An Enhanced Lightweight Anonymous Authentication Scheme for a Scalable Localization Roaming Service in Wireless Sensor Networks

Youngseok Chung 1,2, Seokjin Choi 1, Youngsook Lee 3, Namje Park 4 and Dongho Won 2,*

1 Electronics and Telecommunications Research Institute, Daejeon 34044, Korea; yschung11@nsr.re.kr (Y.C.); choisj@nsr.re.kr (S.C.)
2 Department of Computer Engineering, Sungkyunkwan University, Suwon 16419, Korea
3 Department of Cyber Security, Howon University, Gunsan 54058, Korea; ysooklee@howon.ac.kr
4 Department of Computer Education, Jeju National University, Jeju 63243, Korea; namjepark@jejunu.ac.kr
* Correspondence: dhwon@security.re.kr; Tel.: +82-31-290-7107

Abstract: More security concerns and complicated requirements arise in wireless sensor networks than in wired networks, due to the vulnerability caused by their openness. To address this vulnerability, anonymous authentication is an essential security mechanism for preserving privacy and providing security. Over recent years, various anonymous authentication schemes have been proposed. Most of them reveal both strengths and weaknesses in terms of security and efficiency. Recently, Farash et al. proposed a lightweight anonymous authentication scheme in ubiquitous networks, which remedies the security faults of previous schemes. However, their scheme still suffers from certain weaknesses. In this paper, we prove that Farash et al.’s scheme fails to provide anonymity, authentication, or password replacement. In addition, we propose an enhanced scheme that provides efficiency, as well as anonymity and security. Considering the limited capability of sensor nodes, we utilize only low-cost functions, such as one-way hash functions and bit-wise exclusive-OR operations. The security and lightness of the proposed scheme mean that it can be applied to roaming service in localized domains of wireless sensor networks, to provide anonymous authentication of sensor nodes.

Keywords: anonymity; privacy; authentication; security; roaming service; wireless sensor network

1. Introduction

Privacy protection and security provision have been of great concern in proportion to the number of sensor nodes in wireless sensor networks. In addition, due to the features of wireless environments, efficiency is one noticeable aspect. The characteristics of low transmission bandwidth, insufficient memory, low computing power, and battery dependency demand more lightweight and efficient security mechanisms that provide a similar level of security to wired environments. Considering a mobile sensor node that travels in various networks and wants to receive roaming service from a foreign agent, an anonymous authentication scheme is necessary to preserve the sensor node’s privacy and security. If the scheme is also lightweight, it is more suitable for wireless sensor networks. Figure 1 illustrates a simple model of wireless sensor networks for roaming service. If a mobile sensor node registered for its home agent visits a foreign network, it wants to access the foreign agent to receive roaming service. The foreign agent then needs to check the identification of the sensor node through its home agent. In this situation, a lightweight anonymous authentication scheme is necessary to guarantee secure authentication and efficient communication.
In recent years, various anonymous authentication schemes and related protocols in wireless networks have been proposed [1–25]. They have been followed by proofs of vulnerability of the schemes and associated improvements. Some of these schemes use high-cost functions, such as symmetric cryptographic functions, asymmetric cryptographic functions, and modular operations [1–18]. On the other hand, the others are based on low-cost functions, such as one-way hash functions and bit-wise exclusive-OR operations [19–25]. To analyze these schemes, we categorize them into two groups according to the computation cost: schemes based on high-cost functions and schemes based on low-cost functions. If all of them provide the same security level, schemes based on low-cost functions are more suitable for wireless sensor networks, since they consume less energy. Farash et al. [18] proposed one of the most recent anonymous authentication schemes for roaming service. They claimed that their scheme improved security and reduced computation time. However, their scheme still has security weaknesses, and does not have computational benefit.

The contributions of this paper are two points. Firstly, we point out that Farash et al.’s scheme does not provide anonymity against a legitimate but malicious adversary, foreign agent authentication, or password replacement. In addition, we present that Farash et al.’s scheme has less computational merits than our proposed scheme, even if their scheme is superior to other previous schemes in terms of the computation cost. Secondly, we propose an enhanced lightweight anonymous authentication scheme that resolves the above weaknesses. Our proposed scheme has the advantage of security and efficiency. In other words, it has enhanced security features and resistance against well-known attacks, as well as the fastest running time among other schemes. More specifically, the proposed scheme preserves weak and strong anonymity, hop-by-hop authentication, and untraceability; resistance against password guessing, impersonation, forgery, and known session key attack; and fair key agreement. There have been no recent schemes, which guarantee all the above. In addition, since the proposed scheme is based only on low-cost functions, it runs faster and more efficiently than previous schemes. Although most schemes including ours are adaptable to wireless sensor networks, our proposed scheme, due to better efficiency, has superiority over other previous schemes.

