sensors

MDPI

Article

An Enhanced Lightweight Dynamic Pseudonym Identity Based Authentication and Key Agreement Scheme Using Wireless Sensor Networks for Agriculture Monitoring

Meriske Chen [1], Tian-Fu Lee [1,2,*] and Jiann-I Pan [2]

[1] Institute of Medical Sciences, Tzu Chi University, No. 701, Zhongyang Road, Sec. 3, Hualien 97004, Taiwan; 104325120@gms.tcu.edu.tw

[2] Department of Medical Informatics, Tzu Chi University, No. 701, Zhongyang Road, Sec. 3, Hualien 97004, Taiwan; jipan@mail.tcu.edu.tw

* Correspondence: jackytflee@mail.tcu.edu.tw; Tel.: +886-3-856-5301 (ext. 2403); Fax: +886-3-857-9409

Received: 12 December 2018; Accepted: 1 March 2019; Published: 6 March 2019

Abstract: Agriculture plays an important role for many countries. It provides raw materials for food and provides large employment opportunities for people in the country, especially for countries with a dense population. To enhance agriculture productivity, modern technology such as wireless sensor networks (WSNs) can be utilized to help in monitoring important parameters in thw agricultural field such as temperature, light, soil moisture, etc. During the monitoring process, if security compromises happen, such as interception or modification of the parameters, it may lead to false decisions and bring damage to agriculture productivity. Therefore, it is very important to develop secure authentication and key agreement for the system. Recently, Ali et al. proposed an authentication and key agreement scheme using WSNs for agriculture monitoring. However, it fails to provide user untraceability, user anonymity, and session key security; it suffers from sensor node impersonation attack and perfect forward secrecy attack; and even worse has denial of service as a service. This study discusses these limitations and proposes a new secure and more efficient authentication and key agreement scheme for agriculture monitoring using WSNs. The proposed scheme utilizes dynamic pseudonym identity to guarantee user privacy and eliminates redundant computations to enhance efficiency.

Keywords: agriculture monitoring; agriculture WSN; key agreement; dynamic identity; agriculture decision support system; lightweight authentication

1. Introduction

Agriculture plays an important role for many countries around the world. In some countries, agriculture is not only essential to provide food and raw material supply for its citizen, but also to provide large employment opportunities for its people. In some dense populated countries like India [1], Nigeria [2], and Pakistan [3], agriculture even becomes their economic backbone. Since agriculture is essential for life, when the world population is rising, the demand for agriculture products is also increasing and if there is no improvement in agriculture production, someday people may face challenges about food availability.

Food availability depends on crops productivity and many other diverse factors such as livestock, labor, sophisticated machines, etc. While other diverse factors such as livestock, labor, climates, soils, tools, and technology vary from country to country or even from farm to farm [4]; the factors related to crop productivity almost remain similar anywhere, such as whether the farms have enough water, fertilizer, temperature, light, etc. On the other hand, farmers are also facing many challenges such as

labor shortage, natural disaster (drought, flood, typhoon, etc.), land degradation, water availability, climate change, or any other stressors which might bring a decline in crop productivity. If there is no innovation to maintain or even increasing crop productivity, it is possible that someday the world might suffer from food supply shortage. If this happen, it might bring chaos in many countries and people may suffer from poverty, malnutrition, and hunger. Therefore, a good agriculture system to help in managing and facing those many challenges are important. For example, to make sure the crop succeeds, physical or environmental parameters such as temperature, luminance, humidity, wind direction, wind speed, moisture, acidity, water level, pollutant, etc. need to be routinely and constantly monitored. To make crop productivity sustainable, integrating traditional farming methods and information technology such as Wireless Sensor Networks (WSNs) in agriculture can help this necessity. Applications used for WSNs in agriculture may use different kinds of sensor devices to ones in healthcare, forests, urban areas, and the military since their monitoring range, functionality, and power supply status are different. Additionally, wireless sensor network technology for agriculture monitoring can monitor farm temperature, humidity, light, carbon dioxide, soil moisture, acidity, pests, etc., and can be used for plant epidemic monitoring and early warning systems. For the deployment, the sensors in the forest and military must consider the terrain such as rivers, valleys, etc. that are irregular. The distribution of urban sensors must consider factors such as roads and buildings. The sensors for the human body must consider body shape and portability. The sensors in farms are often arranged in rows due to the arrangement of crops. In terms of function, the sensors in forests and farms sense temperature, humidity, light, carbon dioxide, soil moisture, acidity, pests, etc. The urban sensors sense dust, air pollution, temperature, humidity, etc. In addition to sensing sound and images, sensors in the military must sometimes be able to sense toxic or chemical substances.

WSN is a network infrastructure formed by a large number of sensor nodes to wirelessly monitor physical or environmental parameters. It can constantly monitor those important parameters in agriculture and instantly give clear notifications to user/agriculture professionals if some abnormal conditions are found. Therefore, WSNs can be used to manage or even improved crop productivity by monitoring important parameters in agriculture fields that affect the growth of the crops.

Figure 1 shows the structure of agriculture monitoring system using WSNs. Sensor nodes are manually deployed in the agriculture field and then routinely collect the data from physical or environmental parameters. First, these sensor nodes automatically collect data from physical or environmental parameter in the field and then send the collected data to the gateway node via wireless technology. The gateway node then sends the collected data to the user/agriculture professional via Internet. The user/agriculture professional will use these collected data as a basis to help them in decision making. Any misleading data such as false command, data modification, or wrong parameters may lead to a false decision which in turn could bring damage to crop productivity, such as crop failure and low return. It means that these collected data need to be protected from illegal access or data modification since they play an important role in agriculture decision support systems and are very important to help user/agriculture professionals in decision making. To accommodate this purpose, a clear mechanism about how the collected data are exchanged between legal participants is needed. This study proposes a secure and efficient authentication and key agreement scheme in agriculture monitoring system to meet these necessities.

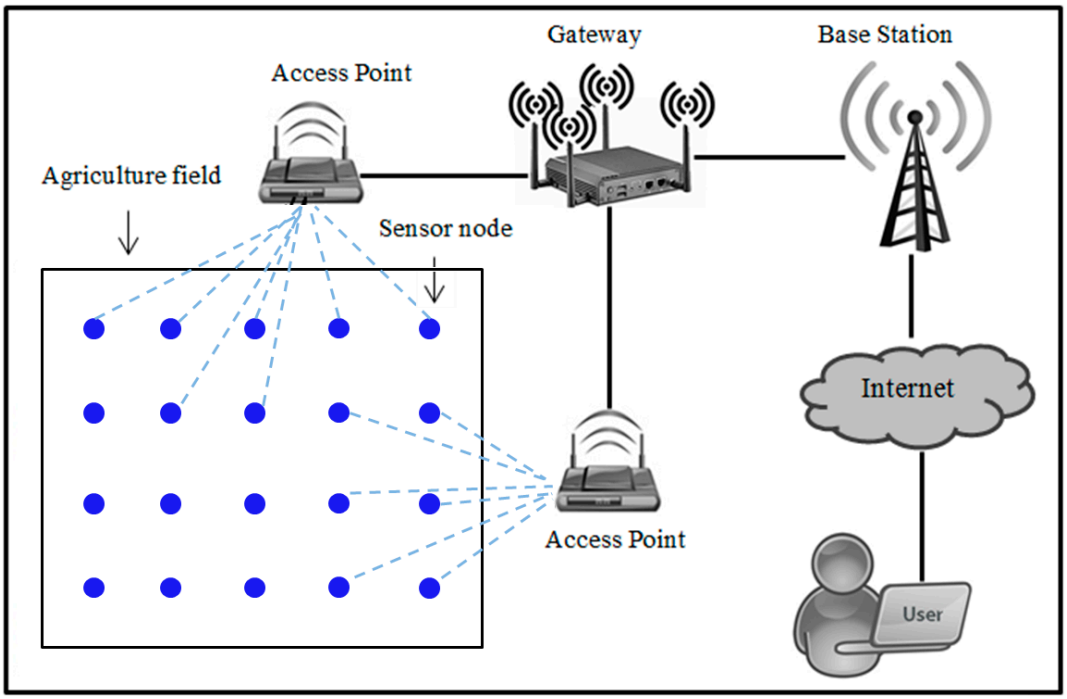

Figure 1. Agriculture monitoring system model of the proposed scheme.

1.1. Literature Reviews

Agriculture monitoring will help farmers to optimize their natural or artificial resources in their agricultural activities, which will influence their crop productivity. Some initial researches about the framework for agriculture monitoring based on WSNs [5–8] give important background about WSNs utilization in agriculture, especially about how this system can help decision support systems through better monitoring of their agriculture field. For example, Luis, et al. [5] gave a review about wireless sensor technologies for the agriculture and food industry; Jiber, et al. [6] and Anurag, et al. [7] presented a precision agriculture monitoring framework using WSNs; and Panchard, et al. [8] showed how wireless sensor technology can be used to help farming decision support. However, those existing frameworks do not explain about how those particular participants are authenticated between each other.

In recent years, more and more researchers proposed an authentication and key-agreement scheme for WSNs environment [9–26]. Most of them proposed schemes for general purposes [9–15,17,24] and few of them proposed schemes for specific purposes [16,18–20,25,26]. For example, in 2009, Pecori and Veltri [25] proposed a new alternative key agreement protocol for setting up multimedia sessions between user agents (UAs) without requiring any pre-shared key or trust relationship or PKI, and it has been implemented and integrated in a publicly available VoIP UA. In 2012, Pecori [26] developed a new protocol for establishing a security association between two peers willing to set up a VoIP or multimedia communication through the standard SIP protocol. The proposed protocol is based on the MIKEY protocol and the Diffie-Hellman algorithm for key establishment, and allows the authentication via peer certificates without using any centralized PKI. In the same year, Das, et al. [9] proposed a dynamic password-based user authentication scheme for large-scale hierarchical WSNs. It consists of three entities which are the user, base station, and cluster head. Then, in 2013, Xue et al. [10] proposed a temporal-credential-based for WSNs and Shi et al. [11] proposed a new user authentication protocol using elliptic curves cryptography for WSNs. In 2015, there was even a study about group key management for WSNs [13]. Followed these studies, there were Li et al. [12] and He et al. [15] whose showed weaknesses of Xue et al.'s scheme [10] and both of them then proposed an improved scheme. In 2015, Lee [14] showed weaknesses of Li et al.'s scheme [12] and then proposed an improved scheme

using extended chaotic maps. In the same year, Mesit and Brusta [24] proposed a secured node-to-node key agreement protocol, whose shared key is based on a symmetric encryption algorithm to solve the resource-constrained problem. Moreover, in 2016, Kumari et al. [17] mentioned weaknesses of both Li et al.'s scheme [12] and He et al.'s scheme [15] and then proposed an improved scheme using chaotic maps.

Fewer researchers discuss about authentication and key agreement scheme using WSNs for specific purposes, for example, WSNs for healthcare through body sensor networks [16,18–20], WSNs for military [21] or multimedia [22] or agriculture monitoring [23].

1.2. Motivation and Contributions

The importance of modern technology utilization in agriculture is already described in above. It also followed by how essential a secure and efficient user authentication and key-agreement scheme for agriculture monitoring using WSNs.

Dynamic pseudonym identity schemes [27–29], which were used by both Ali et al.'s scheme [23] and the proposed scheme, are quite popular and widely used in many security researches area. Dynamic pseudonym identity means that the transaction uses anonym identity and that the specific anonym identity dynamically changes in every new transaction. Anonymity is important in the agriculture area because it provides legitimate users with protection of their real identities. In the agriculture environment, we can assume that sensor nodes are put openly in the field. If a system does not provide anonymity, an attacker who targeting a particular participant can easily distinguish a transaction belongs to whom. Then he/she is able to perform attacks to his/her particular target. For example, Alice is an epidemic specialist and works on a farm. An adversary who tries to harm the farm facilities obtains Alice's identity and knows that she is responsible for assisting in monitoring the farm's temperature, humidity, and pests. The adversary may perform social engineering or dictionary attacks to obtain Alice's password or login information, and then can log in to the system for agriculture monitoring to tamper with information and damage facilities. Therefore, by using dynamic pseudonym identity, the scheme is expected to be able to provide un-traceability, privacy and user anonymity to its user. However, Ali et al.'s scheme fails to provide user anonymity and user un-traceability. It also suffered from other severe security compromises such as insider attack, sensor node attack, perfect forward secrecy, and session key security. Moreover, Ali et al.'s scheme even suffered from denial of service which happened after a user/agriculture professional has successfully updated their password.

