrop over the public channel with simple computations to complete the aforementioned attack procedure. Consequently, the desirable security goal of perfect forward secrecy (PFS) cannot be attained by Li et al.'s scheme.

Despite considerable attention has been paid to forward secrecy issue, many prior works still explicitly or implicitly use an incorrect computation for the session key(s) (e.g., [8, 9, 21, 34]). This is mainly due to the violation of the "PFS principle" suggested in [26]: (i) public-key techniques are indispensable; (ii) at least two exponentiation operations are conducted on server side. Though Ma et al. [26] emphasize this principle on client-server architecture, after careful analysis, we find this "PFS principle" is suitable for WSN environments (i.e., three-party environment). In this cases,

we will take *GWN* and sensors as server side, while keeping users as client.

Accordingly, elliptic curves cryptosystem (ECC) is a reasonable choice for overcoming this pitfall, whereas in their original scheme [25] Li et al. employ this mechanism to greatly attain user anonymity. To make a precisely modification, we assume r_i to be r_iP and r_j to be r_jP, where point P is a generator mentioned before and r_i, r_j are two random numbers chosen by U_i and S_j separately. Note that *GWN* has no need to be involved in negotiating the session key. Then in this way, the session key can be recalculated as $SK = h(ID_i \parallel SID_j \parallel r_iP \parallel r_jP \parallel r_ir_jP)$. As it is generated by session-variant random numbers r_i and r_j and computationally infeasible to guess r_ir_jP from transmitted message due to discrete logarithm problem, Li et al.'s scheme [25] will be secure against node capture attack and provide forward secrecy perfectly after slight modifications.

2.2.3. Mistakes in the Proof. The emergence of BAN logic opens up a new chapter in the proof of user authentication protocol [35, 36]; it can not only be used to prove whether the protocol achieves some desired goals, but also be employed to find some defects in the protocol. However, there still are some problems in the application of BAN logic. On the one hand, BAN logic cannot prove whether the protocol achieves all security goals and desirable properties. For example, it cannot prove that the protocol resists against parallel session attack, denial-of-service attack, node capture attack, etc. On the other hand, the analysis of BAN logic depends on some basic assumptions and the initial hypotheses. If the initial hypotheses was not sound, the formal analysis will lead to erroneous conclusions.

In the formal proof of Li et al.'s scheme [25] with BAN logic, there are several minor problems. Firstly, Li et al. add a new logic rule, *session keys rule*:

$$\frac{P \mid\equiv \#(X), P \mid\equiv Q \mid\equiv X}{P \mid\equiv P \mid \overset{K}{\longleftrightarrow} Q} \quad (1)$$

However, it is better to explain the calculation method of K and the key role of X in K. Otherwise, we cannot derive that P believes P and Q share K from the upper part of the equation.

Secondly, we suggest that the initiative premises p13 and p14, i.e., $GWN| \equiv S_j \Longrightarrow r_j \oplus K$ and $S_j| \equiv GWN \Longrightarrow r_g \oplus h(ID_i \parallel K)$, respectively, should be derived from the translation messages, but not in the premises. Finally, they may ignore some details in the formal proof, such as in the D5, it is better to add $GWN| \equiv \#r_i$, which we cannot find in the assumption or derive from the front. It also can be seen that the correctness of the protocol cannot be guaranteed only by using the formal proof.

3. Cryptanalysis of Amin et al.'s Scheme

Recently, Amin et al. [8] proposed a lightweight protocol for IoT-enabled devices for cloud computing environments. The private information is usually stored in distributed cloud servers (e.g., sensors), so that distributed nodes are confronted with the same security threats of sensors in wireless sensor networks. After careful analysis, we find that though equipped with a formal proof and exhibiting great application prospects, Amin et al.'s scheme still cannot resist against smart card loss attack and also fail to provide user anonymity and forward secrecy.

3.1. Review of Amin et al.'s Scheme. Here we briefly review the scheme proposed by Amin et al. [8], an enhancement over Xue et al.'s scheme [37] and Chuang et al.'s scheme [38].

3.1.1. Registration Phase. The registration phase of Amin et al.'s scheme can be divided into cloud server registration and user registration.

Cloud Server Registration Phase. In this phase, any cloud server S_j sends a self-chosen identity and random number pair $\{SID_j, d\}$ to control server (CS). Then CS chooses a random number y, computes $P_j = h(SID_j \parallel d)$, $BS_j = h(P_j \parallel y)$, and responds $\{BS_j\}$ to S_j securely. Finally, S_j stores $\{BS_j, d\}$ in the memory.

User Registration Phase. Firstly, a user U_i chooses his/her identity ID_i, password PW_i, and two random numbers $< b_1, b_2 >$. Then U_i computes $A_i = h(PW_i \parallel b_1)$, $PID_i = h(ID_i \parallel b_2)$, and $bb_i = b_2 \oplus A_i$ and sends $\{A_i, PID_i\}$ to CS via secure channel. Upon receiving the registration request, the CS computes $C_i = h(A_i \parallel PID_i)$, $D_i = h(PID_i \parallel x)$, and $E_i = D_i \oplus A_i$ with its secret key. Finally, CS replies U_i a smart card with $\{C_i, E_i, h(\cdot)\}$. After getting the smart card SC, U_i computes $DP = h(ID_i \parallel PW_i) \oplus b_1$ and records DP, bb_i into it.