The remainder of this paper is organized as follows. Section 2 briefly describes related works, and Section 3 reviews Farash et al.’s scheme. Section 4 then presents its weaknesses. Section 5 proposes our enhanced scheme, and Section 6 presents the formal analysis of our proposed protocol. Sections 7 and 8 then analyze the security and performance of our scheme, respectively. Finally, Section 9 concludes the paper.

2. Related Works

Previous schemes, which have recently been proposed, show the following research trends. Zhu and Ma [1] in 2004 proposed an anonymous authentication scheme based on high-cost functions, and Lee et al. [2] proved that it has security weaknesses. Wu et al. [3] argued that both Zhu and
Ma’s and Lee et al.’s schemes fail to preserve anonymity and backward secrecy, and they presented improvements of Lee et al.’s scheme. However, Lee et al. [5] and Xu [6] showed vulnerabilities of Wu et al.’s scheme. Kun et al. [7] improved Xu and Feng’s scheme, but Tsai et al. [8] showed that Kun et al.’s scheme is also vulnerable. In addition, Mun et al. [9] showed Wu et al.’s scheme suffers from various attacks, and proposed an enhanced scheme. However, Zhao et al. [10] proved that Mun et al.’s scheme is insecure.

Independently, Chang et al. [19] in 2009 proposed an enhanced authentication scheme that uses only low-cost functions. Unfortunately, Youn et al. [20] proved that Chang et al.’s scheme is vulnerable. In addition, Zhou and Xu [13] have proved that Chang et al.’s scheme has weaknesses, and they proposed an improved scheme. Lately, Gope and Hwang [24] have proved that Zhou and Xu’s scheme suffers from some security faults, such as unsuccessful key agreement and vulnerability to replay attack. They showed that a malicious adversary, by replacing transmission messages, can disturb valid communication between a normal user and a foreign agent. In addition, they proved that an attacker can successfully retransmit authentication messages that have been transmitted during a previous session of communication. At the same time, they proposed an improved scheme. Their improved scheme guarantees several security features as follows. Since all participants can normally verify parameters in each message, their scheme preserves mutual authentication. The fact that each participant makes the same contribution to the freshness of a session key provides their scheme with fair key agreement. In addition, both passive eavesdroppers and active intruders cannot identify or keep track of a normal user. Since only a legitimate user can form a valid one-time-alias using a real identity, secret value, nonce, and timestamp, no attackers can forge the alias to cheat users. In addition, it is impossible to accomplish a known session key attack because there is no significant relation among any session keys. It means that the compromised session key never helps to recover any past or future session keys. Moreover, since their scheme is based on low-cost functions, it has computational merits.

Meanwhile, He et al. [21] proved that Chang et al.’s scheme has a security fault, and that their scheme is not efficient. After that, Jiang et al. [14] showed the weaknesses of He et al.’s scheme. They proposed an enhanced protocol, but Wen et al. [15] presented its weaknesses. Subsequently, Gope and Hwang [17] showed that Wen et al.’s scheme suffers from several attacks. In Wen et al.’s scheme, an attacker, by performing an exhaustive search operation of all possible values, can obtain secret information stored in the lost or stolen smart card. After jamming all transmission messages and resetting a counter, he or she can also establish a session key between a normal user and a foreign agent. Since a session key contains only one random number generated by one side of the participants, it fails to preserve fair key agreement. In addition, Gope and Hwang proposed an enhanced scheme that preserves mutual authentication, fair key agreement, user anonymity, resistance against forgery attack, and security assurance in the case of a lost smart card. In their schemes, all participants can authenticate each other by verifying parameters. While computing a session key, each participant contributes equally, by providing independent random numbers. Since the difficulty of the quadratic residue problem makes a real identity secure, the identity cannot be revealed. No attackers can forge transmission messages because they do not have the knowledge of a secret key and a real identity. In addition, an attacker cannot use the lost or stolen smart card to perform any masquerade attacks because there is no way to obtain a secret key, an identity, and a password from the smart card.