The rest of this study is organized as follows. Section 2 reviews Ali et al.'s scheme and discusses the detail of security weaknesses in Ali et al.'s scheme. Section 3 presents the proposed scheme. Section 4 presents security analysis of the proposed scheme. Section 5 analyzes security and performances comparisons with Ali et al.'s scheme. Finally, Section 6 draws conclusions.

2. Preliminary

Although this study discusses the weaknesses of Ali et al.'s scheme, this study also recognizes the importance and advantages of their scheme, especially because of the novelty of their study. This study also followed their architecture for agriculture monitoring using WSNs, also utilizes dynamic pseudonym identity and three-factor-security, which are similar with Ali et al.'s scheme. This section consists of three sub-sections which discuss about the importance and advantage of Ali et al.'s scheme, Ali et al.'s scheme, and the weaknesses of Ali et al.'s scheme.

The notations used in Ali et al.'s scheme and in the proposed scheme are elaborated in Table 1.

Table 1. Notations of the proposed scheme.

Symbol	Description
U_i	User/agriculture professional
BS	Base station
GWN_j	Gateway node
SN_j	Sensor node
ID_i, ID_{GWN_j}, ID_{SN_j}	Identity of U_i, GWN_j, SN_j, respectively
PW_i	Password of U_i
F_i	Biometric of U_i
A_i	Shared key between BS and U_i
X	Secret key of BS
X_{BS-GWN_j}	Secret key shared between BS and GWN_j
RI_j	Secret key shared between BS and SN_j
R_U, R_{BS}, R_{GWN_j}, R_{SN_j}	Random nonce of U_i, GWN_j, SN_j, respectively
SK	Session key
T_i	Timestamp of i
ΔT_i	Time differences between T_i
E_{key}, D_{key}	Encryption, Decryption using shared *key*, respectively
$Gen(.)$	Generate function of fuzzy extractor
$Rep(.)$	Reproduce function of fuzzy extractor
$h(.)$	Hash function
$\oplus$	Exclusive OR operation
$\|\|$	Concatenation operation

2.1. The Importance and Advantage of Ali et al.'s Scheme

Ali et al. proposed a novel authentication and key agreement scheme using WSNs for agriculture monitoring. At first, they mentioned about how important agriculture is for economic systems and how WSNs technology can be utilized to face many challenges that exist in agriculture. Then, they reviewed some literature that related to security in the WSNs environment and summarized security requirements that need to be fulfilled in a scheme. Then, they presented their scheme, the security analysis and the performance evaluation of their scheme.

Compare with other existing WSNs schemes where most of them consist of three entities, Ali et al.'s scheme consist of four entities instead, which are the user/agriculture professional, base station BS, sensor node, and gateway node. The BS acts as system administrator and becomes the central entity to authenticate other entities. Without BS, other entities will never have the chance to truly trust each other in the authentication and key agreement scheme.

2.2. Ali et al.'s Scheme

In 2017, Ali et al. [23] proposed a WSNs scheme for agriculture monitoring, which consists of system setup phase; user/agriculture professional registration phase; login, authentication, and session key agreement phase; password update phase; and dynamic node addition phase.

2.2.1. System Setup Phase

To initialize the organization, the system administrator SA selects distinct identity ID_{SN_j} for m sensor node SN_j, where $1 \leq j \leq m$ and also selects distinct identity ID_{GWN_j} for each gateway node GWN_j. SA computes the shared key $\mathrm{RI_j} = \mathrm{h}\left(\mathrm{ID_{SN_j}} \| \mathrm{X}\right)$ for SN_j, where X is the secret key of the base station BS and computes the shared key $\mathrm{h}(\mathrm{X_{BS-GWN_j}})$ for $\mathrm{GWN_j}$. Finally, SA keeps $\left\{\mathrm{RI_j},\ \mathrm{ID_{SN_j}}\right\}$ into SN_j's memory and keeps $\{\mathrm{h}(\mathrm{X_{BS-GWN_j}}), \mathrm{ID_{GWN_j}}\}$ into $\mathrm{GWN_j}$'s memory. Then, SA deploys each sensor node SN_j and $\mathrm{GWN_j}$ in a target area. Here, the SA acts as BS representative to initialize the identity and the shared key with $\mathrm{SN_j}$ and $\mathrm{GWN_j}$.

2.2.2. User/Agriculture Professional Registration Phase

As shown in Figure 2, user/agriculture professional U_i needs to register to the base station BS. The following steps were executed when U_i want to become a legitimate user in this agriculture monitoring system.

Step 1: The U_i selects his/her own identity ID_i, password PW_i and imprints biometric F_i on the sensor device and then computes $Gen(F_i) = (X_F,\ P_F)$, $RPW_i = h(PW_i \| X_F)$, where $Gen(.)$ is a generate function of fuzzy extractor and $(X_F,\ P_F)$ are, respectively, secret and public keys. Now, U_i sends $\{ID_i,\ RPW_i\}$ to BS via trustworthy channel.

Step 2: When obtained the registration request from U_i, BS firstly calculates $A_i = h(ID_i \| X)$, $B_i = A_i \oplus h(RPW_i \| ID_i)$, $C_i = A_i \oplus h(B_i \| X)$ and $D_i = h(A_i \| RPW_i \| ID_i)$. Afterwards, BS issues a smartcard having parameters, i.e., $\{B_i,\ C_i,\ D_i,\ h(.)\}$ and sends it to U_i via the same channel.

Step 3: After obtaining the smartcard from BS, U_i embeds P_F and $Gen(.)$ in the memory of smartcard, i.e., $\{B_i,\ C_i,\ D_i,\ h(.),\ P_F,\ Gen(.)\}$.

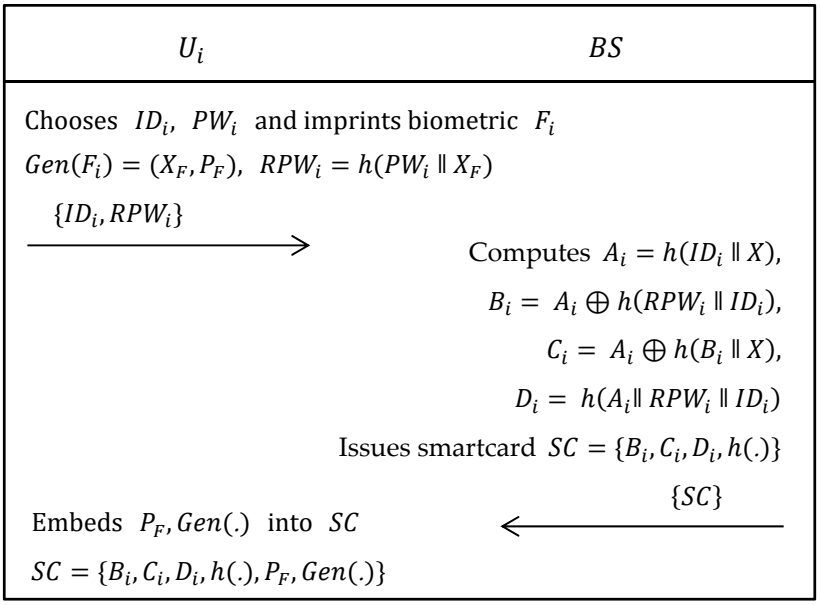

Figure 2. Registration phase of Ali et al.'s scheme.

2.2.3. Login Phase

When a user/agriculture professional U_i wants to know the environmental information such as temperature, light, humidity, soil etc., he/she has to login to access these information. As shown in Figure 3, the following steps were executed to accomplish this login phase.

Step 1: The U_i inserts his/her own smartcard into card reader and inputs ID_i, PW_i and also imprints F_i on a sensor device. Now, the card reader computes $Rep(F_i,\ P_F) = X_F^*$, $RPW_i^* = h\left(PW_i \| X_F^*\right)$, $A_i^* = B_i \oplus h(RPW_i^* \| ID_i)$, $D_i^* = h(A_i^* \| RPW_i^* \| ID_i)$, $[h(B_i \| X)]^* = C_i \oplus A_i^*$ and verifies if D_i^* equals D_i. If this verification holds then the system continues the process. Otherwise, the session is terminated.

Step 2: Now, U_i generates a random nonce R_U and enumerates $DID_i = ID_i \oplus h(B_i \| X)$, $M_1 = E_{Ai}\left(R_U \| ID_{SN_j} \| ID_{GWN_j} \| T_1\right)$, $M_2 = h(R_U \| ID_i \| T_1 \| h(B_i \| X))$ and sends $\{B_i,\ DID_i,\ M_1,\ M_2\}$ to BS via public channel.

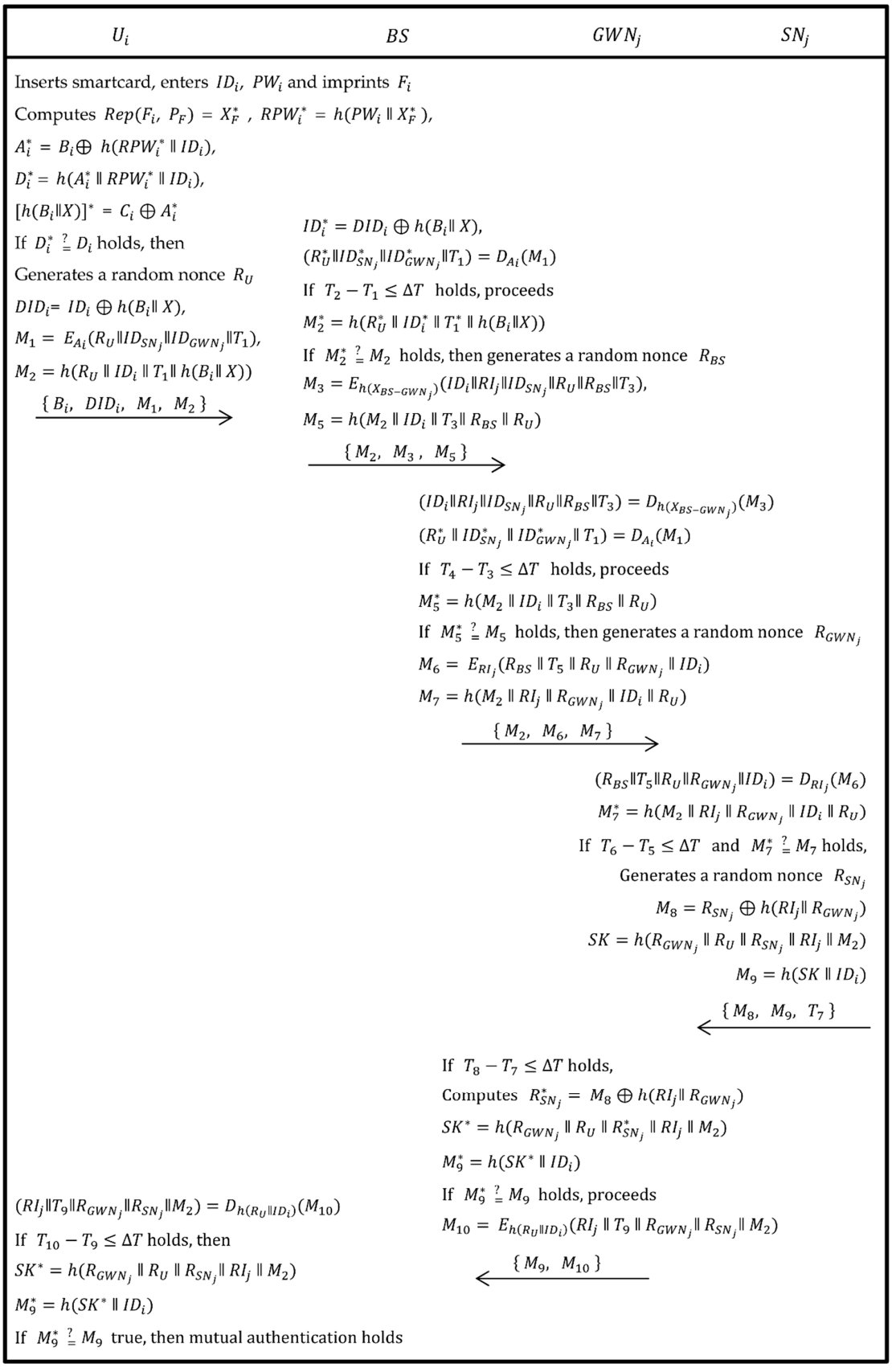

Figure 3. Login, authentication and key agreement phase of Ali et al.'s scheme.