3.1.2. Login and Authentication Phase. In order to access remote server resources, a legal user U_i inserts his/her smart card into a card reader and inputs ID_i, PW_i. Then the following steps are performed:

(1) SC computes $b_1 = DP \oplus h(ID_i \parallel PW_i)$, $A_i = h(PW_i \parallel b_1)$, $b_2 = bb_i \oplus A_i$, $PID_i = h(ID_i \parallel b_2)$, and $C_i^* = h(A_i \parallel PID_i)$ and verifies whether $C_i^* \overset{?}{=} C_i$. If so, SC selects a random number N_i and computes $D_i = E_i \oplus A_i$, $G_i = h(PID_i \parallel SID_j \parallel N_i \parallel T_i \parallel D_i)$, $F_i = D_i \oplus N_i$, and $Z_i = SID_j \oplus h(D_i \parallel N_i)$, where SID_j is S_j's identity chosen by U_i and T_i is the current timestamp. Otherwise, SC terminates the session.

(2) $U_i \longrightarrow S_j : \{G_i, F_i, Z_i, PID_i, T_i\}$.

(3) Upon receiving the login request, S_j checks whether $|T_j - T_i| < \Delta T$ holds, where T_j is S_j's current timestamp and ΔT is the expected valid time interval. If it does not hold, S_j rejects the connection. Otherwise, S_j produces a random number N_j and computes $J_i = BS_j \oplus N_j$, $K_i = h(N_j \parallel BS_j \parallel G_i \parallel T_j)$.

(4) $S_j \longrightarrow CS : \{J_i, K_i, P_j, G_i, F_i, Z_i, PID_i, T_i, T_j\}$.

(5) Once receiving the message from S_j, CS first checks the validity of time interval $|T_{cs} - T_j| < \Delta T$. If the verification holds, CS continues to compute $D_i = h(PID_i \parallel x)$, $N_i = F_i \oplus D_i$, $SID_j = Z_i \oplus h(D_i \parallel N_i)$, and $G_i^* = h(PID_i \parallel SID_j \parallel N_i \parallel T_i \parallel D_i)$ and checks whether $G_i^* \overset{?}{=} G_i$. If either of the above

verification fails, CS terminates the procedure. Otherwise, CS keeps on calculating $BS_j = h(P_j \parallel y)$, $N_j = BS_j \oplus J_i$, and $K_i^* = h(N_j \parallel BS_j \parallel G_i \parallel T_j)$ and verifies whether the computed K_i^* equals the received one. If not, CS aborts the session. Otherwise, CS chooses a random number N_{cs} and computes $P_{cs} = N_j \oplus N_{cs} \oplus h(N_i \parallel D_i)$, $R_{cs} = N_i \oplus N_{cs} \oplus h(BS_j \parallel N_j)$, $SK_{cs} = h(N_i \oplus N_j \oplus N_{cs})$, $Q_{cs} = h((N_j \oplus N_{cs}) \parallel SK_{cs})$, and $V_{cs} = h((N_i \parallel N_{cs}) \parallel SK_{cs})$.

(6) $CS \longrightarrow S_j : \{P_{cs}, R_{cs}, Q_{cs}, V_{cs}\}$.

(7) While receiving the message from CS, S_j computes $W_j = h(BS_j \parallel N_j)$, $N_i \oplus N_{cs} = R_{cs} \oplus W_j$, $SK_j = h(N_i \oplus N_j \oplus N_{cs})$, and $V_{cs}^* = h((N_i \parallel N_{cs}) \parallel SK_j)$ and checks the condition $V_{cs}^* \stackrel{?}{=} V_{cs}$ holds or not. If it does not hold, S_j terminates the connection. Otherwise, S_j sends $\{P_{cs}, Q_{cs}\}$ to U_i via public channel.

(8) After receiving the response message from S_j, U_i computes $L_i = h(N_i \parallel D_i)$, $N_j \oplus N_{cs} = P_{cs} \oplus L_i$, $SK_i = h(N_i \oplus N_j \oplus N_{cs})$, and $Q_{cs}^* = h((N_j \oplus N_{cs}) \parallel SK_{cs})$ and verifies whether $Q_{cs}^* \stackrel{?}{=} Q_{cs}$. If so, U_i successfully authenticates S_j and CS and establish a session key $SK_i = SK_j = SK_{cs}$.

3.2. Cryptanalysis of Amin et al.'s Scheme. The four assumptions made in Section 2.2 are also explicitly employed in Amin et al.'s work [8] when they analyze the security of Xue et al.'s scheme [37] and Chuang et al.'s scheme [38] and proof the safety of their scheme. Consequently, our following discussions will base on these four assumptions.

3.2.1. No Provision of User Anonymity. Nowadays, privacy concerns are attracting more and more attention among governments, organizations, and individuals, and anonymous privacy-preserving authentication protocols are of particular interest. This is because the violation of user anonymity, say the leakage of some user-specific (static) information, may facilitate a malicious adversary to track the victim's current activities and login history [7, 39]. Generally, there are two kinds of user anonymity attributes, basic and advanced [7]: (i) user ID protection, which means $\mathscr{A}$ cannot obtain the real ID of the user; (ii) user untraceability, which means $\mathscr{A}$ is unable to tell who the user is and distinguish whether two communications are coming from the same user. In wireless sensor networks, the latter notion has been widely adopted (e.g., [40–42]), so does Amin et al.'s scheme.