In addition, Shin et al. [16] proved He et al.’s scheme is vulnerable, and proposed an improved scheme. Then, Farash et al. simultaneously presented the vulnerabilities of both Wen et al.’s scheme and Shin et al.’s scheme, proving that Wen et al.’s scheme suffers from session key disclosure attack and known session key attack, while Shin et al.’s scheme does not guarantee untraceability, secrecy of the sensitive parameter of home agent, secrecy against impersonation attack, or session key secrecy. Farash et al. also proposed an improved scheme that preserves security and reduces the computation time of their scheme.
3. Review of Farash et al.’s Scheme

In this section, we review the lightweight anonymous authentication scheme proposed by Farash et al. Their scheme consists of three phases: registration, login and authentication, and password change. Three different entities are involved in each phase. MN is a mobile node that wants to receive roaming service while visiting a foreign network. FA is the foreign agent of a foreign network, and HA is the home agent of the mobile node MN. When MN visits a foreign network, it sends a login request message to FA to be authenticated. Then, FA sends an authentication request message to HA for authentication of MN, since FA is not the home agent of MN, and it cannot directly check MN’s identity. After HA authenticates MN using the message received from FA, HA sends a response message to FA. Finally, FA sends a response message to MN and shares a common session key with MN. In this process, it is supposed that HA and FA are in a trusting relationship, and that they secretly share and store a long-term secret key. Because of this, it is possible for FA to anonymously authenticate MN through HA. Table 1 denotes the notations used in this paper.

| Notation | Description |
|----------|-------------|
| HA | Home agent |
| FA | Foreign agent |
| MN | Mobile node |
| ID_X | Identity of an entity X |
| PW_MN | Password of MN |
| KFH | Pre-shared secret key between HA and FA |
| K_X | Secret key of an entity X |
| n_X | Random nonce generated by an entity X |
| t_X | Timestamp generated by an entity X |
| E_K(.)/D_K(.) | Symmetric encryption and decryption using a secret key K |
| h(.) | Collision free one-way hash function |
| || | Concatenation |
| ⊕ | Bit-wise exclusive-OR operation |

3.1. Registration Phase

To register for HA, MN first selects ID_MN, PW_MN, and the random number r. Then, MN sends ID_MN and h(PW_MN||r) to HA in a secure manner. After receiving ID_MN and h(PW_MN||r), HA computes the following parameters for MN:

\[ A_{MN} = h(K_H) \oplus h(ID_{MN}) \]  

\[ B_{MN} = h(K_H||ID_{MN}) \oplus h(PW_{MN}||r) \]  

Next, HA sends A_MN, B_MN, and h(.) to MN; and MN stores r, as well as A_MN, B_MN, and h(.)

3.2. Login and Authentication Phase

MN and FA perform the login and authentication phase to achieve the following goals with the aid of HA:

- FA anonymously authenticates MN;
- MN and FA mutually authenticate each other;
- MN and FA share a session key.

In this phase, it is supposed that the common secret key KFH is shared between FA and HA beforehand. Figure 2 illustrates the login and authentication phase. The procedure of this phase is as follows:
In this phase, it is supposed that the common secret key $KFH$ is shared between $FA$ and $HA$.

(1) $MN$ inputs $ID_{MN}$ and $PW_{MN}$. Then $MN$ generates the random nonce $n_{MN}$, and loads $r$, $A_{MN}$, $B_{MN}$, and $h(.)$ to compute $MN$’s verifiers:

$$MV_1 = A_{MN} \oplus h(ID_{MN})$$ (3)

$$MV_2 = MV_1 \oplus n_{MN}$$ (4)

$$MV_3 = h(MV_1 || n_{MN}) \oplus ID_{MN}$$ (5)

$$MV_4 = B_{MN} \oplus h(PW_{MN} || r)$$ (6)

$$MV_5 = h(MV_2 || MV_3 || MV_4)$$ (7)

Next, $MN$ sends the message $M_1 = \{MV_2, MV_3, MV_4\}$ to $FA$.

(2) Upon receiving $M_1$, $FA$ generates the random nonce $n_{FA}$, and encrypts $M_1$ and $n_{FA}$ using the symmetric encryption function such that $E_{KFH}(M_1, n_{FA})$. Then, $FA$ sends the message $M_2 = \{ID_{FA}, E_{KFH}(M_1, n_{FA})\}$ to $HA$.