2.2.4. Authentication and Session Key Agreement Phase

As shown in Figure 3, after login phase is successfully authenticated, the authentication and session-key phase were executed in the following steps.

Step 1: Upon obtaining the message $\{B_i,\ DID_i,\ M_1,\ M_2\}$ from U_i, the BS computes $ID_i^* = DID_i \oplus h(B_i \| X)$, $\left(R_U^* \| ID_{SN_j}^* \| ID_{GWN_j}^* \| T_1\right) = D_{Ai}(M_1)$ and checks if $T_2 - T_1 \leq \Delta T$ holds. If this does not true, then session expires. Otherwise, BS computes $M_2^* = h\left(R_U^* \| ID_i^* \| T_1^* \| h(B_i \| X)\right)$ and verifies if M_2^* equals M_2 or not. If it holds, then U_i is legal and BS goes to next step. Otherwise, the session is rejected.

Step 2: Now, the BS produces a random nonce R_{BS} and computes $M_3 = E_{h(X_{BS-GWN_j})}\left(ID_i \| RI_j \| ID_{SN_j} \| R_U \| R_{BS} \| T_3\right)$, $M_5 = h(M_2 \| ID_i \| T_3 \| R_{BS} \| R_U)$ and then sends $\{M_3,\ M_2,\ M_5\}$ to GWN_j via public channel.

Step 3: After getting request message $\{M_3,\ M_2,\ M_5\}$ from BS, the GWN_j computes $\left(ID_i \| RI_j \| ID_{SN_j} \| R_U \| R_{BS} \| T_3\right) = D_{h(X_{BS-GWN_j})}(M_3)$ and $\left(R_U^* \| ID_{SN_j}^* \| ID_{GWN_j}^* \| T_1\right) = D_{A_i}(M_1)$, then checks if two condition $T_4 - T_3 \leq \Delta T$ and $M_5^* \stackrel{?}{=} M_5$ hold. If both conditions are true then it proceeds further. Otherwise, the session is terminated.

Step 4: Now, the GWN_j generates a random nonce R_{GWN_j} and calculates $M_6 = E_{RI_j}\left(R_{BS} \| T_5 \| R_U \| R_{GWN_j} \| ID_i\right)$, $M_7 = h\left(M_2 \| RI_j \| R_{GWN_j} \| ID_i \| R_U\right)$ and then sends $\{M_2,\ M_6,\ M_7\}$ to SN_j.

Step 5: Upon obtaining the message from GWN_j, the SN_j computes $\left(R_{BS} \| T_5 \| R_U \| R_{GWN_j} \| ID_i\right) = D_{RI_j}(M_6)$, $M_7^* = h\left(M_2 \| RI_j \| R_{GWN_j} \| ID_i \| R_U\right)$ and then verifies if two conditions $T_6 - T_5 \leq \Delta T$ and $M_7^* \stackrel{?}{=} M_7$ hold. If both are true, then SN_j goes to the next step, Otherwise, the session is terminated.

Step 6: Now, the SN_j generates a random nonce R_{SN_j}, computes $M_8 = R_{SN_j} \oplus h\left(RI_j \| R_{GWN_j}\right)$, $SK = h\left(R_{GWN_j} \| R_U \| R_{SN_j} \| RI_j \| M_2\right)$, $M_9 = h(SK \| ID_i)$ and sends $\{M_8,\ M_9,\ T_7\}$ to GWN_j via public channel.

Step 7: After getting the message from SN_j, GWN_j firstly verifies if $T_8 - T_7 \leq \Delta T$ holds. If true, the process continues. Otherwise, the session expires. Then, GWN_j calculates $R_{SN_j}^* = M_8 \oplus h\left(RI_j \| R_{GWN_j}\right)$, $SK^* = h\left(R_{GWN_j} \| R_U \| R_{SN_j}^* \| RI_j \| M_2\right)$ and $M_9^* = h(SK^* \| ID_i)$, then checks if $M_9^* \stackrel{?}{=} M_9$. If it holds, the next step proceeds. Otherwise, the session is terminated.

Step 8: The GWN_j computes $M_{10} = E_{h(R_U \| ID_i)}\left(RI_j \| T_9 \| R_{GWN_j} \| R_{SN_j} \| M_2\right)$ and sends $\{M_9,\ M_{10}\}$ to U_i via public channel.

Step 9: After getting the message from GWN_j, U_i computes $\left(RI_j \| T_9 \| R_{GWN_j} \| R_{SN_j} \| M_2\right) = D_{h(R_U \| ID_i)}(M_{10})$ and verifies if $T_{10} - T_9 \leq \Delta T$ holds. If it holds, the next step proceeds.

Step 10: The U_i calculates $SK^* = h\left(R_{GWN_j} \| R_U \| R_{SN_j} \| RI_j \| M_2\right)$, $M_9^* = h(SK^* \| ID_i)$ and checks if $M_9^* \stackrel{?}{=} M_9$ holds. If it holds, mutual-authentication and session-key agreement holds.

2.2.5. Password Updates or Change Phase

In Ali et al.'s password update or change phase, user U_i modifies his/her password without intervention with the base station. As shown in Figure 4, the following steps were executed to update or change password.

Step 1: The U_i inserts his/her own smartcard into the card reader and enters ID_i, PW_i and imprints F_i on a sensor device. Now, the card reader computes $Rep(F_i, P_F) = X_F^*$, $RPW_i^* = h\left(PW_i \| X_F^*\right)$, $A_i^* = B_i \oplus h(RPW_i^* \| ID_i)$, $D_i^* = h(A_i^* \| RPW_i^* \| ID_i)$, $[h(B_i \| X)]^* = C_i \oplus A_i^*$ and verifies if $D_i^* \stackrel{?}{=} D_i$. If this verification holds, then continues the process. Otherwise, the session is terminated.

Step 2: The U_i enters new password PW_i^{new} and computes $RPW_i^{new} = h\left(PW_i^{new} \| X_F\right)$, $B_i^{new} = B_i \oplus h(RPW_i \| ID_i) \oplus h\left(RPW_i^{new} \| ID_i\right)$, $A_i^{new} = B_i^{new} \oplus \left(RPW_i^{new} \| ID_i\right)$, $C_i^{new} = C_i \oplus A_i \oplus A_i^{new}$ and $D_i^{new} = h(A_i^{new} \| RPW_i^{new} \| ID_i)$. Then, $\{B_i,\ C_i,\ D_i\}$ are replaced with $\left\{B_i^{new},\ C_i^{new}, D_i^{new}\right\}$ respectively.

U_i
Inserts smartcard into card reader, enters ID_i, PW_i and imprints F_i
Computes $Rep(F_i, P_F) = X_F^*$, $RPW_i^* = h(PW_i \| X_F^*)$,
$A_i^* = B_i \oplus h(RPW_i^* \| ID_i)$,
$D_i^* = h(A_i^* \| RPW_i^* \| ID_i)$,
$[h(B_i \| X)]^* = C_i \oplus A_i^*$
If $D_i^* \stackrel{?}{=} D_i$ holds, U_i enters new password PW_i^{new}
Computes $RPW_i^{new} = h(PW_i^{new} \| X_F)$,
$B_i^{new} = B_i \oplus h(RPW_i \| ID_i) \oplus h(RPW_i^{new} \| ID_i)$,
$A_i^{new} = B_i^{new} \oplus h(RPW_i^{new} \| ID_i)$,
$C_i^{new} = C_i \oplus A_i \oplus A_i^{new}$,
$D_i^{new} = h(A_i^{new} \| RPW_i^{new} \| ID_i)$
Replace $\{B_i, C_i, D_i\}$ with $\{B_i^{new}, C_i^{new}, D_i^{new}\}$

Figure 4. Password update phase of Ali et al.'s scheme.

2.2.6. Dynamic Node Addition Phase

This phase was used to add, replace, or drop a sensor node in the field. Let S_n becomes a sensor node that will be added into the field. SA chooses ID_n of S_n, calculates $RI_n = h(ID_n \| X)$ and keeps $\{RI_n, ID_n\}$ into sensor nodes memory. At last, SA deploys S_n to the field.

2.3. Weaknesses of Ali et al.'s Scheme

This section discusses the weaknesses of Ali et al.'s scheme in detail. Ali et al.'s scheme weaknesses are divided into three sections which are violation of traceability, insider attack, and denial of service as a service. For the insider attack, it is divided into four other sub-sections which are violation of user anonymity, sensor node impersonation attack, perfect forward secrecy, and violation of session key security. The details are described as follows.

2.3.1. Violation of User Traceability

User traceability means the ability to distinguish if any transactions belong to or came from a certain user. Ali et al.'s scheme was trying to protect users' real identity by using pseudonym identity DID_i, where $DID_i = ID_i \oplus h(B_i \| X)$. However, the value of DID_i is constant in every transaction. By using or checking the DID_i, the adversary is able to distinguish existing transactions easily whether they are generated from the same user or not. Since the transaction is easily be distinguished, therefore, the scheme of Ali et al. fails to provide user un-traceability.

2.3.2. Insider Attack

Insider attack happens when a malicious legal participant successfully captures key values of others, such as a shared key, and then uses that key to launch some security violations or attacks. In Ali et al.'s scheme, each sensor node SN_j has a shared key RI_j with base station BS, where $RI_j = h\left(ID_{SN_j} \| X\right)$ and it should be known only by BS and SN_j. But, other legal participants such as GWN_j and U_i can also obtain RI_j automatically from a legal transaction during the authentication and session key agreement phase. GWN_j and U_i obtain RI_j when they decrypt $D_A(M_1)$ and $D_{h(R_U \| ID_i)}(M_{10})$, respectively, where $M_1 = E_{Ai}\left(R_U \| ID_{SN_j} \| ID_{GWN_j} \| T_1\right)$ and $M_{10} = E_{h(RU \| IDi)}\left(RI_j \| T_9 \| R_{GWN_j} \| R_{SN_j} \| M_2\right)$. After these legal GWN_j and U_i obtain RI_j, they can

use RI_j to release some security violations or attacks such as sensor node capture attacks, impersonation attacks, and perfect forward secrecy attacks.

Violation of User Anonymity

User anonymity is important since it protects the real identity ID_i of a user U_i and ensures his/her privacy. In Ali et al.'s scheme, once a legal participant obtained a shared key RI_j, he/she can catch others' existing transactions $\{M_2,\ M_6,\ M_7\}$ from the public channel, use RI_j to decrypt M_6, and then get the ID_i of U_i, where $D_{RI_j}(M_6) = \left(R_{BS} \parallel T_5 \parallel R_U \parallel R_{GWN_j} \parallel ID_i\right)$. Therefore, the proposed scheme fails to provide user anonymity.

Sensor Node Impersonation Attack

Sensor node impersonation attack occurred when a malicious insider successfully acts as a legitimate sensor node. When a legitimate sensor node is breached or captured by an adversary, it might result in severe security breaches [30], such as eavesdropping, node malfunctioning, denial of service, node subversion, node outage, message corruption, false nodes, and node replication. In Ali et al.'s scheme, when a malicious user U_{adv} or gateway node GWN_{adv} tries to impersonate a sensor node SN_j by using the shared key RI_j, first they catch the request message $\{M_2,\ M_6,\ M_7\}$ and then decrypt M_6, such as shown in previous subsection Violation of user anonymity. After that, they generate a timestamp T_7, a random nonce R_{SN_j} and then compute M_8, SK and M_9, where $M_8 = R_{SN_j} \oplus h\left(RI_j \parallel R_{GWN_j}\right)$, $SK = h\left(R_{GWN_j} \parallel R_U \parallel R_{SN_j} \parallel RI_j \parallel M_2\right)$, $M_9 = h(SK \parallel ID_i)$. Then, he/she sends $\{M_8,\ M_9,\ T_7\}$ to GWN_j and GWN_j will send it to the user. Since both the key and procedure are true during computation, both U_i and GWN_j will not find any suspicious activity and will trust that malicious SN_j. Therefore, Ali et al.'s scheme cannot withstand sensor node impersonation attack.