In 2014, Das et al. [43] firstly introduced a "dynamic ID technique" to achieve user anonymity: a user's real ID is concealed in the session-variant pseudonym identities. Subsequently, many schemes (e.g., [25, 44, 45]) follow this technique, which are so-called "dynamic ID" schemes, and Amin et al.'s scheme [8] falls into this category. However, after careful analysis, we find that Amin et al.'s scheme cannot achieve user anonymity in practice. To be specific, in the login phase of their scheme, Amin et al. try to compute a pseudonym identity $PID_i = h(ID_i \parallel b_2)$ as a dynamic identity. On one hand, PID_i is specific to the legitimate user U_i; on the other hand, PID_i is kept static and transmitted in plain of all the U_i's login messages $\{G_i, F_i, Z_i, PID_i, T_i\}$.

Accordingly, this specific value PID_i can be seen as U_i's "identification", and thus $\mathscr{A}$ can exploit it to identify and track U_i in the whole system. To conduct the aforementioned attack, an adversary $\mathscr{A}$ only needs to eavesdrop the transmission channel without other contact operations and computations. This well serves to show the violation of user anonymity on Amin et al.'s scheme [8], thereby contradicting their claim.

3.2.2. Smart Card Loss Attack. Amin et al. [8] showed that, in Xue et al.'s protocol [37], users' passwords can be offline guessed once $\mathscr{A}$ has somehow obtained (lost or stolen) the victim's smart card and extracted the stored secret information. Then Amin et al. attempt to overcome this pitfall in their new proposed scheme. However, precisely the same deficiency still exits in Amin et al.'s enhanced version. Let us consider the following scenario, suppose that $\mathscr{A}$ has obtained the secret parameters $\{C_i, E_i, DP, bb_i, h(\cdot)\}$ stored in U_i's smart card (e.g., by side-channel attack [46–48] and reverse engineering technique [49]), which is reasonable under assumption (3). Then $\mathscr{A}$ can conduct the following procedure to guess U_i's password.

Step 1. Choose a pair of (ID_i^*, PW_i^*) from the identity space $\mathscr{D}_{id}$ and password space $\mathscr{D}_{pw}$.

Step 2. Compute $b_1 = DP \oplus h(ID_i^* \parallel PW_i^*)$, $A_i = h(PW_i^* \parallel b_1)$, $b_2 = bb_i \oplus A_i$, $PID_i = h(ID_i^* \parallel b_2)$, and $C_i^* = h(A_i \parallel PID_i)$.

Step 3. Verify whether the computed C_i^* equals the extracted C_i.

Step 4. Repeat Steps 1, 2, and 3 until finding the correct values.

Let $|\mathscr{D}_{id}|$ and $|\mathscr{D}_{pw}|$ denote the size of $\mathscr{D}_{id}$ and $\mathscr{D}_{pw}$, and the time complexity of the aforementioned attack is $\mathcal{O}(4T_H \times |\mathscr{D}_{id}| \times |\mathscr{D}_{pw}|)$, which is linearly associated with the running time of Hash operation and can be finished in a few days as the limited size of $|\mathscr{D}_{id}| \le |\mathscr{D}_{pw}| \le 10^6$ [29, 30].

Further, according to assumption (1), $\mathscr{A}$ is capable of eavesdropping and intercepting the normal (previous successful) login message $\{G_i, F_i, Z_i, PID_i, T_i\}$ between U_i and S_j over the public channel. It is fair to assume that $\mathscr{A}$ has already obtained the correct value of PID_i, then Step 2 might be changed to compute b_1, A_i, b_2, and PID_i^* and compared the computed PID_i^* with the intercepted PID_i in Step 3. In this way, the time complexity of the above procedure reduces to $\mathcal{O}(3T_H \times |\mathscr{D}_{id}| \times |\mathscr{D}_{pw}|)$, where the exclusive and concatenation operations are too small to overlook.

Note that both of the above two attacks are carried out *offline* without any interaction with the control server. Hence, there is no way for CS to find abnormality and the adversary $\mathscr{A}$ can impersonate U_i at anytime until CS revokes the victim's smart card. All in all, our analysis demonstrates the feasibility of smart card loss attack on Amin et al.'s scheme [8].

3.2.3. The Violation of Forward Secrecy. As mentioned in Section 2.2.2, Amin et al.'s scheme [8] also subjects to node

capture attack. In such cases, the captured nodes may enable an adversary to compromise communications between other noncaptured nodes or obtain previous session keys. We will show this pitfall in this subsection. Assume that a malicious adversary $\mathcal{A}$ has compromised a cloud server S_k and extracted the secret parameters $\{BS_j, d\}$ stored in its memory, $\mathcal{A}$ can recover the previous session key as follows.

Step 1. Intercept the message $\{J_i, K_i, P_k, G_i, F_i, Z_i, PID_i, T_i, T_k\}$ sending from S_k to CS.

Step 2. Compute $N_k = BS_k \oplus J_i$, where BS_k is extracted from the compromised node S_k.

Step 3. Intercept the message $\{P_{cs}, R_{cs}, Q_{cs}, V_{cs}\}$ sending from CS to S_k.

Step 4. Compute $N_i \oplus N_{cs} = R_{cs} \oplus h(BS_k \parallel N_k)$.

Step 5. Compute the session key $SK = h(N_k \oplus N_i \oplus N_{cs})$.

In light of N_k and $N_i \oplus N_{cs}$ which are all correct values, $\mathcal{A}$ manages to find the previous session key. Hence, the desirable property of forward secrecy can not be attained by Amin et al.'s scheme [8]. Similar to Li et al.'s scheme [25], this also due to the violation of "PFS principle". Except the ECC technique mentioned before, we suggest this issue to be well addressed by introducing another high-efficiency technique, i.e., Chebyshev polynomials semigroup property (so-called chaotic maps).