(3) After receiving $M_2$, $HA$ first checks $ID_{FA}$ to confirm that $FA$ is a valid agent. If so, $HA$ retrieves $KFH$, and makes the following computations:

$$D_{KFH}(E_{KFH}(M_1, n_{FA}))$$ (8)

$$n^*_{MN} = MV_2 \oplus h(K_H)$$ (9)

$$ID^*_{MN} = MV_3 \oplus h(h(K_H)||n^*_{MN})$$ (10)

$$MV^*_4 = h(K_H||ID^*_{MN})$$ (11)

If $HA$ checks the equivalence between the received $MV_5$ and the computed $h(MV_2 || MV_3 || MV^*_4)$ normally, $HA$ computes the following session key, and encrypts it with $KFH$:

$$SK_{FA} = h(MV^*_4 || n_{MN} || n_{FA} || ID_{MN} || ID_{FA})$$ (12)

Then, $HA$ sends the message $M_3 = \{E_{KFH}(SK_{FA})\}$ to $FA$. 

**Figure 2.** Login and authentication phase in Farash et al.’s scheme.
(4) After receiving $M_3$, $FA$ decrypts the encrypted session key, and computes $FA$’s verifier:

$$FV_1 = h(SK_{FA} || n_{FA})$$  (13)

Then, $FA$ sends the message $M_4 = \{ID_{FA}, FV_1, n_{FA}\}$ to $MN$. Upon receiving $M_4$, $MN$ computes the session key:

$$SK_{MN} = h(MV_4 \| n_{MN} \| n_{FA} \| ID_{MN} \| ID_{FA})$$  (14)

By checking the validity of the session key after computing $h(SK_{MN} || n_{FA})$, $MN$ confirms that $FA$ successfully authenticates $MN$, and the session key is established between them at the same time.

3.3. Password Change Phase

$MN$, which wants to change its password, is supposed to perform the password change phase. In this phase, $MN$ renews the password after acquiring the confirmation from $HA$. Figure 3 describes the password change phase.

![Figure 3. Password change phase in Farash et al.’s scheme.](image)

(1) $MN$ inputs the identity $ID_{MN}$, the current $PW_{MN}$, and the new password $PW_{MN}^{\text{new}}$. Then, $MN$ generates the random nonce $n_{MN}$, and computes the following verifiers in a similar way to what it does in the login and authentication phase:

$$MV_1 = A_{MN} \oplus h(ID_{MN})$$  (15)

$$MV_2 = MV_1 \oplus n_{MN}$$  (16)

$$MV_3 = h(MV_1 \| n_{MN}) \oplus ID_{MN}$$  (17)

$$MV_4 = B_{MN} \oplus h(PW_{MN})$$  (18)

$$MV_5 = h(MV_2 \| MV_3 \| MV_4 \| ID_{MN})$$  (19)

Next, $MN$ sends the message $M_1 = \{MV_2, MV_3, MV_5\}$ to $HA$.

(2) Upon receiving $M_1$, $HA$ computes $n^*_{MN}$, $ID^*_{MN}$, and $MV^*_{4}$ as shown in Equations (20)–(22):

$$n^*_{MN} = MV_2 \oplus h(K_H)$$  (20)

$$ID^*_{MN} = MV_5 \oplus h(K_H) || n^*_{MN}$$  (21)

$$MV^*_{4} = h(K_H) || ID^*_{MN}$$  (22)

After computing $h(MV_2 \| MV_3 \| MV^*_{4} || ID^*_{MN})$, $HA$ checks the validity of $MV_5$. The successful check means $HA$ authenticates $MN$ normally. Then, $HA$ computes the following verifier $HV_1$, and sends $M_2 = \{HV_1\}$ to $MN$: 

$$HV_1 = h(MV_2 \| MV_3 \| MV^*_{4} || ID^*_{MN})$$
After receiving $M_2$, $MN$ checks the equivalence between $HV_1$ and $h(MV_4||n_{MN}||ID_{MN})$ to confirm that $HA$ has successfully authenticated $MN$. Finally, $MN$ computes the following $B_{MN}^{new}$ and replaces $B_{MN}$ with $B_{MN}^{new}$.

$$B_{MN}^{new} = B_{MN} \oplus h(PW_{MN}) \oplus h(PW_{MN}^{new})$$

4. Weaknesses of Farash et al.’s Scheme

Farash et al. proved that their scheme guarantees $MN$ authentication, $FA$ authentication, anonymity and untraceability, resistance against offline password guessing attack, secure key establishment, and no verification table at $HA$. However, there still remain several security weaknesses in their scheme. In this section, we will prove that Farash et al.’s scheme does not guarantee anonymity or $FA$ authentication. In addition, we will show that their scheme does not achieve password replacement.