Perfect Forward Secrecy Attack

A perfect forward secrecy attack occurs when an adversary can successfully obtain previous session keys by using a compromised key. In Ali et al.'s scheme, a malicious user U_{adv} or gateway node GWN_{adv} tries to generate previous session key SK by using known shared key RI_j. First, he/she obtains R_U and ID_i through M_6, such as shown in in previous subsection Violation of user anonymity. Then, using R_U and ID_i, he/she decrypts M_{10}, where $\left(RI_j \| T_9 \| R_{GWN_j} \| R_{SN_j} \| M_2\right) = D_{h(R_U \| ID_i)}(M_{10})$. Then, he/she calculates SK^* and M_9^*, respectively, where $SK^* = h\left(R_{GWN_j} \| R_U \parallel R_{SN_j} \| RI_j \parallel M_2\right)$ and $M_9^* = h(SK^* \parallel ID_i)$. To verify if SK^* is true, the attacker compares M_9^* with previous publicly known M_9 in $\{M_8,\ M_9\ ,\ T_7\}$. If equals, the adversary has confirmation that SK^* is true. Therefore, Ali et al.'s scheme cannot withstand perfect forward secrecy attack.

Violation of Session Key Security

A session key is important to ensure the communication between legal participants in each session is secure. Violation of session key security happens when a non-legal participant can successfully generate a session key with other legal participants. In Ali et al.'s scheme, such as described in previous subsection Sensor node impersonation attack, a malicious insider successfully acts as a legitimate sensor node and is authenticated by a legal user U_i. When authentication and key agreement succeed, they will generate a session key and use that session key to communicate with each other. Therefore, Ali et al.'s scheme fails to provide session key security.

2.3.3. Denial of Services as a Service in Authentication and Key Agreement Phase

Denial of Service as a Service (DoSaaS) happened when a service cannot continue to the next step simply because of the incompatibility procedures of the exchange scheme or because of false data

calculation procedures in the scheme. In Ali et al.'s scheme, the denial of services as a service happens after user U_i successfully updates his/her password.

In the update password phase, when U_i wants to update his/her password, he/she first inserts his/her smart card and password, then inserts his/her new password PW_i^{new}. Then, RPW_i^{new}, B_i^{new}, A_i^{new}, C_i^{new} and D_i^{new} are computed, where $RPW_i^{new} = h(PW_i^{new} \| X_F)$, $B_i^{new} = B_i \oplus h(RPW_i \| ID_i) \oplus h(RPW_i^{new} \| ID_i)$, $A_i^{new} = B_i^{new} \oplus h(RPW_i^{new} \parallel ID_i)$, $C_i^{new} = C_i \oplus A_i \oplus A_i^{new}$ and $D_i^{new} = h(A_i^{new} \| RPW_i^{new} \| ID_i)$. At last, previous $\{B_i, C_i, D_i\}$ that were saved in the smart card are replaced with $\{B_i^{new}, C_i^{new}, D_i^{new}\}$, respectively. When U_i wants to login after successfully updating his/her password, the login process fails due to denial of service. Details are explained below.

As shown in the login phase in Section 2.3, U_i computes $[h(B_i \parallel X)]^* = C_i \oplus A_i^*$, where C_i is the new C_i^{new} and A_i^* is the new A_i^{new}, which means $[h(B_i \parallel X)]^* = C_i^{new} \oplus A_i^{new}$. Unfortunately, in the update password phase, $C_i^{new} = C_i^{old} \oplus A_i^{old} \oplus A_i^{new}$, which means $C_i^{new} \oplus A_i^{new} = \left(C_i^{old} \oplus A_i^{old} \oplus A_i^{new}\right) \oplus A_i^{new} = C_i^{old} \oplus A_i^{old} = [h(B_i \parallel X)]^{old}$. Using this $[h(B_i \parallel X)]^{old}$, U_i computes DID_i, M_1 and M_2, where $DID_i = ID_i \oplus h(B_i \parallel X)^{old}$, $M_1 = E_{Ai}\left(R_U \parallel ID_{SN_j} \| ID_{GWN_j} \parallel T_1\right)$, $M_2 = h\left(R_U \parallel ID_i \parallel T_1 \| h(B_i \| X)^{old}\right)$, respectively, and send $\{B_i^{new}, DID_i, M_1, M_2\}$ to BS.

When BS get the request message $\{B_i^{new}, DID_i, M_1, M_2\}$ from U_i, BS will calculate ID_i and M_2, where $ID_i^* = DID_i \oplus h(B_i^{new} \| X)$ and $M_2^* = h(R_U \| ID_i^* \parallel T_1 \| h(B_i^{new} X))$. Then compare whether M_2 equals M_2^*. Since M_2 from U_i was calculated by using $[h(B_i \parallel X)]^{old}$ and M_2^* from BS was calculated by using $h(B_i^{new} X)$, M_2 and M_2^* will never be equal. When they do not equal, BS will reject the request message from U_i. Therefore, Ali et al.'s scheme suffers from DoSaaS.

3. Proposed Authentication and Key-Agreement Scheme Using WSNs for Agriculture Monitoring

The proposed scheme proposed some significant improvements compared to Ali et al.'s scheme. For example, to overcome violation of traceability in Ali et al.'s scheme, instead of using static A_i and DID_i, the proposed scheme uses dynamic A_i and DID_i. The proposed scheme also eliminates sensor node impersonation attack, perfect forward secrecy and violation of user anonymity by keeping the shared secret key RI_j to be known only by BS and SN_j, while in Ali et al.'s scheme, the RI_j is known by all participants. To overcome Denial of Service as a Service in Ali et al.'s scheme, the proposed scheme proposes a different structure for password update phase, where in order to complete the password update process, the user U_i needs to send the new updated parameters to the base station BS to be processed. Moreover, to significantly improved efficiency, the proposed scheme only uses hash function in its computation, while Ali et al. used symmetric encryption-decryption for their scheme.

The proposed scheme consists of six phases, which are system setup phase; user/agriculture professional registration phase; login phase, authentication and session key agreement phase; password update or change phase; and dynamic node addition phase. Since the system setup phase and the dynamic node addition phase of the proposed scheme are similar with Ali et al.'s scheme, they are not presented here. Therefore, only user/agriculture professional registration phase; login phase; authentication and session key agreement phase; and password update or change phase are described in detail as follows.

3.1. User/Agriculture Professional Registration Phase

In this phase, the user/agriculture professional U_i registers to the base station BS. Each user U_i has a SC which contains a pre-configured identity ID_i^{pre} and a random number r_0. The pre-configured data is also stored in BS's storage. The SC is transferred by using physical delivery. As shown in Figure 5, the following steps are executed to complete the registration phase.

Step 1: The U_i selects his/her own identity ID_i, password PW_i, and imprints biometric F_i on the sensor device and then computes $Gen(F_i) = (X_F,\ P_F)$, $RPW_i = h(PW_i \| X_F)$, where $Gen(.)$ is a generate function of fuzzy extractor and $(X_F,\ P_F)$ are secret and public key respectively. Now, U_i computes $REG_i = r_0 \oplus (ID_i||RPW_i||A_i)$ and sends $\{\ ID_i{}^{pre}, REG_i\}$ to BS.

Step 2: When the registration request is received from U_i, if BS successfully verifies that $(ID_i{}^{pre}, r_0)$ is in BS's storage and has not been registered, then BS computes $(ID_i||RPW_i||A_i) = REG_i \oplus r_0$, $B_i = h(A_i \| X) \oplus h(ID_i \| RPW_i)$ and $D_i = h(A_i \parallel RPW_i \parallel ID_i)$. Afterwards, BS computes $RSP_i = h((ID_i||r_0) \oplus (B_i||D_i)$ and sends $\{RSP_i\}$ to U_i.

Step 3: After receiving the response from BS, U_i computes $(B_i||D_i) = RSP_i \oplus h((ID_i||r_0)$, and embeds $A_i,\ B_i,\ D_i,\ h(.),\ P_F$ and $Gen(.)$ in the memory of SC.

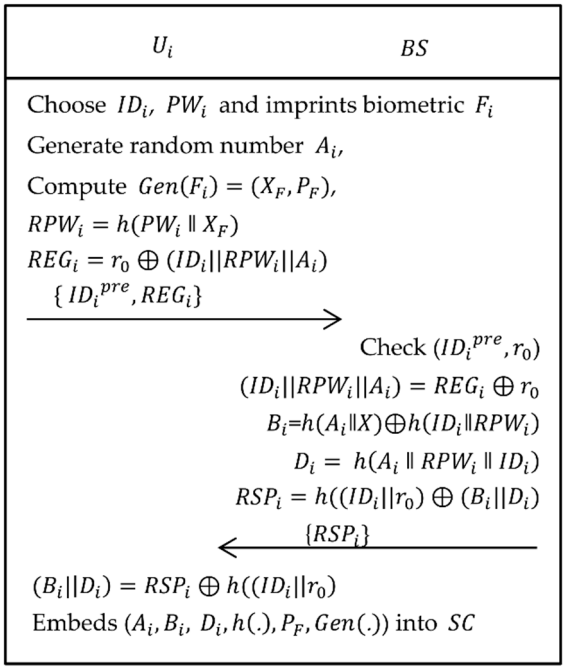

Figure 5. User registration phase of the proposed scheme.

3.2. Login Phase

When a user/agriculture professional U_i wants to know the environmental information such as temperature, light, humidity, soil etc., he/she has to login to access these information. As shown in Figure 6, the following steps are executed to accomplish the login phase.

Step 1: The U_i inserts his/her own smartcard into card reader, inputs ID_i, PW_i and imprints his/her biometric F_i on sensor device. Now, the card reader computes $Rep(F_i, P_F) = X_F^*$, $RPW_i^* = h(PW_i \parallel X_F^*)$, $[h(A_i \| X)]^* = B_i \oplus h(ID_i \| RPW_i^*)$, $D_i^* = h(A_i^* \parallel RPW_i^* \parallel ID_i)$ and verifies if D_i^* equals D_i. If the verification holds, the system continues to process the request. Otherwise, the session is terminated.

Step 2: Now, U_i generates a random nonce R_U, computes $DID_i = (ID_i \| R_U) \oplus h(h(A_i \| X) \| T_1)$ and $M_1 = h(R_U \| ID_i \| T_1 \| h(A_i \| X))$, then send $\left\{A_i,\ DID_i,\ T_1,\ M_1,\ ID_{SN_j},\ ID_{GWN_j}\right\}$ to BS via public channel.

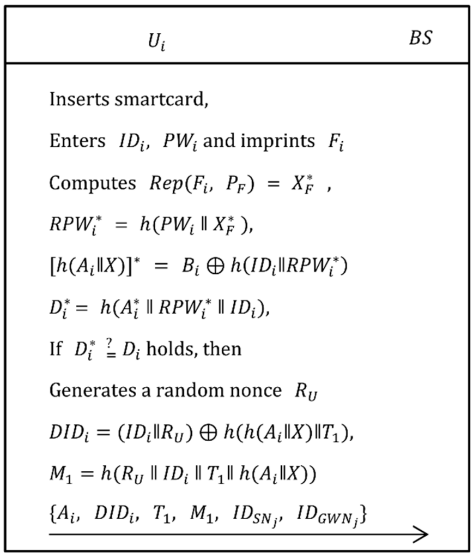

Figure 6. Login phase of the proposed scheme.

3.3. Authentication and Session Key Agreement Phase

As shown in Figure 7, after the U_i is successfully authenticated in the login phase, the authentication and session-key agreement phase is executed as the following steps.

Step 1: Upon obtaining the message $\left\{A_i, DID_i, T_1, M_1, ID_{SN_j}, ID_{GWN_j}\right\}$ from U_i, the BS checks if $T_2 - T_1 \leq \Delta T$ holds. If this does not true then session expires. Otherwise, BS calculates $(ID_i \| R_U) = DID_i \oplus h(h(A_i\|X) \| T_1)$ and computes $M_1^* = h(R_U \| ID_i \| T_1 \| h(A_i\|X))$, then verifies if M_1^* equals M_1 or not. If this holds, BS goes to next step. Otherwise, the session is rejected.