For this property, given x, $T_u(x)$, and $T_v(x)$, it is intractable to find $T_u T_v(x)$, where x is a variable and u, v denote the integer degree [45]. Assume the control server chooses and writes a variable value x in each user's smart card in the registration phase. Then we slightly modify the random numbers N_i to be $T_{N_i}(x)$ and N_k to be $T_{N_k}(x)$, and thus the session key can be calculated as $SK = h(T_{N_i}(x) \parallel T_{N_k}(x) \parallel T_{N_i} T_{N_k}(x))$. For higher security, it is better to involve other secret parameters such as ID_i, SID_j. In this way, the improvement of Amin et al.'s scheme [8] can achieve perfect forward secrecy based on computational Diffie-Hellman problem.

3.2.4. Mistakes in the Proof. Similarly, the security proof in Amin et al.'s scheme [8] does not capture realistic security threat. There are three main reasons: (1) The error of initial hypothesis. In the formal proof of Amin et al.'s scheme [8], they make an assumption A11: $S_j| \equiv U_i \overset{SK}{\longleftrightarrow} S_j$, which is the same as **Goal 3**. This demonstrates that the proof of **Goal 3** is not necessary. (2) The wrong usage of logic rules. We take Step S2 as an example. This step is based on the message meaning rule and derives that S_j believes U_i said N_i from A11 and S1. However, according to the message meaning rule, we cannot obtain this conclusion from A11. Hence, A11 should be changed to $S_j| \equiv U_i \overset{D_i}{\longleftrightarrow} S_j$. (3) Using undefined new rules. Amin et al. [8] also employ a new *session keys rule*, but they did not give a definition of the new rule.

4. Cryptanalysis of Wu et al.'s Scheme

In this section, we will review and analyze Wu et al.'s scheme [9], which is a lightweight and relatively robust two-factor authentication scheme for wireless medical sensor networks. In [9], Wu et al. have found some security pitfalls in historical schemes and attempted to overcome all these flaws in the new proposed one. Besides, Wu et al. [9] use NS-3, a simulation tool to prove the security of their proposed protocol. Note that, the simulation process can only prove the validity of their protocol, including the viable communication between the sensor node and the user, the probable communication time, system size, etc. However, it can not prove whether their protocol resists against various known attacks. In the following section, we find Wu et al.'s improved scheme still fails to attain the most important goal of forward secrecy and is prone to user friendliness issue.

4.1. Review of Wu et al.'s Scheme. This subsection briefly reviews Wu et al.'s [9] scheme, which involves four critical phases: registration, login, authentication and password change, and a previous initialization. We simplify initialization phase in the registration phase.

4.1.1. Registration Phase. Initially, GWN is equipped with an identity GID and its own secret key G. The registration phase is further divided into sensor node registration and user registration.

Sensor Node Registration Phase. Each sensor node S_j chooses an identity SID_j and sends to GWN via a secure channel. Then GWN decides to deploy it in a sensor set numbered N_l and computes the secret key $SG = h(SID_j \parallel G \parallel N_l)$. Finally, $\{SID_j, SG, GID\}$ is injected to the memory of S_j and (SID_j, N_l) is stored in the database of GWN.

User Registration Phase. In this phase, U_i first selects an identity ID_i, a password PW_i, and a nonce r_0, and then

(1) U_i computes $HPW_i = h(PW_i \parallel r_0)$;

(2) $U_i \Longrightarrow GWN$: $\{ID_i, HPW_i\}$;

(3) GWN checks if ID_i has already existed in the database. If so, it denies the registration request. Otherwise, GWN chooses a pseudoidentity CID_i and computes $A_1 = h(CID_i \parallel GID \parallel G) \oplus HPW_i$ and $A_2 = h(ID_i \parallel G) \oplus h(ID_i \parallel HPW_i)$ and then stores ID_i in database;

(4) $GWN \Longrightarrow U_i$: a smart card contains sensitive parameters $\{A_1, A_2, CID_i, GID\}$;

(5) after receiving the message, U_i computes $A_3 = h(ID_i \parallel PW_i) \oplus r_0$ and inserts it into SC.

4.1.2. Login and Authentication Phase. U_i conducts the following procedures to access sensitive information of the target sensor S_j:

(1) U_i inputs ID_i and PW_i to the smart card. Then SC computes $r_0 = A_3 \oplus h(ID_i \parallel PW_i)$ and $HPW_i = h(PW_i \parallel r_0)$. SC chooses a random number r_u and the required sensor node SID_j and further computes $B_1 = A_1 \oplus HPW_i = h(CID_i \parallel GID \parallel G)$, $B_2 = B_1 \oplus r_u$, $B_3 = ID_i \oplus h(r_u \parallel B_1)$, and $B_4 = h(CID_i \parallel GID \parallel SID_j \parallel B_1 \parallel ID_i \parallel r_u)$.

(2) $U_i \longrightarrow GWN$: $\{CID_i, GID, SID_j, B_2, B_3, B_4\}$.