4.1. Anonymity

Farash et al.’s scheme guarantees anonymity against a foreign agent and a normal mobile node. However, it does not preserve anonymity against a malicious mobile node. Suppose that there is a malicious mobile node $MN'$ normally registered for $HA$, as in Farash et al.’s attack scenario. Then, $MN'$ can get $ID_{MN}$ by accomplishing the following procedures:

1. $MN'$ inputs $ID_{MN'}$ and $PW_{MN'}$. Then, $MN'$ can get $h(K_H)$.
2. To get $n_{MN}$, $MN'$ eavesdrops $M_1 = \{MV_2, MV_3, MV_5\}$, and computes $MV_2 \oplus h(K_H)$. Since the equation described below holds, $MN'$ can successfully get $n_{MN}$:

$$MV_2 \oplus h(K_H) = MV_1 \oplus n_{MN} \oplus h(K_H) = A_{MN} \oplus h(ID_{MN}) \oplus n_{MN} \oplus h(K_H)$$

$$= h(K_H) \oplus h(ID_{MN}) \oplus h(ID_{MN}) \oplus n_{MN} \oplus h(K_H) = n_{MN}$$

3. Next, by computing as follows, $MN'$ gets $ID_{MN}$:

$$MV_3 \oplus h(h(K_H)||n_{MN}) = h(MV_1||n_{MN}) \oplus ID_{MN} \oplus h(K_H)||n_{MN})$$

$$= h(A_{MN} \oplus h(ID_{MN})||n_{MN}) \oplus ID_{MN} \oplus h(K_H)||n_{MN})$$

$$= h(h(K_H) \oplus h(ID_{MN}) \oplus h(ID_{MN})||n_{MN}) \oplus ID_{MN} \oplus h(K_H)||n_{MN}) = ID_{MN}$$

As a result, a malicious mobile node that eavesdrops the message can easily know the other’s identity, so anonymity is not guaranteed.

4.2. Authentication

In Farash et al.’s scheme, $HA$ can authenticate both $MN$ and $FA$. In $MN$’s case, after computing $n_{MN}$ from $MV_2$ and $ID_{MN}$ from $MV_3$, $HA$ can authenticate $MN$ by checking the equivalence between $MV_5$ and $h(MV_2||MV_5||MV_4)$. In addition, $HA$ can authenticate $FA$ by a successful decryption using the pre-shared secret key $KFH$ corresponding to $FA$’s identity. Meanwhile, $FA$ is able to anonymously authenticate $MN$ with the aid of $HA$. In addition, a successful decryption using $KFH$ makes $FA$ check $HA$’s identity. This is the same as what $HA$ does. However, while it is possible to authenticate $HA$, $MN$ cannot authenticate $FA$. There is no obvious way for $MN$ to confirm the received message that is made with the aid of $HA$. The reason is as follows. After receiving $M_4 = \{ID_{FA}, FV_1, n_{FA}\}$ from $FA$, $MN$ just computes the following session key $SK_{MN}$, without checking any verifiers computed by $HA$:

$$SK_{MN} = h(MV_4||n_{MN}||n_{FA}||ID_{MN}||ID_{FA})$$
Clearly, $SK_{MN}$ contains $MV_4$ which is computed as follows:

$$MV_4 = B_{MN} \oplus h(PW_{MN}||r) = h(K_H||ID_{MN})$$

Since $HA$ is the only entity that can compute $h(K_H||ID_{MN})$, $MN$ can only authenticate $HA$ through a successful checking of $SK_{MN}$, namely $FV_1 = h(SK_{FA}||n_{FA})$. This implies that there is no way for $MN$ to authenticate $FA$. In addition, if failure while checking $FV_1$ occurs, $MN$ cannot confirm whether $HA$ or $FA$ is illegal. For these reasons, authenticating the foreign agent is impossible.