Step 2: Now, BS generates a random nonce R_{BS} and computes a new A_i^{new}, where A_i^{new}=$h(R_U\|h(A_i\|X))$. Then, BS computes M_2= $h\left(A_i^{new}\|X\right) \oplus h(h(A_i\|X)\|R_{BS})$, $M_3 = (R_U \| R_{BS} \| ID_i) \oplus h\left(X_{BS-GWN_j}\| T_3\right)$, $M_4 = h(M_1\|M_2\|ID_i\|T_3\|R_{BS}\|R_U)$ and $M_5 = R_{BS} \oplus h\left(h\left(ID_{SN_j}\|X\right) \| T_3\right)$ and sends $\{M_1, M_2, M_3, M_4, M_5, T_3\}$ to GWN_j via public channel.

Step 3: After getting the request message from BS, the GWN_j checks if $T_4 - T_3 \leq \Delta T$ holds. If this does not true then session expires. Otherwise, GWN_j calculates $(R_U\|R_{BS}\|ID_i) = M_3 \oplus h\left(X_{BS-GWN_j}\| T_3\right)$ and $M_4^* = h(M_1\|M_2\|ID_i\|T_3\|R_{BS}\|R_U)$, then checks if $M_4^* \stackrel{?}{=} M_4$ holds. If the condition is true then it proceeds further. Otherwise, the session is terminated.

Step 4: Now, the GWN_j generates a random nonce R_{GWN_j}, calculates M_6= $\left(R_U\|R_{GWN_j}\|ID_i\right) \oplus h(R_{BS}\|M_5\|T_5)$ and $M_7 = h\left(M_1\|M_2\|R_{BS}\|R_{GWN_j}\|ID_i\|R_U\right)$, then sends $\{M_1, M_2, M_5, M_6, M_7, T_3, T_5\}$ to SN_j.

Step 5: Upon obtaining the message from GWN_j, the SN_j checks if $T_6 - T_5 \leq \Delta T$ holds. If this does not true then session expires. Otherwise, SN_j calculates $R_{BS}^* = M_5 \oplus h\left(RI_j \| T_3\right)$, $\left(R_U \| R_{GWN_j}\| ID_i\right) = M_6 \oplus h\left(R_{BS}^* \| M_5 \| T_5\right)$ and $M_7^* = h\left(M_1 \| M_2 \| R_{BS}^* \| R_{GWN_j}\| ID_i \| R_U\right)$. Then, SN_j verifies if $M_7^* \stackrel{?}{=} M_7$ holds. If the condition is true then it proceeds further. Otherwise, the session is terminated.

Step 6: Now, the SN_j generates a random nonce R_{SN_j}, computes $M_8 = R_{SN_j} \oplus h\left(R_{GWN_j}\| R_{BS}^* \| T_7\right)$, $SK = h\left(R_{GWN_j}\| R_U \| R_{SN_j}\| R_{BS}^* \| ID_i \| M_1\right)$ and $M_9 = h\left(SK \| R_{BS}^* \| R_U \| M_2 \| T_7\right)$. Then, SN_j sends $\{M_1, M_2, M_8, M_9, T_7\}$ to GWN_j.

Step 7: Upon receiving the message from SN_j, GWN_j firstly verifies if $T_8 - T_7 \leq \Delta T$ holds. If this is not true then the session expires. Otherwise, GWN_j calculates $R_{SN_j}^* = M_8 \oplus h\left(R_{GWN_j}\| R_{BS} \| T_7\right)$, $SK^* = h\left(R_{GWN_j} \| R_U \| R_{SN_j}^*\|R_{BS} \| ID_i \| M_1\right)$ and $M_9^* = h(SK^* \| R_{BS} \| R_U \| M_2 \| T_7)$, then checks if $M_9^* \stackrel{?}{=} M_9$ holds. If the condition is true then further is proceeded. Otherwise, the session is terminated.

Step 8: The GWN_j computes $M_{10} = h(R_U \parallel ID_i) \oplus \left(R_{GWN_j} \| R^*_{SN_j} \| R_{BS}\right)$ and $M_{11} = h\left(R_U \parallel R^*_{SN_j} \| M_2 \parallel SK \parallel T_9\right)$. Then, GWN_j sends $\{ M_2, M_{10}, M_{11}, T_9 \}$ to U_i.

Step 9: Upon receiving the message from GWN_j, U_i firstly verifies if $T_8 - T_7 \leq \Delta T$ holds. If this does not true then session expires. Otherwise, U_i computes $\left(R_{GWN_j} \| R^*_{SN_j} \| R_{BS}\right) = h(R_U \| ID_i) \oplus M_{10}$, $SK^* = h\left(R_{GWN_j} \| R_U \| R^*_{SN_j} \| R_{BS} \| ID_i \| M_1\right)$, $h\left(A_i^{new} \parallel X\right) = M_2 \oplus h(h(A_i \| X) \parallel R_{BS})$, $M^*_{11} = h\left(R_U \parallel R^*_{SN_j} \| M_2 \parallel SK^* \| T_9\right)$ and verifies if $M^*_{11} \stackrel{?}{=} M_{11}$ holds. If the condition is true then mutual authentication and session key agreement holds. Otherwise, the session is terminated.

Step 10: The U_i computes $B_i^{new} = h\left(A_i^{new} \parallel X\right) \oplus h(ID_i \| RPW_i)$ and $D_i^{new} = h\left(A_i^{new} \parallel RPW_i \parallel ID_i\right)$. Then, U_i replaces A_i, B_i, D_i with $A_i^{new}, B_i^{new}, D_i^{new}$, respectively.

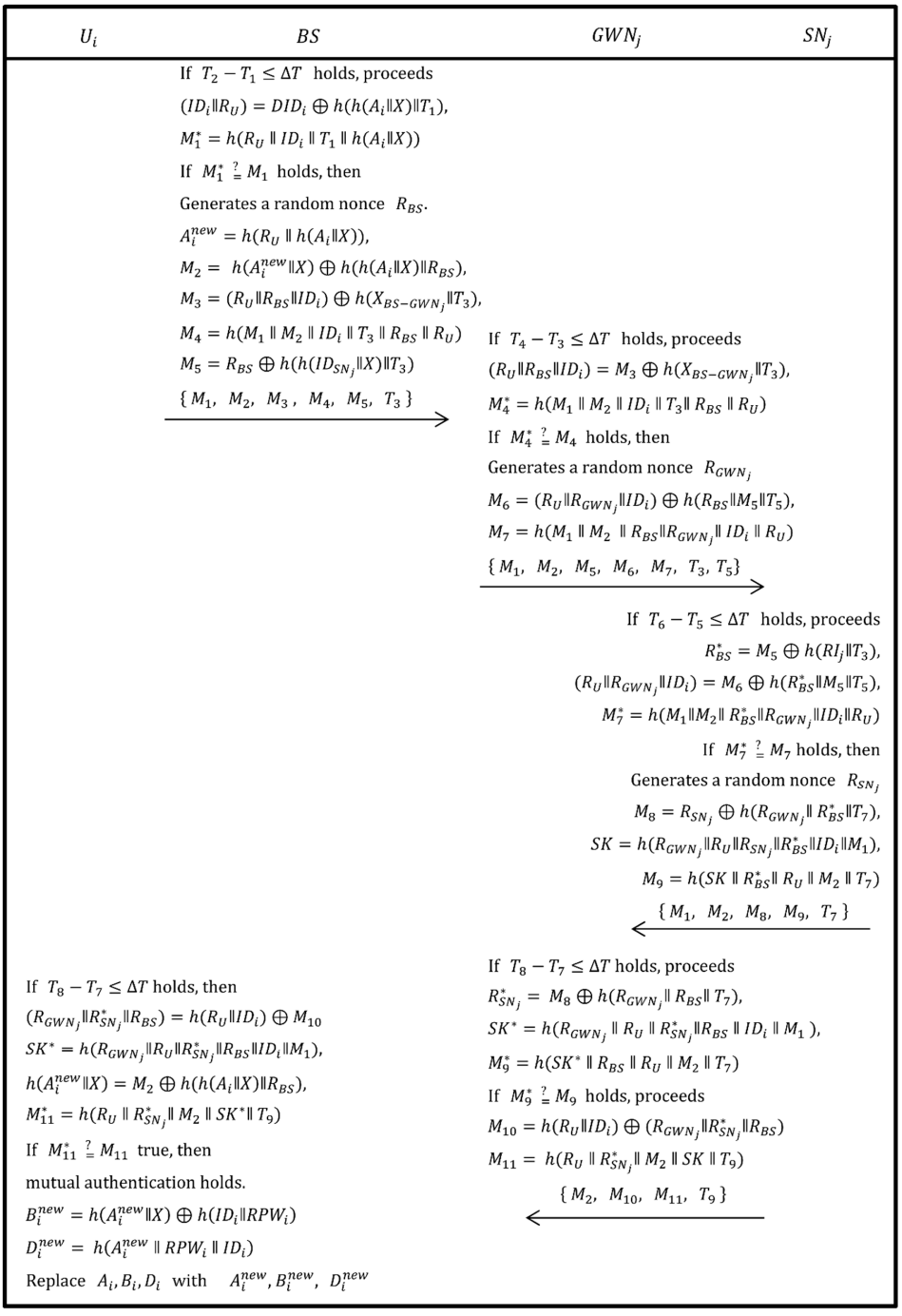

Figure 7. Authentication and key agreement phase of the proposed scheme.

3.4. Password Updates or Change Phase

As shown in Figure 8, the following steps were executed to update or change user's password.

Step 1: The user U_i inserts his/her own smartcard into card reader and enters ID_i, PW_i and imprints F_i on sensor device. The card reader computes $Rep(F_i, P_F) = X_F^*$, $RPW_i^* = h(PW_i \parallel X_F^*)$, $A_i^* = h(RPW_i^* \parallel ID_i \parallel X_F^*)$, $[h(A_i \parallel X)]^* = B_i \oplus h(ID_i \parallel RPW_i^*)$ and $D_i^* = h(A_i \| RPW_i^* \parallel ID_i)$. Then, U_i verifies if $D_i^* \stackrel{?}{=} D_i$ holds. If condition is true then further is proceeded. Otherwise, the session is terminated.

Step 2: The U_i enters new password PW_i^{new} and computes $RPW_i^{new} = h(PW_i^{new} \parallel X_F^*)$, $B_i^{new} = h(A_i \| X) \oplus h(ID_i \| RPW_i^{new})$ and $D_i^{new} = h(A_i \parallel RPW_i^{new} \parallel ID_i)$. Then, $\{B_i, D_i\}$ are replaced with $\{B_i^{new}, D_i^{new}\}$, respectively.

U_i
Inserts smartcard into card reader, enters ID_i, PW_i and imprints F_i
Computes $Rep(F_i, P_F) = X_F^*$, $RPW_i^* = h(PW_i \| X_F^*)$,
$A_i = h(RPW_i^* \| ID_i \| X_F^*)$,
$[h(A_i \| X)]^* = B_i \oplus h(ID_i \| RPW_i^*)$,
$D_i^* = h(A_i \| RPW_i^* \| ID_i)$,
If $D_i^* \stackrel{?}{=} D_i$ holds, then U_i enters new password PW_i^{new}.
Computes $RPW_i^{new} = h(PW_i^{new} \| X_F^*)$
$B_i^{new} = h(A_i \| X) \oplus h(ID_i \| RPW_i^{new})$
$D_i^{new} = h(A_i \| RPW_i^{new} \| ID_i)$
Replaces B_i, D_i with B_i^{new}, D_i^{new}
$SC = \{A_i, B_i^{new}, D_i^{new}, h(.), P_F, Gen(.)\}$

Figure 8. Password update phase of the proposed scheme.

4. Security Analysis

4.1. Authentication Proof of the Proposed Scheme Using BAN Logic

This section validates session key agreement and mutual authentication of the proposed scheme using BAN (Burrows-Abadi-Needham) logic [31]. The BAN includes a set of rules to verify the message source, freshness, and trustworthiness of the scheme. Table 2 lists the notations and their respective abbreviations related to the BAN logic.

Table 2. BAN (Burrows-Abadi-Needham) logic notations and respective abbreviations.