(3) When receiving the message from U_i, GWN first checks if GID is correct. If so, GWN computes $B_1 = h(CID_i \parallel GID \parallel G)$, $r_u = B_1 \oplus B_2$, and $ID_i = B_3 \oplus h(r_u \parallel B_1)$ and verifies whether ID_i is in the database and $B_4 \stackrel{?}{=} h(CID_i \parallel GID \parallel SID_j \parallel B_1 \parallel ID_i \parallel r_u)$. If either of the two verifications does not hold, GWN will terminate the session. Otherwise, GWN searches (SID_j, N_l) from the database, generates a random number r_g, and computes $SG = h(SID_j \parallel G \parallel N_l)$, $B_5 = h(SG \parallel GID) \oplus r_u$, $B_6 = h(r_u) \oplus r_g$, and $B_7 = h(SG \parallel r_u \parallel r_g)$.

(4) $GWN \longrightarrow S_j : \{SID_j, B_5, B_6, B_7\}$.

(5) Once receiving the message, the corresponding node S_j checks if SID_j is correct and computes $r_u = B_5 \oplus h(SG \parallel GID)$ and $r_g = B_6 \oplus h(r_u)$. Then S_j verifies whether $B_7 \stackrel{?}{=} h(SG \parallel r_u \parallel r_g)$. If either is incorrect, S_j rejects the session. Otherwise, S_j generates r_s and computes $sk_s = h(r_u \parallel r_g \parallel r_s)$, $B_8 = h(SG \parallel r_g) \oplus r_s$, and $B_9 = h(sk_s \parallel SID_j \parallel GID \parallel r_s)$.

(6) $S_j \longrightarrow GWN$: $\{B_8, B_9\}$.

(7) Once received the response message, GWN computes $r_s = B_8 \oplus h(SG \parallel r_g)$, $sk_g = h(r_u \parallel r_g \parallel r_s)$ and checks whether $B_9 \stackrel{?}{=} h(sk_g \parallel SID_j \parallel GID \parallel r_s)$. If so, GWN chooses a new pseudoidentity CID_i^{new} and calculates $B_{10} = h(CID_i^{new} \parallel GID \parallel G) \oplus h(r_u \parallel CID_i)$, $B_{11} = h(r_u \parallel ID_i) \oplus r_g$, $B_{12} = h(r_u \parallel r_g) \oplus r_s$, $B_{13} = h(h(ID_i \parallel G) \parallel r_s) \oplus CID_i^{new}$, and $B_{14} = h(sk_g \parallel ID_i \parallel B_{10} \parallel CID_i^{new})$.

(8) $GWN \longrightarrow U_i : \{B_{10}, B_{11}, B_{12}, B_{13}, B_{14}\}$.

(9) When receiving the response message, SC computes $r_g = B_{11} \oplus h(r_u \parallel ID_i)$, $r_s = B_{12} \oplus h(r_u \parallel r_g)$, $sk_u = h(r_u \parallel r_g \parallel r_s)$, and $CID_i^{new} = B_{13} \oplus h((A_2 \oplus h(ID_i \parallel HPW_i)) \parallel r_s)$ and verifies whether $B_{14} \stackrel{?}{=} h(sk_u \parallel ID_i \parallel B_{10} \parallel CID_i^{new})$. If it is equal, SC computes $A_1^{new} = B_{10} \oplus h(r_u \parallel CID_i) \oplus HPW_i$ and replaces (A_1, CID_i) with (A_1^{new}, CID_i^{new}).

4.2. Cryptanalysis of Wu et al.'s Scheme. Due to its simplicity and admirable provision of user anonymity, Wu et al.'s scheme [9] exhibits great application prospects, and yet there are still some security pitfalls being overlooked by Wu et al. In the following, we will demonstrate that Wu et al.'s scheme [9] has some user friendliness issue and fails to achieve the critical property of forward security.

4.2.1. No Provision of User Friendliness. According to the collected data from Dashlane [56], "we are online hoarders" that the average user maintains over 107 accounts registered to one email address and this figure will rise to 207 by 2020. This statistical shows that users are creating and virtually stashing more online account information than ever, which leads to an insanely high number of accounts to manage. In that case, freely password change is a recommended practice, for users have to reset a forgotten password (an average of 37 accounts [56]) and the fixed password is definitely vulnerable. Moreover, users may make a slip in writing passwords or identities; the rapid response and decisive action are quite necessary for a user friendly authentication protocol.

Early in 1968, Robert Miller [57] published a classic paper about response time in man-computer conversational transactions, which pointed out that "response times exceed 10 seconds will completely lose the user's attention". In this way, locally secure password change, i.e., providing an explicit and secure process to verify the correctness of user-keyed password in smart card, is essential. That is, the smart card has no need to interact with remote server in user input and password changing phases. However, as stated above, both Li et al.'s scheme [25] and Amin et al.'s scheme [8] provide local password change, but their strategies introduce new vulnerabilities-offline dictionary attack.

Back to Wu et al.'s scheme [9], there is no verifier in the smart card, which means their scheme even cannot provide timely detection mechanism and reasonable password change. Fortunately, Wang et al. [10] introduced a "fuzzy verifier" technique to effectively solve this security-usability issue. In the following, we will take Wu et al.'s scheme [9] as an example to show this strategy. Firstly, U_i submits $\{ID_i, HPW_i\}$ to GWN in the registration phase. Then GWN computes $A_i = h((h(ID_i) \oplus HPW_i)) \bmod n)$ and stores it in U_i's smart card, where n denotes the size of (ID, PW) pool and $2^6 \le n \le 2^8$. Assume $|\mathscr{D}_{id}| \approx |\mathscr{D}_{pw}| \approx 10^6$ and $n = 2^8$ [29, 30], we can be assured that there have the possibilities of $(|\mathscr{D}_{id}| \times |\mathscr{D}_{id}|)/n \approx 2^{32}$ identity and password pairs to thwart the adversary from guessing out the correct password.