### 4.3. Password Replacement

In the password change phase, $MN$ computes the following $MV_4$, while $HA$ computes $MV^*_4$:

$$MV_4 = B_{MN} \oplus h(PW_{MN}) = h(K_H||ID_{MN}) \oplus h(PW_{MN}||r) \oplus h(PW_{MN})$$

$$MV^*_4 = h(K_H||ID^*_{MN})$$

Since $MV_4$ is not equal to $MV^*_4$, there is no equivalence between $MV_5$ and $h(MV_2||MV_3||MV^*_4||ID^*_{MN})$, as shown in:

$$MV_5 = h(MV_2||MV_3||MV_4||ID_{MN}) = h(MV_2||MV_3||B_{MN} \oplus h(PW_{MN})||ID_{MN})$$

$$= h(MV_2||MV_3||h(K_H||ID_{MN}) \oplus h(PW_{MN}||r) \oplus h(PW_{MN})||ID_{MN})$$

$$\neq h(MV_2||MV_3||MV^*_4||ID^*_{MN}) = h(MV_2||MV_3||h(K_H||ID^*_{MN})||ID^*_{MN})$$

To originally authenticate $MN$, $HA$ needs to confirm the validity of $MV_5$; but it is impossible to check this. As a result, the home agent cannot authenticate a mobile node in the password change phase, and the password replacement cannot be accomplished. Moreover, there is no $HA$ contribution to change the password. This means that by computing $B_{MN}^{new} = B_{MN} \oplus h(PW_{MN}) \oplus h(PW_{MN}^{new})$, $MN$ can change the password as it wants, without accomplishing any other steps.

### 5. The Proposed Scheme

In this section, we propose an enhanced scheme to remedy the faults of Farash et al.’s scheme. Our scheme also consists of three phases. In each phase, $MN$, $FA$, and $HA$ are involved, and use a timestamp as a nonce. After receiving a message, they first validate a timestamp to ensure that old messages cannot be used in replay attacks. We use the same terms as Farash et al.’s scheme does. However, to apply our scheme to wireless sensor networks, we regard $MN$ as a mobile sensor node. Clearly, $FA$ and $HA$ are server systems, which have a powerful computing capability. On the other hand, $MN$ is a battery-powered sensor node, which has less computing capability. In the registration phase, $MN$ registers for $HA$, and $HA$ gives $MN$ secret parameters in a secure manner. $MN$ and $HA$ establish a trusting relationship through this phase. Then, $MN$ roams in a foreign network, and tries to receive roaming service from $FA$. Since $MN$ is not a mobile sensor node of $FA$, $FA$ wants to authenticate $MN$ through $HA$. For this, the login and authentication phase is necessary. It is assumed that $FA$ and $HA$ share a long-term secret key $KFH$ beforehand, the same as in Farash et al.’s scheme. $FA$ and $HA$ are supposed to use $KFH$ when they try to authenticate each other. Meanwhile, in the password change phase, $MN$, with the aid of $HA$, securely changes the secret key, as well as the password. Each phase is described in detail as follows.

#### 5.1. Registration Phase

The first thing $MN$ accomplishes is to register for $HA$ in the registration phase. Figure 4 shows this phase:
MN selects its identity $ID_{MN}$ and the password $PW_{MN}$. In addition, MN chooses the random number $r$ as a salt of a one-way hash function. MN then submits $ID_{MN}$ and $h(PW_{MN}||r)$ to HA through a secure channel. Upon receiving the registration request message, HA, using its secret key $K_H$, computes three secret parameters for MN as follows:

$$A_{MN} = h(K_H) \oplus h(ID_{MN})$$  \hspace{1cm} (25)

$$B_{MN} = h(K_H||ID_{MN}) \oplus h(PW_{MN}||r)$$  \hspace{1cm} (26)

$$C_{MN} = K_M \oplus h(PW_{MN}||r)$$  \hspace{1cm} (27)

where $K_M$ is the secret key allocated only to MN. Then, HA secretly stores $\{h(K_H||ID_{MN}), K_M, ID_{MN}\}$ in its database, and sends $A_{MN}, B_{MN}, C_{MN}$, and $h(.)$ to MN in a secure way. Finally, MN stores $r, A_{MN}, B_{MN}, C_{MN}$, and $h(.)$.

5.2. Login and Authentication Phase

When MN visits a foreign network and logs in to FA, FA anonymously authenticates MN through HA. MN and FA then share a session key for secure communication. Figure 5 illustrates the login and authentication phase:

Figure 4. Registration phase in the proposed scheme.