Notation	Abbreviation
$P \mid\equiv X$	The entity P believes the statement X
$P \Longrightarrow X$	P has jurisdiction on the statement X
$P \mid\sim X$	P once said X
$P \lhd X$	P sees X
$\{X\}_K$	Formula X is encrypted under the key K
$P \overset{K}{\leftrightarrow} Q$	P and Q communicate via shared key K
$P \rightarrow Q : m$	P sends the message m and Q receives it
$\#X$	The message $\#X$ is freshly generated

4.1.1. Basic Rules of BAN Logic

Some rules or logical postulates used in the BAN logic are given as follows:

- **Rule 1. Message-meaning rule:** $\frac{P|\equiv P\overset{K}{\leftrightarrow}Q, P \ \{X\}_K}{P|\equiv Q|\sim X}$ If the entity P believes that the secret K is shared with Q and sees message X is encrypted using K, then P believes that Q once said X.
- **Rule 2. Jurisdiction rule:** $\frac{P|\equiv Q \Longrightarrow X, P|\equiv Q|\equiv X}{P|\equiv X}$ If the entity P believes that Q has jurisdiction over X and Q believes X, then P believes that X is true.
- **Rule 3. Nonce-verification rule:** $\frac{P|\equiv \#(X), \ P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$ If the entity P believes that X is fresh and the entity Q once said X, then P believes that Q believes X.
- **Rule 4. Session key rule:** $\frac{P|\equiv \#(X), P|\equiv Q|\equiv X}{P|\equiv P\overset{K}{\leftrightarrow}Q}$ If the entity P believes that X is fresh and Q believes X, then P believes the secret K that is shared between both entities P and Q.
- **Rule 5. Freshness-conjuncatenation rule:** $\frac{P|\equiv \#(X)}{P|\equiv \#(X, Y)}$ If the entity P believes that X is fresh, then P believes the freshness of (X, Y).

4.1.2. Goals

The proposed scheme needs to satisfy the following goals to ensure its security under BAN logic, using the above assumptions and postulates.

Goal 1: $BS \mid\equiv U_i \overset{SK}{\leftrightarrow} BS$

Goal 2: $BS \mid\equiv U_i| \equiv U_i \overset{SK}{\leftrightarrow} BS$

Goal 3: $GWN_j \mid\equiv BS \overset{SK}{\leftrightarrow} GWN_j$

Goal 4: $GWN_j \mid\equiv BS \mid\equiv BS \overset{SK}{\leftrightarrow} GWN_j$

Goal 5: $SN_j \mid\equiv GWN_j \overset{SK}{\leftrightarrow} SN_j$

Goal 6: $SN_j \mid\equiv GWN_j \mid\equiv GWN_j \overset{SK}{\leftrightarrow} SN_j$

Goal 7: $GWN_j \mid\equiv SN_j \overset{SK}{\leftrightarrow} GWN_j$

Goal 8: $GWN_j \mid\equiv SN_j \mid\equiv SN_j \overset{SK}{\leftrightarrow} GWN_j$

Goal 9: $U_i \mid\equiv GWN_j \overset{SK}{\leftrightarrow} U_i$

Goal 10: $U_i \mid\equiv GWN_j \mid\equiv GWN_j \overset{SK}{\leftrightarrow} U_i$

4.1.3. Idealized Form

Initially, the message of login, authentication, and key agreement scheme in the proposed scheme can be transformed into idealized form in the following manner.

Message 1. $(U_i \ BS) \ : \mathrm{A_i}, DID_i, T_1, M_1, ID_{SN_j}, ID_{GWN_j}: \langle R_U \rangle_{h(A_i \| X)}$

Message 2. $\left(BS \ GWN_j\right) \ : M_1, M_2, M_3, M_4, M_5, T_3: \langle R_U, \ R_{BS} \rangle_{X_{BS-GWN_j}}$

Message 3. $(GWN_j \ SN_j) \ : M_1, M_2, M_5, M_6, M_7, T_3, T_5: \left\langle R_U, \ R_{GWN_j} \right\rangle_{R_{BS}}$

Message 4. $(SN_j \ GWN_j) \ : M_1, M_2, M_8, M_9, T_7: \left\langle R_{SN_j} \right\rangle_{R_{GWN_j} \| R_{BS}}$

Message 5. $(GWN_j \ U_i) \ : M_2, M_{10}, M_{11}, T_9: \left\langle R_{GWN_j}, \ R_{SN_j}, R_{BS} \right\rangle_{h(R_u \| ID_i)}$

4.1.4. Assumptions

The following initial assumptions have been established to prove the security of the proposed scheme using BAN logic.

$\boldsymbol{A_1}$**:** $U_i \mid\equiv \#(R_U, R_{BS}, R_{GWN_j}, R_{SN_j})$

$\boldsymbol{A_2}$**:** $BS \mid\equiv \#(R_U, R_{BS})$

$\boldsymbol{A_3}$**:** $GWN_j \mid\equiv \#(R_U, R_{BS}, R_{GWN_j}, R_{SN_j})$

$\boldsymbol{A_4}$: $SN_j \mid\equiv \#(R_U, R_{BS}, R_{GWN_j}, R_{SN_j})$

$\boldsymbol{A_5}$: $BS \mid\equiv BS \overset{h(A_i \| X)}{\leftrightarrow} U_i$

$\boldsymbol{A_6}$: $BS \mid\equiv U_i \Longrightarrow R_U$

$\boldsymbol{A_7}$: $GWN_j \mid\equiv GWN_j \overset{X_{BS-GWN_j}}{\leftrightarrow} BS$

$\boldsymbol{A_8}$: $GWN_j \mid\equiv BS \Longrightarrow R_{BS}$

$\boldsymbol{A_9}$: $SN_j \mid\equiv SN_j \overset{R_{BS}}{\leftrightarrow} GWN_j$

$\boldsymbol{A_{10}}$: $SN_j \mid\equiv GWN_j \Longrightarrow R_{GWN_j}$

$\boldsymbol{A_{11}}$: $GWN_j \mid\equiv GWN_j \overset{R_{GWN_j} \| R_{BS}}{\leftrightarrow} SN_j$

$\boldsymbol{A_{12}}$: $GWN_j \mid\equiv SN_j \Longrightarrow R_{SN_j}$

$\boldsymbol{A_{13}}$: $U_i \mid\equiv U_i \overset{h(R_U \| ID_i)}{\leftrightarrow} GWN_j$

$\boldsymbol{A_{14}}$: $U_i \mid\equiv GWN_j \Longrightarrow R_{GWN_j}, R_{SN_j}, R_{BS}$

4.1.5. Verification

Verification shows the correctness of the proposed scheme confirmed by analyzing the idealized form using the above assumptions and the rules of the BAN logic.

By using **Message 1**:

$\boldsymbol{V_1}$: $BS \lhd \{\mathrm{A_i}, DID_i, T_1, M_1, ID_{SN_j}, ID_{GWN_j}: \langle R_U \rangle_{h(A_i \| X)}\}$

From $\boldsymbol{A_5}$, $\boldsymbol{V_1}$ and Rule 1:

$\boldsymbol{V_2}$: $BS \mid\equiv U_i \mid\sim R_U$

From $\boldsymbol{A_2}$, $\boldsymbol{V_2}$ and **Rule 3**:

$\boldsymbol{V_3}$: $BS \mid\equiv U_i \mid\equiv R_U$

Then, from $\boldsymbol{A_6}$, $\boldsymbol{V_3}$ and **Rule 2**:

$\boldsymbol{V_4}$: $BS \mid\equiv R_U$

According to $\boldsymbol{A_2}$, $\boldsymbol{V_3}$ and **Rule 4**:

$\boldsymbol{V_5}$: $BS \mid\equiv BS \overset{sk}{\leftrightarrow} U_i$ $\boldsymbol{Goal_1}$

Further, using $\boldsymbol{A_2}$, $\boldsymbol{V_5}$ and **Rule 3**:

$\boldsymbol{V_6}$: $BS \mid\equiv U_i \mid\equiv BS \overset{sk}{\leftrightarrow} U_i$ $\boldsymbol{Goal_2}$

By using **Message 2**:

$\boldsymbol{V_6}$: $GWN_j \ \{M_1, M_2, M_3, M_4, M_5, T_3: \langle R_U, R_{BS} \rangle_{X_{BS-GWN_j}}\}$

From $\boldsymbol{A_7}$, $\boldsymbol{V_7}$ and **Rule 1**:

$\boldsymbol{V_8}$: $GWN_j \mid\equiv BS \mid\sim R_{BS}$

From $\boldsymbol{A_3}$, $\boldsymbol{V_8}$ and **Rule 3**:

$\boldsymbol{V_9}$: $GWN_j \mid\equiv BS \mid\equiv R_{BS}$

Then, from $\boldsymbol{A_8}$, $\boldsymbol{V_9}$ and **Rule 2**:

$\boldsymbol{V_{10}}$: $GWN_j \mid\equiv R_{BS}$

According to $\boldsymbol{A_3}$, $\boldsymbol{V_9}$ and **Rule 4**:

$\boldsymbol{V_{11}}$: $GWN_j \mid\equiv GWN_j \overset{sk}{\leftrightarrow} BS$ $\boldsymbol{Goal_3}$

Further, using $\boldsymbol{A_3}$, $\boldsymbol{V_{11}}$ and **Rule 3**:

$\boldsymbol{V_{12}}$: $GWN_j \mid\equiv BS \mid\equiv GWN_j \overset{sk}{\leftrightarrow} BS$ $\boldsymbol{Goal_4}$

By using **Message 3**:

$\boldsymbol{V_{13}}$: $SN_j \lhd \{M_1, M_2, M_5, M_6, M_7, T_3, T_5: \left\langle R_U, R_{GWN_j} \right\rangle_{R_{BS}}\}$

From $\boldsymbol{A_9}$, $\boldsymbol{V_{13}}$ and **Rule 1**:

$\boldsymbol{V_{14}}$: $SN_j \mid\equiv GWN_j \mid\sim R_{GWN_j}$

From $\boldsymbol{A_4}$, $\boldsymbol{V_{14}}$ and **Rule 3**:

$\boldsymbol{V_{15}}$: $SN_j \mid\equiv GWN_j \mid\equiv R_{GWN_j}$

Then, from $\boldsymbol{A_{10}}$, $\boldsymbol{V_{15}}$ and **Rule 2**:

$\boldsymbol{V_{16}}$: $SN_j \mid\equiv R_{GWN_j}$

According to $\boldsymbol{A_4}$, $\boldsymbol{V_{15}}$ and **Rule 4**:

$\boldsymbol{V_{17}}$: $SN_j \mid\equiv SN_j \overset{sk}{\leftrightarrow} GWN_j$ $\boldsymbol{Goal_5}$

Further, using $\boldsymbol{A_4}$, $\boldsymbol{V_{17}}$ and **Rule 3**:

$\boldsymbol{V_{18}}$: $SN_j \mid\equiv GWN_j \mid\equiv SN_j \overset{sk}{\leftrightarrow} GWN_j$ $\boldsymbol{Goal_6}$

By using **Message 4**:

$\boldsymbol{V_{19}}$: $GWN_j \triangleleft \{M_1, M_2, M_8, M_9, T_7: \left\langle R_{SN_j} \right\rangle_{R_{GWN_j} \| R_{BS}}\}$

From $\boldsymbol{A_{11}}$, $\boldsymbol{V_{19}}$ and **Rule 1**:

$\boldsymbol{V_{20}}$: $GWN_j \mid\equiv SN_j \mid\sim R_{SN_j}$

From $\boldsymbol{A_3}$, $\boldsymbol{V_{20}}$ and **Rule 3**:

$\boldsymbol{V_{21}}$: $GWN_j \mid\equiv SN_j \mid\equiv R_{SN_j}$

Then, from $\boldsymbol{A_{12}}$, $\boldsymbol{V_{21}}$ and **Rule 2**:

$\boldsymbol{V_{22}}$: $GWN_j \mid\equiv R_{SN_j}$

According to $\boldsymbol{A_3}$, $\boldsymbol{V_{21}}$ and **Rule 4**:

$\boldsymbol{V_{23}}$: $GWN_j \mid\equiv GWN_j \overset{sk}{\leftrightarrow} SN_j$ $\boldsymbol{Goal_7}$

Further, using $\boldsymbol{A_4}$, $\boldsymbol{V_{17}}$ and **Rule 3**:

$\boldsymbol{V_{24}}$: $GWN_j \mid\equiv SN_j \mid\equiv GWN_j \overset{sk}{\leftrightarrow} SN_j$ $\boldsymbol{Goal_8}$

By using **Message 5**:

$\boldsymbol{V_{25}}$: $U_i \triangleleft \{M_2, M_{10}, M_{11}, T_9: \left\langle R_{GWN_j}, R_{SN_j}, R_{BS} \right\rangle_{h(R_u \| ID_i)}\}$

From $\boldsymbol{A_{13}}$, $\boldsymbol{V_{25}}$ and **Rule 1**:

$\boldsymbol{V_{26}}$: $U_i \mid\equiv GWN_j \mid\sim R_{GWN_j}, R_{SN_j}, R_{BS}$

From $\boldsymbol{A_1}$ and **Rule 5**:

$\boldsymbol{V_{27}}$: $U_i \mid\equiv \#(R_{GWN_j}, R_{SN_j}, R_{BS})$

Then, from $\boldsymbol{V_{26}}$, $\boldsymbol{V_{27}}$ and **Rule 3**:

$\boldsymbol{V_{28}}$: $U_i \mid\equiv GWN_j \mid\equiv R_{GWN_j}, R_{SN_j}, R_{BS}$

Moreover, from $\boldsymbol{A_{14}}$, $\boldsymbol{V_{28}}$ and **Rule 2**:

$\boldsymbol{V_{29}}$: $U_i \mid\equiv R_{GWN_j}, R_{SN_j}, R_{BS}$

According to $\boldsymbol{V_{27}}$, $\boldsymbol{V_{28}}$ and **Rule 4**:

$\boldsymbol{V_{30}}$: $U_i \mid\equiv U_i \overset{sk}{\leftrightarrow} GWN_j$ $\boldsymbol{Goal_9}$

Then, using $\boldsymbol{V_{27}}$, $\boldsymbol{V_{30}}$ and **Rule 3**:

$\boldsymbol{V_{31}}$: $U_i \mid\equiv GWN_j \mid\equiv U_i \overset{sk}{\leftrightarrow} GWN_j$ $\boldsymbol{Goal_{10}}$

4.2. Informal Security Analysis

This section presents informal security analysis of the proposed scheme. Table 2 summarizes security analysis comparisons between Ali et al.'s scheme [23] and the proposed scheme.