The same considerations can also be applied to Li et al.'s scheme [25] and Amin et al.'s scheme [8]. The large-scale candidates will effectively frustrate $\mathscr{A}$ from random guessing the password by a brute force method as well as providing a timely detection of the mistyped identity or password.

4.2.2. The Violation of Forward Secrecy. Forward secrecy is an important property, for the unattended environment and security-critical applications in wireless sensor networks [7, 11]. In [9], Wu et al. explicitly stated that "the sensor nodes may be captured by the intruder", which accords with assumption (4) made in Section 2.2. Under this statement, we find that Wu et al.'s scheme cannot achieve the forward secrecy. Once a sensor node S_k has been compromised, the stored information SG might be obtained by $\mathscr{A}$ and the following attacks can be launched.

Step 1. Intercept the message $\{CID_i, GID, SID_k, B_2, B_3, B_4\}$ sending from U_i to GWN and the message $\{SID_k, B_5, B_6, B_7\}$ sending from GWN to S_k.

Step 2. Compute $r_u = B_5 \oplus h(SG \parallel GID)$, $r_g = B_6 \oplus h(r_u)$, where SG is extracted from the compromised node S_k.

Step 3. Intercept the message $\{B_8, B_9\}$ sending from GWN to S_k.

Step 4. Compute $r_s = B_8 \oplus h(SG \parallel r_g)$.

Step 5. Compute the session key $= h(r_u \parallel r_g \parallel r_s)$.

The above attack demonstrates that once a sensor node S_k has been captured, the previous sessions might be decoded. This is the same failure with Li et al.'s scheme [25] and Amin et al.'s scheme [8]. Besides the above two techniques

Table 2: Performance and security comparison.

Related schemes	Computation cost on log-auth phases	PFS	Key technology			
			Hash	ECC	CM*	RSA
Li et al. (2018) [25]	$21T_H + 3T_E$	×	√	√	×	×
Amin et al. (2018) [8]	$22T_H$	×	√	×	×	×
Wu et al. (2017) [9]	$34T_H$	×	√	×	×	×
Roy et al. (2017) [45]	$15T_H + 3T_C$	×	√	×	√	×
Moon et al. (2017) [50]	$11T_H + 2T_C$	×	√	×	√	×
Srinivas et al. (2017) [51]	$29T_H$	×	√	×	×	×
Das et al. (2016) [52]	$31T_H + 4T_E$	√	√	√	×	×
Chang et al. (2016) [53]	$20T_H$	×	√	×	×	×
Vaidya et al. (2016) [54]	$29T_H$	×	√	×	×	×
Odelu et al. (2015) [55]	$15T_H + 6T_M$	×	√	×	×	√

Note[1]. T_H: one-time hash operation time; T_E: elliptic curve point multiplication computation time; T_C: running time of chaotic maps; T_M: time for modular multiplication/division.
Note[2]. × means the corresponding scheme fails to achieve this property; CM denotes chaotic maps; PFS denotes perfect forward secrecy.

(ECC cryptosystem and chaotic maps), we also suggest employing some other public-key cryptography techniques, such as Pairing [58] and RSA cryptosystem. Note that when using RSA cryptosystem to achieve forward secrecy, a new temporary RSA key must be generated by user side for each session [59].

To demonstrate the necessity and effectiveness of our suggestions, we provide a comparison of 10 recently proposed schemes by assessing whether they achieve forward secrecy and what main technology do they use. The result are shown in Table 2. One can see that only Das et al.'s scheme [52] successfully provides forward secrecy. This failure is mainly due to the fact that half of them (i.e., [8, 9, 51, 54]) only use Hash operation that are virtually impossible to provide forward secrecy ("PFS principle" [26]), yet the other 4 schemes (i.e., [25, 45, 50, 55]) that make use of public-key techniques (e.g., ECC, Chaotic maps, RSA) violate the principle that the random numbers must be generated by U_i and S_j separately and cannot be transmitted over the public channel.

5. Conclusion

In this paper, we first analyze three state-of-the-art authentication schemes presented by Li et al., Amin et al., and Wu et al., which are mainly applied to realize real-time data access for security-critical wireless sensor networks. We demonstrate that although their schemes are equipped with formal proof, they still suffer from smart card loss attack and fail to achieve some important properties of forward secrecy, user anonymity, and user friendliness. Our cryptanalysis results discourage the practical application of these three schemes and reveal some challenges in designing a robust scheme for WSNs. We then suggest several possible countermeasures on account of their weaknesses and provide a comparison of 10 representative schemes in terms of forward secrecy and key technology to demonstrate the necessity of our suggestions. For the future work, a natural direction is to employ our recommended technologies and countermeasures to design robust and efficient schemes for WSNs.

Data Availability

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Conflicts of Interest

The authors have declared that no conflicts of interest exist.

Acknowledgments

This research was partially supported by the National Natural Science Foundation of China (NSFC) under Grants no. 61472016 and no. 61772548, the National Key R&D Program of China under Grants no. 2016YFB0800603 and no. 2017YFB1200700, and the Foundation of Science and Technology on Information Assurance Laboratory No. KJ-17-001.

References

[1] K.-A. Shim, "BASIS: a practical multi-user broadcast authentication scheme in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 7, pp. 1545–1554, 2017.