Figure 5. Login and authentication phase in the proposed scheme.
(1) MN inputs $ID_{MN}$ and $PW_{MN}$ to make the login request message. Then, MN generates the timestamp $t_{MN}$, and loads $r, A_{MN}, B_{MN}, C_{MN},$ and $h(\cdot)$ to compute MN’s verifiers:

$$MV_1 = A_{MN} \oplus h(ID_{MN})$$
$$MV_2 = B_{MN} \oplus (PW_{MN} \| r)$$
$$MV_3 = C_{MN} \oplus (PW_{MN} \| r)$$
$$MV_4 = MV_1 \oplus MV_2 \oplus t_{MN}$$
$$MV_5 = h(MV_3 \| ID_{MN} \| t_{MN})$$

Next, MN sends the login request message $M_1 = \{MV_4, MV_5, t_{MN}, ID_{HA}\}$ to FA.

(2) FA, which receives $M_1$ from MN, first checks $t_{MN}$ to confirm whether it is valid or not. If FA confirms the validity of $t_{MN}$, FA also generates the timestamp $t_{FA}$, and computes FA’s verifier as follows:

$$FV_1 = h(KFH \| MV_4 \| MV_5 \| t_{MN} \| t_{FA})$$

Then, FA sends the authentication request message $M_2 = \{M_1, FV_1, t_{FA}, ID_{FA}\}$ to HA.

(3) After receiving $M_2$, HA first checks $t_{MN}$ and $ID_{FA}$ to confirm whether $t_{FA}$ is valid or not, as well as whether FA is an ally or not. If HA confirms the validities of both $t_{FA}$ and $ID_{FA}$, HA fetches the secret key $KFH$ corresponding to $ID_{FA}$, and checks the equivalence between $FV_1$ and $h(KFH \| MV_4 \| MV_5 \| t_{MN} \| t_{FA})$. If they are equal, HA computes:

$$h(K_H \| ID_{MN}) = MV_4 \oplus h(K_H) \oplus t_{MN}$$

Then, HA searches $\{K_{MN}, ID_{MN}\}$ from its database, using $h(K_H \| ID_{MN})$ as a keyword. If there are no value matches with $h(K_H \| ID_{MN})$ in the database, HA regards $M_2$ as a forged message. In this case, HA does not move on to the next step, and informs FA of this. Otherwise, HA checks the equivalence between $MV_5$ and $h(K_H \| ID_{MN} \| t_{MN})$. If this is successfully verified, HA generates the timestamp $t_{HA}$, and computes the session key and HA’s verifiers:

$$SK = h(K_M \| t_{MN} \| t_{FA} \| ID_{MN} \| ID_{FA})$$
$$HV_1 = h(SK \| K_M \| t_{MN})$$
$$HV_2 = SK \oplus h(KFH \| t_{FA})$$
$$HV_3 = h(KFH \| HV_1 \| HV_2 \| ID_{HA} \| t_{HA})$$

Then, HA sends the authentication response message $M_3 = \{HV_1, HV_2, HV_3, t_{HA}\}$ to FA.

(4) Upon receiving $M_3$, FA computes $h(KFH \| HV_1 \| HV_2 \| ID_{HA} \| t_{HA})$ after checking $t_{HA}$, and checks it equals $HV_3$. If the equality holds, FA computes the following session key and FA’s verifier, to send the login response message $M_4 = \{HV_1, FV_2, t_{FA}\}$ to MN:

$$SK = HV_2 \oplus h(KFH \| t_{FA})$$
$$FV_2 = SK \oplus h(SK \| t_{FA})$$

(5) After receiving $M_4$, MN first checks $t_{FA}$, and computes the session key:

$$SK = h(K_M \| t_{MN} \| t_{FA} \| ID_{MN} \| ID_{FA})$$

To authenticate HA, MN checks the equivalence between $HV_1$ and $h(SK \| K_M \| t_{MN})$. If this is confirmed normally, MN obtains $SK'$ by computing $FV_2 \oplus h(SK \| t_{FA})$. Then, MN checks $SK$
equals SK’ to authenticate FA. Finally, MN and FA complete mutual authentication of each other, and share the session key between them.