4.2.1. User Anonymity

When base station BS gets a request message $\left\{A_i,\ DID_i,\ T_1,\ M_1,\ ID_{SN_j},\ ID_{GWN_j}\right\}$ from user U_i, base station BS checks whether the request message comes from a legitimate user U_i by calculating $(ID_i \| R_U) = DID_i \oplus h(h(A_i \| X) \| T_1)$, where X is BS's secret key that is known only by BS. BS checks whether $h(R_U \parallel ID_i \parallel T_1 \parallel h(A_i \| X))$ equals M_1. If it holds, BS confirms that the request message is coming from a legitimate U_i.

Assume an adversary tries to get the real identity ID_i of a legitimate user U_i from an existing message $\left\{A_i,\ DID_i,\ T_1,\ M_1,\ ID_{SN_j},\ ID_{GWN_j}\right\}$ that can be obtained from public channel. In order to successfully get the real ID_i, the adversary needs to calculate $DID_i \oplus h(h(A_i \| X) \| T_1)$. However, X is only known by the legitimate BS and is also protected by the hash operation that makes X is

computationally infeasible to calculate. Without knowledge of X, the adversary cannot derive the real ID_i. Therefore, the proposed scheme provides user anonymity.

4.2.2. User Traceability

In the proposed scheme, the real identity ID_i of a user is protected by using dynamic pseudonym identity DID_i, where $DID_i = (ID_i \| R_U) \oplus h(h(A_i \| X) \| T_1)$. R_U is random and T_1 is timestamp that newly generated for each transaction, means DID_i is dynamic for every transaction. Moreover, different values of A_i, DID_i, T_1, M_1 for each transaction prevents adversaries to identify a transaction belonging to whom or related with any specific user. Therefore, the proposed scheme provides protection to user traceability.

4.2.3. Three-Factor Security

To provide protection in the login phase, the proposed scheme uses three-factor security which means only a user with the correct password, correct biometric characteristics, and correct smart card is allowed to login to the remote server [32].

Assume an adversary has any two factors of security which are password and smartcard, or smartcard and biometric, or password and smartcard. When he/she tries to login into the system, the proposed scheme will check whether $h(A_i \| RPW_i^* \| ID_i)$ equals with D_i, where $A_i = h(RPW_i^* \| ID_i \| X_F^*)$ and $RPW_i^* = h(PW_i \| X_F^*)$. Based on this checking, the system always completely checks three factor security first before allowed any request to successfully login into the system. This process means an adversary who has only two factors of security does not have a chance to enter into the system. Therefore, the proposed scheme provides three-factor security.

4.2.4. Session Key Security

In an authentication and key agreement phase, the session key SK must be made and known only by legal participants. In the proposed scheme, SK is computed by using random numbers from each legal participant that freshly generated in each session. Furthermore, SK also depends on ID_i and M_1, where $M_1 = h(R_U \| ID_i \| T_1 \| h(A_i \| X))$ and it was protected by $h(A_i \| X)$ that is only known by U_i and BS. It is also computationally infeasible to calculate the session key $SK = h\left(R_{GWN_j} \| R_U \| R_{SN_j} \| R_{BS}^* \| ID_i \| M_1\right)$ due to the characteristics of the hash operation. Therefore, the proposed scheme withstands session key computation attack.

4.2.5. Perfect Forward Secrecy Attack

The proposed scheme ensures the secrecy of previous session keys even if the master secret key of the server or shared secret key between legal participants are compromised.

In the proposed scheme, the session key is not related to the master secret key X that belongs to the base station BS. Also, the session key is not related with any shared secret key that exists between legal participants, such as the shared key between BS and gateway node X_{BS-GWN_j} or the shared key between BS and sensor node RI_j. Instead, the session key is built from each random number that is freshly generated by every legal participant from each session. Therefore, the proposed scheme provides perfect forward secrecy.

4.2.6. Sensor Node Impersonation Attack

Assume an adversary tries to impersonate a sensor node by sending a request message $\{M_1, M_2, M_8, M_9, T_7\}$ to a gateway node GWN_j. Upon receiving the request message, to verify if the request message comes from a legitimate sensor node SN_j or not, GWN_j computes $M_9^* = h(SK^* \| R_{BS} \| R_U \| M_2 \| T_7)$ and checks if M_9^* equals M_9 or not.

In the proposed scheme, in order to compute verifiable M_9, both SN_j and GWN_j need to obtain R_{BS}^*, where R_{BS} is a random nonce belongs to the base station BS. As shown in authentication and

session key agreement phase, in order to obtain R_{BS}, both SN_j and GWN_j need to use their own shared secret key with the BS, where GWN_j uses its shared secret key X_{BS-GWN_j} and SN_j uses its shared secret key RI_j. Without shared secret key, an adversary will not be able to obtain R_{BS}. Without the right R_{BS}, M_9 will never be successfully verified by GWN_j. By verifying the M_9, GWN_j will immediately detect that the request message is coming from legal SN_j or not. Therefore, the proposed scheme withstands sensor node impersonation attack.

4.2.7. Gateway Node Impersonation Attack

Gateway node impersonation attack occurs when an adversary acts as a legitimate gateway node GWN_j by sending a request message to user U_i or sensor node SN_j and that request message is successfully authenticated as a legitimate GWN_j by U_i or SN_j.

Assume an adversary tries to impersonate GWN_j by sending a request message $\{ M_1, M_2, M_5, M_6, M_7, T_3, T_5\}$ to SN_j or $\{ M_2, M_{10}, M_{11}, T_9 \}$ to U_i, where $M_6 = \Big(R_U \parallel R_{GWN_j} \parallel ID_i\Big) \oplus h(R_{BS} \parallel M_5 \parallel T_5)$, $M_7 = h\Big(M_1 \parallel M_2 \parallel R_{BS} \parallel R_{GWN_j} \parallel ID_i \parallel R_U\Big)$, $M_{10} = h(R_U \parallel ID_i) \oplus \Big(R_{GWN_j} \parallel R^*_{SN_j} \parallel R_{BS}\Big)$ and $M_{11} = h\Big(R_U \parallel R^*_{SN_j} \parallel M_2 \parallel SK \parallel T_9\Big)$. To compute M_6, M_7, M_{10} and M_{11}, the adversary needs to obtain R_U, R_{BS} and ID_i using shared key X_{BS-GWN_j}, where $(R_U \parallel R_{BS} \parallel ID_i) = M_3 \oplus h\Big(X_{BS-GWN_j} \parallel T_3\Big)$. However, without the knowledge of X_{BS-GWN_j}, it is computationally infeasible to calculate these parameters due to the characteristics of the hash operation. Without the right parameters, U_i and SN_j will immediately recognize if the request is not coming from a legitimate GWN_j. Therefore, the proposed scheme withstands a gateway node impersonation attack.

4.2.8. User/Agriculture Impersonation Attack

A user impersonation attacks occur when an adversary acts as a legitimate user and is successfully authenticated by the base station BS.

Assume an adversary tries to impersonate a legitimate user U_i by sending a request message $\Big\{A_i, DID_i, T_1, M_1, ID_{SN_j}, ID_{GWN_j}\Big\}$ to BS, where $DID_i = (ID_i \parallel R_U) \oplus h(h(A_i \| X) \parallel T_1)$, $M_1 = h(R_U \parallel ID_i \parallel T_1 \| h(A_i \| X))$, $[h(A_i \| X)]^* = B_i \oplus h\big(ID_i \parallel RPW^*_i\big)$ and $RPW^*_i = h(PW_i \parallel X^*_F)$. However, it is impossible for an adversary to calculate $[h(A_i \parallel X)]^*$ due to biometrics, unknown user identity ID_i and user password PW_i.

Upon receiving the request message from U_i, BS will immediately recognize that the request message is coming from a legitimate user or not by checking whether $h(R_U \parallel ID_i \parallel T_1 \parallel h(A_i \| X))$ equals M_1 or not. Therefore, the proposed scheme withstands user/agriculture impersonation attack.

4.2.9. Offline Password Guessing Attack

An off-line password guessing attack occurs when a smart card is lost or stolen and the adversary tries to guess the password to log into the system. Let us assume an adversary obtains information within the smart card by using channel side attacks and successfully obtains $\{A_i, B_i, D_i, h(.), P_F, Gen(.)\}$, where $B_i = h(A_i \| X) \oplus h(ID_i \| RPW_i)$, $D_i = h(A_i \parallel RPW_i \parallel ID_i)$, and $RPW_i = h(PW_i \parallel X_F)$. To guess the password through the parameter that are stored inside the smart card, the adversary needs to invert the value of B_i or D_i. However, inverting the values of B_i or D_i is computationally infeasible due to the characteristics of the hash operation. Neither ID_i. or PW_i are ever directly revealed or exposed and an adversary for sure cannot guess or change the password. Therefore, the proposed scheme withstands an offline password guessing attack.

4.2.10. Replay Attack

An off-line password guessing attack occurs when a smart card is lost or stolen and the adversary replay attack happens when an adversary tries to retransmit previous request message as a new

transaction request and it has successfully been accepted as a new legitimate request by other legal participants.

Assume an adversary tries to replay existing messages as a new transaction request. However, any message contains a timestamp T_1, T_3, T_5, T_7, or T_9. Other legitimate participants will immediately identify the replay attack when they check the freshness of T_1, T_3, T_5, T_7, and T_9. Therefore, the proposed scheme withstands replay attack.

4.2.11. Insider Attack

An insider attack happens when a malicious legal participant successfully captures key values of others, such as a shared key, and then uses that key to launch some security violations or attacks. The proposed scheme ensures that the key shared between participants is known only by the right participant and will never be leaked to other irrelevant participants.

Assume a malicious legal participant tries to obtain a shared secret key that belongs to another legal participant. In the proposed scheme, there are three shared secret key which are X_{BS-GWN_j}, RI_j and $h(A_i||X)$. All of them are generated by base station *BS* and contains *BS* secret key X. Since the secret key of *BS* is only known by *BS* and never revealed to others, and since the shared key between participants is never revealed to other irrelevant participants too, the proposed scheme is safe from shared secret key leakage. Therefore, the proposed scheme withstands insider attack.

5. Performance and Functionality Comparisons

This section analyzes and compares Ali et al.'s scheme with the proposed scheme. Security functionality comparisons and performance comparisons in login, authentication, and key agreement phase are presented as follows.

5.1. Security Functionality Comparisons

Table 3 shows comparisons between the proposed scheme and Ali et al.'s scheme in terms of functionality in security. It shows that the proposed scheme enables provision of more security functionality where there are lacks in Ali et al.'s scheme. Detailed explanations about how Ali et al.'s scheme suffers from those attacks was already described in Section 2.3.