[2] J.-H. Bang, Y.-J. Cho, and K. Kang, "Anomaly detection of network-initiated LTE signaling traffic in wireless sensor and actuator networks based on a hidden semi-Markov model," *Computers & Security*, vol. 65, pp. 108–120, 2017.

[3] D. He, S. Zeadally, L. Wu, and H. Wang, "Analysis of handover authentication protocols for mobile wireless networks using identity-based public key cryptography," *Computer Networks*, vol. 128, pp. 154–163, 2017.

[4] E. C. Whitman, "Sosus the 'secret weapon' of undersea surveillance," *Undersea Warfare*, vol. 7, no. 2, 2005.

[5] C. M. Talbot, M. A. Temple, T. J. Carbino, and J. A. Betances, "Detecting rogue attacks on commercial wireless Insteon home automation systems," *Computers & Security*, vol. 74, pp. 296–307, 2017.

[6] X. Li, M. H. Ibrahim, S. Kumari, A. K. Sangaiah, V. Gupta, and K. R. Choo, "Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks," *Computer Networks*, vol. 129, pp. 429–443, 2017.

[7] D. Wang and P. Wang, "On the anonymity of two-factor authentication schemes for wireless sensor networks: attacks, principle and solutions," *Computer Networks*, vol. 73, pp. 41–57, 2014.

[8] R. Amin, N. Kumar, G. P. Biswas, R. Iqbal, and V. Chang, "A light weight authentication protocol for IoT-enabled devices in distributed Cloud Computing environment," *Future Generation Computer Systems*, vol. 78, pp. 1005–1019, 2018.

[9] F. Wu, X. Li, A. K. Sangaiah et al., "A lightweight and robust two-factor authentication scheme for personalized healthcare systems using wireless medical sensor networks," *Future Generation Computer Systems*, 2017.

[10] D. Wang and P. Wang, "Two birds with one stone: two-factor authentication with security beyond conventional bound," *IEEE Transactions on Dependable and Secure Computing*, 2016.

[11] Q. Jiang, J. Ma, F. Wei, Y. Tian, J. Shen, and Y. Yang, "An untraceable temporal-credential-based two-factor authentication scheme using ECC for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 76, pp. 37–48, 2016.

[12] X. Huang, X. Chen, J. Li, Y. Xiang, and L. Xu, "Further observations on smart-card-based password-authenticated key agreement in distributed systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 7, pp. 1767–1775, 2014.

[13] K. H. M. Wong, Z. Yuan, C. Jiannong, and W. Shengwei, "A dynamic user authentication scheme for wireless sensor networks," in *Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, vol. 1, pp. 244–251, Taichung, Taiwan, June 2006.

[14] H. R. Tseng, R. H. Jan, and W. Yang, "An improved dynamic user authentication scheme for wireless sensor networks," in *Proceedings of the 50th Annual IEEE Global Telecommunications Conference (GLOBECOM '07)*, pp. 986–990, November 2007.

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

[16] M. K. Khan, S. K. Kim, and K. Alghathbar, "Cryptanalysis and security enhancement of a more efficient secure dynamic id-based remote user authentication scheme," *Computer Communications*, vol. 34, no. 3, pp. 305–309, 2011.

[17] 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.

[18] H.-L. Yeh, T.-H. Chen, P.-C. Liu, T.-H. Kim, and H.-W. Wei, "A secured authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 11, no. 5, pp. 4767–4779, 2011.

[19] W. B. Shi and P. Gong, "A new user authentication protocol for wireless sensor networks using elliptic curves cryptography," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 730831, 7 pages, 2013.

[20] B. Vaidya, D. Makrakis, and H. T. Mouftah, "Improved two-factor user authentication in wireless sensor networks," in *Proceedings of the 6th Annual IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '10)*, pp. 600–606, October 2010.

[21] 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, 2016.

[22] Y. Choi, D. Lee, J. Kim, J. Jung, J. Nam, and D. Won, "Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 14, no. 6, pp. 10081–10106, 2014.

[23] 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, 2013.

[24] D. He, N. Kumar, and N. Chilamkurti, "A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks," *Information Sciences*, vol. 321, Article ID 11403, pp. 263–277, 2015.

[25] X. Li, J. Niu, S. Kumari, F. Wu, A. K. Sangaiah, and K. R. Choo, "A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments," *Journal of Network and Computer Applications*, vol. 103, pp. 194–204, 2018.

[26] C.-G. Ma, D. Wang, and S.-D. Zhao, "Security flaws in two improved remote user authentication schemes using smart cards," *International Journal of Communication Systems*, vol. 27, no. 10, pp. 2215–2227, 2014.

[27] J. Zhao, "On resilience and connectivity of secure wireless sensor networks under node capture attacks," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 3, pp. 557–571, 2017.

[28] X. Huang, Y. Xiang, E. Bertino, J. Zhou, and L. Xu, "Robust multi-factor authentication for fragile communications," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 6, pp. 568–581, 2014.

[29] D. Wang, Z. Zhang, P. Wang, J. Yan, and X. Huang, "Targeted online password guessing: an underestimated threat," in *Proceedings of the 23rd ACM Conference on Computer and Communications Security (CCS '16)*, pp. 1242–1254, October 2016.

[30] D. Wang, H. Cheng, P. Wang, X. Huang, and G. Jian, "Zipfs law in passwords," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2776–2791, 2017.

[31] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in *Proceedings of the 6th ACM Conference on Computer and Communications Security (ACM CCS '99)*, pp. 28–36, November 1999.

[32] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS '02)*, pp. 41–47, Washington, DC, USA, November 2002.