Table 3. Functionality comparisons.

Attributes	Ali et al. Scheme	Proposed Scheme
User anonymity	N	Y
User traceability	N	Y
Three-factor security	Y	Y
Session key security	N	Y
Perfect forward secrecy attack	N	Y
Sensor node impersonation attack	N	Y
Gateway node impersonation attack	Y	Y
User/agriculture impersonation attack	Y	Y
Offline password guessing attack	Y	Y
Replay attack	Y	Y
Insider attack	N	Y
Denial of Service as a Service	N	Y

5.2. Performance Comparisons

As WSNs has limited power capacity, the computation cost for login, authentication, and key-agreement scheme must be made as minimal as possible. Table 4 shows the comparison of login, authentication, and key agreement phases between the proposed scheme and Ali et al.'s scheme in terms of performance. Table 5 shows hardware/software specifications and used algorithms in our simulation environment. The proposed scheme involves a user U_i, a base station *BS*, a gateway node

GWN_j, and a sensor node SN_j. T_H denotes the execution time of hash operation and T_s donates the execution time of symmetric encryption/decryption.

Table 4. Performance comparisons.

	U_i	BS	GWN_j	SN_j	Total
Ali et al.'s scheme	$7T_H + 2T_S$	$3T_H + 2T_S$	$5T_H + 4T_S$	$4T_H + 1T_S$	$19T_H + 9T_S$
Response time	0.00836 s	0.00832 s	0.01044 s	0.00514 s	0.03512 s
Proposed scheme	$14T_H$	$8T_H$	$9T_H$	$6T_H$	$37T_H$
Response time	000280 s	0.00176 s	0.00126 s	0.00144 s	0.00740 s

Table 5. Simulation environment.

Hardware/Software Specification		
U_i	Mainboard	ASUSTeK Computer INC. CM5571
	CPU	Intel Core 2 Quad Q8300 @ 2.50 GHz 2.50 GHz
	Memory	4.00 GB Dual-Channel DDR3 @ 533 MHz
	OS	Windows 7 64-bit SP1
BS	Mainboard	ASUSTeK Computer INC. CM5571
	CPU	Intel Core 2 Quad Q8300 @ 2.50 GHz 2.50 GHz
	Memory	4.00 GB Dual-Channel DDR3 @ 533 MHz
	OS	Windows 7 64-bit SP1
GWN	Mainboard	IBM 46W9191
	CPU	Intel Xeon E3 1231 v3 @ 3.40 GHz 3.40 GHz
	Memory	8.00 GB Dual-Channel DDR3 @ 800 MHz
	OS	Windows Server 2008 R2 Standard 64-bit SP1
SN_j	Mainboard	ASUSTeK Computer INC. UX303LN
	CPU	Intel Core i3/i5/i7 4xxx @ 1.70 GHz
	Memory	4.00 GB Single-Channel DDR3 @ 798 MHz
	OS	Windows 8.1 64-bit
Used Programming Language and Algorithms C/C++ Hash function: SHA-1		

Ali et al.'s scheme [23] requires 19 hash function and nine symmetric en/decryption operations. Although the proposed scheme requires more hash function operations, it does not require nine symmetric en/decryption operations. Therefore, the proposed scheme provides better efficiency compared with the previous scheme.

6. Conclusions

This paper reviewed Ali et al.'s scheme and demonstrated that it cannot provide user anonymity, user traceability, session key security, and is insecure against insider attacks, perfect forward secrecy attacks, and sensor node impersonation attacks. Moreover, after a user successfully updates his/her password, Ali et al.'s scheme will immediately suffer from Denial of Service as a Service (DoSaaS) in its authentication and key agreement scheme. The proposed scheme eliminated those security weaknesses by proposed four new phases of six existing phases. To promote efficiency, the proposed scheme eliminates symmetric encryption–decryption computation that was used in the previous scheme and only utilizes hash operation in its computation. The proposed scheme not only eliminates weaknesses in Ali et al.'s scheme, but is also 80 times more efficient compares with Ali et al.'s scheme. The efficiency, security and functionalities showed in the proposed scheme overcomes Ali et al.'s scheme. Therefore, the proposed scheme is more suitable for agriculture monitoring using WSNs.

Author Contributions: Conceptualization, M.C.; formal analysis, M.C.; investigation, M.C. and T.-F.L.; methodology, T.-F.L.; project administration, T.-F.L.; resources, J.-I.P.; validation, J.-I.P.; writing – original draft, M.C.; writing – review and editing, T.-F.L.

Funding: This research was funded by Ministry of Science and Technology under the grants MOST 106-2221-E-320-001 and Tzu Chi University under the grants TCRPP107013.

Acknowledgments: Ted Knoy is appreciated for his editorial assistance.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Arjun, K.M. Indian agriculture—Status, importance and role in Indian economy. *Int. J. Res. Agric. Food Sci.* **2013**, *4*, 343–346.
2. Omorogiuwa, O.; Zivkovic, J.; Ademoh, F. The role of agriculture in the economic development of Nigeria. *Eur. Sci. J.* **2014**, *10*, 4.
3. Raza, S.A.; Ali, Y.; Mehboob, F. Role of agriculture in economic growth of Pakistan. *Int. Res. J. Financ. Econ.* **2012**, *83*.
4. Sustainable Development Solutions Network, A Global Initiative for the United Nations. Solutions for Sustainable Agriculture and Food Systems. Available online: http://unsdsn.org/resources/publications/solutions-for-sustainable-agriculture-and-food-systems/ (accessed on 11 December 2018).
5. Luis, R.G.; Lunadei, L.; Barreiro, P.; Robla, J.I. A review of wireless sensor technologies and applications in agriculture and food industry: State of the art and current trends. *Sensors* **2009**, *9*, 4728–4750. [CrossRef]
6. Jiber, Y.; Harroud, H.; Karmouch, A. Precision agriculture monitoring framework based on WSN. In Proceedings of the 7th International Wireless Communications and Mobile Computing Conference, Istanbul, Turkey, 4–8 July 2011; pp. 2015–2020. [CrossRef]
7. Anurag, D.; Roy, S.; Bandyopadhyay, S. Agro-sense: Precision agriculture using sensor-based wireless mesh networks. In Proceedings of the First ITU-T Kaleidoscope Academic Conference Innovation in NGN: Future Network and Services, Geneva, Switzerland, 12–13 May 2008; pp. 383–388. [CrossRef]
8. Panchard, J.; Papadimitratos, P.; Hubaux, J.P.; Rao, P.R.S.; Sheshshayee, M.S.; Kumar, S. Wireless Sensor Networking for Rain-fed Farming Decision Support. In Proceedings of the ACM SIGCOMM Workshop on Networked Systems for Developing Regions, Seattle, WA, USA, 17–22 August 2008; pp. 31–36. [CrossRef]
9. Das, A.K.; Sharma, P.; Chatterje, S.; Sing, J.K. A dynamic password-based user authentication scheme for hierarchical wireless sensor networks. *J. Netw. Comput. Appl.* **2012**, *35*, 1646–1656. [CrossRef]
10. Xue, K.P.; Ma, C.S.; Hong, P.L.; Ding, R. A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks. *J. Netw. Comput. Appl.* **2013**, *36*, 316–323. [CrossRef]
11. Shi, W.B.; Gong, P. A new user authentication protocol for wireless sensor networks using elliptic curves cryptography. *Int. J. Distrib. Sens. Netw.* **2013**, *9*, 4. [CrossRef]
12. Li, C.T.; Weng, C.Y.; Lee, C.C. An advanced temporal credential-based security scheme with mutual authentication and key agreement scheme for wireless sensor networks. *Sensors* **2013**, *13*, 9589–9603. [CrossRef]
13. Rahman, M.; Sampalli, S. An Efficient Pairwise and Group Key Management Protocol for Wireless Sensor Network. *Wirel. Pers. Commun.* **2015**, *84*, 2035–2053. [CrossRef]
14. Lee, T.F. Efficient and secure temporal credential-based authenticated key agreement scheme using extended chaotic maps for wireless sensor networks. *Sensors* **2015**, *15*, 14960–14980. [CrossRef]
15. He, D.P.; Kumar, N.; Chilamkurti, N. A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks. *Inf. Sci.* **2015**, *321*, 263–277. [CrossRef]
16. Amin, R.; Islam, S.K.H.; Biswas, G.P.; Khan, M.K.; Kumar, N. A robust and anonymous patient monitoring system using wireless sensor networks. *Future Gener. Comput. Syst.* **2018**, *80*, 483–495. [CrossRef]
17. Kumari, S.; Li, X.; Wu, F.; Das, A.K.; Arshad, H.; Khan, M.K. A user friendly mutual authentication and key agreement scheme for wireless sensor networks using chaotic maps. *Future Gener. Comput. Syst.* **2016**, *63*, 56–75. [CrossRef]
18. Zebboudj, S.; Cherifi, F.; Mohammedi, M.; Omar, M. Secure and efficient ECG-based authentication scheme for medical body area sensor networks. *Smart Health* **2017**, *3–4*, 75–84. [CrossRef]
19. Li, X.; Ibrahim, M.H.; Kumari, S.; Sangaiah, A.K.; Gupta, V.; Choo, K.K.R. Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks. *Comput. Netw.* **2017**, *2*, 429–443. [CrossRef]

20. Liu, J.W.; Li, Q.; Yan, R.; Sun, S. Efficient authenticated key exchange protocols for wireless body area networks. *Eur. J. Wirel. Commun. Netw.* **2015**, *188*. [CrossRef]
21. Gupta, R.; Sultania, K.; Singh, P.; Gupta, A. Security for Wireless Sensor Networks in Military Operations. In Proceedings of the Fourth International Conference on Computing, Communications and Networking Technologies, Tiruchengode, India, 4–6 July 2013. [CrossRef]
22. Costa, D.G.; Figueredo, S.; Oliveira, G. Cryptography in Wireless Multimedia Sensor Networks: A Survey and Research Directions. *Cryptography* **2017**, *1*, 4. [CrossRef]
23. Ali, R.; Pal, A.K.; Kumari, S.; Karuppiah, M.; Conti, M. A secure user authentication and key-agreement scheme using wireless sensor networks for agriculture monitoring. *Future Gener. Comput. Syst.* **2018**, *84*, 200–215. [CrossRef]
24. Mesit, J.; Brust, M.R. Secured node-to-node key agreement for wireless sensor networks. In Proceedings of the 2015 International Conference on Information Networking (ICOIN), Siem Reap, Cambodia, 12–14 January 2015; pp. 37–39.
25. Pecori, R.; Veltri, L. A Key Agreement Protocol for P2P VoIP Applications. In Proceedings of the International Conference on Software Telecommunications and Computer Networks, Hvar, Croatia, 24–26 September 2009; pp. 276–280.
26. Pecori, R. A PKI-free Key Agreement Protocol for P2P VoIP Applications. In Proceedings of the ICC 2012 1st International Workshop on Security and Forensics in Communication Systems, Ottawa, ON, Canada, 10–12 June 2012; pp. 6748–6752. [CrossRef]
27. Lee, T.F. An efficient dynamic id-based user authentication scheme using smart cards without verifier tables. *Appl. Math. Inf. Sci.* **2015**, *9*, 485–490. [CrossRef]
28. Xue, K.P.; Hong, P.L.; Ma, C.S. A lightweight dynamic pseudonym identity based authentication and key agreement protocol without verification tables for multi-server architecture. *J. Comput. Syst. Sci.* **2014**, *80*, 195–206. [CrossRef]
29. Lee, C.C.; Li, C.T.; Chiu, S.T.; Lai, Y.M. A new three-party-authenticated key agreement scheme based on chaotic maps without password table. *Nonlinear Dyn.* **2015**, *79*, 2485–2495. [CrossRef]
30. Jokhio, S.H.; Jokhio, I.A.; Kemp, A.H. Node capture attack detection and defence in wireless sensor networks. *IET Wirel. Sens. Syst.* **2012**, *2*, 161–169. [CrossRef]
31. Burrows, M.; Abadi, M.; Needham, R. A logic of authentication. *ACM Trans. Comput. Syst.* **1990**, *8*, 1836. [CrossRef]
32. Fan, C.I.; Lin, Y.H. Provably secure remote truly three-factor authentication scheme with privacy protection on biometrics. *IEEE Trans. Inf. Forensics Secur.* **2009**, *4*, 933–945. [CrossRef]