[33] T. M. Vu, R. Safavi-Naini, and C. Williamson, "Securing wireless sensor networks against large-scale node capture attacks," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communication Security (ASIACCS '10)*, pp. 112–123, April 2010.

[34] 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, 2014.

[35] M. Burrows, M. Abad, and M. Needham, "A logic of authentication," *Proceedings of the Royal Society A Mathematical, Physical and Engineering Sciences*, vol. 426, no. 1871, pp. 233–271, 1989.

[36] V. Odelu, A. K. Das, and A. Goswami, "A Secure Biometrics-Based Multi-Server Authentication Protocol Using Smart

Cards," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 1953–1966, 2015.

[37] K. Xue, P. Hong, and C. Ma, "A lightweight dynamic pseudonym identity based authentication and key agreement protocol without verification tables for multi-server architecture," *Journal of Computer and System Sciences*, vol. 80, no. 1, pp. 195–206, 2014.

[38] M.-C. Chuang and M. C. Chen, "An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1411–1418, 2014.

[39] C. Tang and D. O. Wu, "Mobile privacy in wireless networks-revisited," *IEEE Transactions on Wireless Communications*, vol. 7, no. 3, pp. 1035–1042, 2008.

[40] R. Ali, A. K. Pal, S. Kumari, M. Karuppiah, and M. Conti, "A secure user authentication and key-agreement scheme using wireless sensor networks for agriculture monitoring," *Future Generation Computer Systems*, vol. 84, pp. 200–215, 2018.

[41] J. Jung, J. Moon, D. Lee, and D. Won, "Efficient and security enhanced anonymous authentication with key agreement scheme in wireless sensor networks," *Sensors*, vol. 17, no. 3, p. 644, 2017.

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

[43] M. L. Das, A. Saxena, and V. P. Gulati, "A dynamic ID-based remote user authentication scheme," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 629–631, 2004.

[44] P. K. Dhillon and S. Kalra, "Secure multi-factor remote user authentication scheme for internet of things environments," *International Journal of Communication Systems*, vol. 30, no. 16, pp. 1–20, 2017.

[45] S. Roy, S. Chatterjee, A. K. Das, S. Chattopadhyay, S. Kumari, and M. Jo, "Chaotic map-based anonymous user authentication scheme with user biometrics and fuzzy extractor for crowdsourcing internet of things," *IEEE Internet of Things Journal*, 2017.

[46] Y. Xiao, M. Li, S. Chen, and Y. Zhang, "Stacco: differentially analyzing side-channel traces for detecting ssl/tls vulnerabilities in secure enclaves," in *Proceedings of the ACM SIGSAC Conference*, pp. 859–874, Dallas, TX, USA, October 2017.

[47] N. Veyrat-Charvillon and F. Standaert, "Generic side-channel distinguishers: improvements and limitations," in *Advances in Cryptology—CRYPTO 2011*, vol. 6841 of *Lecture Notes in Computer Science*, pp. 354–372, Springer Berlin Heidelberg, 2011.

[48] Y. Zhou, Y. Yu, F.-X. Standaert, and J.-J. Quisquater, "On the need of physical security for small embedded devices: a case study with COMP128-1 implementations in SIM cards," in *Proceedings of the FC*, vol. 7859, pp. 230–238, 2013.

[49] G. Chalupar, S. Peherstorfer, E. Poll, and J. De Ruiter, "Automated reverse engineering using lego," in *Proceedings of the 8th USENIX WOOT*, vol. 14, pp. 1–10, 2014.

[50] J. Moon, Y. Lee, J. Kim, and D. Won, "Improving an anonymous and provably secure authentication protocol for a mobile user," *Security and Communication Networks*, vol. 2017, Article ID 1378128, 13 pages, 2017.

[51] J. Srinivas, S. Mukhopadhyay, and D. Mishra, "Secure and efficient user authentication scheme for multi-gateway wireless sensor networks," *Ad Hoc Networks*, vol. 54, pp. 147–169, 2017.

[52] A. K. Das, S. Kumari, V. Odelu, X. Li, F. Wu, and X. Huang, "Provably secure user authentication and key agreement scheme for wireless sensor networks," *Security and Communication Networks*, vol. 9, no. 16, pp. 3670–3687, 2016.

[53] 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, 2016.

[54] B. Vaidya, D. Makrakis, and H. Mouftah, "Two-factor mutual authentication with key agreement in wireless sensor networks," *Security and Communication Networks*, vol. 9, no. 2, pp. 171–183, 2016.

[55] V. Odelu, A. K. Das, and A. Goswami, "An effective and robust secure remote user authenticated key agreement scheme using smart cards in wireless communication systems," *Wireless Personal Communications*, vol. 84, no. 4, pp. 2571–2598, 2015.

[56] T. L. Bras, "Online overload its worse than you thought," July 2015, https://bit.ly/2IjgkGL.

[57] R. B. Miller, "Response time in man-computer conversational transactions," in *Proceedings of the Fall Joint Computer Conference (AFIPS '68)*, pp. 267–277, San Francisco, Calif, USA, December 1968.

[58] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology—CRYPTO 2001*, vol. 2139 of *Lecture Notes in Computer Science*, pp. 213–229, Springer, 2001.

[59] D. Park, C. Boyd, and S. Moon, "Forward secrecy and its application to future mobile communications security," in *Public Key Cryptography*, vol. 1751 of *Lecture Notes in Computer Science*, pp. 433–445, Springer Berlin Heidelberg, 2000.