5.3. Password Change Phase

In this phase, MN not only renews its password, but also the secret key. MN can change the password for itself, without being authenticated by HA. However, in order to change the secret key, it is necessary for MN to accomplish the password change procedure with HA. Figure 6 shows this phase:

(1) MN inputs its identity ID_MN, the current PW_MN, and the new password PW^new_MN. In addition, MN chooses the new random number r^new as a new salt of a one-way hash function, and generates the timestamp t_MN. Then, MN computes the following verifiers in the same form as they are in the login and authentication phase:

\[ MV_6 = A_{MN} \oplus h (ID_{MN}) \]  
\[ MV_7 = B_{MN} \oplus (PW_{MN} || r) \]  
\[ MV_8 = C_{MN} \oplus (PW_{MN} || r) \]  
\[ MV_9 = MV_6 \oplus MV_7 \oplus t_{MN} \]  
\[ MV_{10} = h (MV_6 || ID_{MN} || t_{MN}) \]

Next, MN sends the message \( M_S = \{ MV_9, MV_{10}, t_{MN} \} \) to HA.

(2) HA, after receiving \( M_S \), checks \( t_{MN} \) to confirm whether it is valid or not. Then, HA computes \( h(K_H || ID_{MN}) = MV_6 \oplus h (K_H) \oplus t_{MN} \). In addition, HA, using \( h(K_H || ID_{MN}) \) as a keyword, searches for \( \{ K_{MN}, ID_{MN} \} \) from its database. If it is impossible to search \( \{ K_{MN}, ID_{MN} \} \) due to no matching value, HA immediately stops continuing, and informs MN of this. If not, HA computes \( h (K_M || ID_{MN} || t_{MN}) \), and checks that it equals \( MV_{10} \). To renew the secret key \( K_M \), HA generates the timestamp \( t_HA \) and the new secret key \( K^{'new}_M \), and computes the following verifiers:

\[ HV_4 = K^{'new}_M \oplus h (K_M || t_{MN}) \]
Then, after replacing \( \{h(K_H || ID_{MN}), K_M, ID_{MN}\} \) with \( \{h(K_H || ID_{MN}), K_{M}^{\text{new}}, ID_{MN}\} \), HA sends the message \( M_6 = \{HV_4, HV_5, t_{HA}\} \) to MN.

(3) Upon receiving \( M_6 \), MN validates \( t_{HA} \), and then checks the equivalence between \( HV_5 \) and \( h(h(K_M || TV_4) || t_{HA}) \). If both \( t_{HA} \) and \( HV_5 \) are successfully verified, MN computes the new secret key \( K_{M}^{\text{new}} \) after checking:

\[
K_{M}^{\text{new}} = HV_4 \oplus h(K_M || ID_{MN})
\]  

Finally, MN computes the following secret parameters, and replaces \( \{r, A_{MN}, B_{MN}, C_{MN}, h(.)\} \) with \( \{r_{\text{new}}, A_{MN}, B_{MN}^{\text{new}}, C_{MN}^{\text{new}}, h(.)\} \):

\[
B_{MN}^{\text{new}} = B_{MN} \oplus h(PW_{MN} || r) \oplus h(PW_{MN}^{\text{new}} || r_{\text{new}})
\]
\[
C_{MN}^{\text{new}} = K_{M}^{\text{new}} \oplus h(PW_{MN}^{\text{new}} || r_{\text{new}})
\]

6. Protocol Analysis

In this section, we present the formal analysis of our proposed scheme using Burrows–Abadi–Needham logic [26] (also known as the BAN logic), which is a useful model to prove the validity of authentication and key agreement protocol. The main goal of the login and authentication phase in our scheme is that MN and FA authenticate each other and share a session key. Since both MN and FA participate and equally contribute while establishing a session key, it can be regarded as a two-way key agreement. In addition, in the password change phase, it is the main goal that MN and FA authenticate each other, and renew MN’s secret key. Only FA contributes while generating MN’s secret key. Therefore, renewing MN’s secret key can be regarded as a one-way key agreement.

To prove that our proposed scheme meets these goals, we need to transform the scheme into the idealized form by the analytic procedures of BAN logic. We first define the constructs and some rules of BAN logic as follows:

[Constructs]
- \( P \models X \): P believes X.
- \( P \vdash X \): P sees X.
- \( P \models \neg X \): P said X.
- \( P \models X \): P has jurisdiction over X.
- \( \#(X) \): Formula X is fresh.
- \( P \not\models K \): P and Q may use the shared key K to communicate.

[Rules]
- \( R_1 \), Message-meaning rule:
  \[
  \frac{P \models P \models K}{P \models Q} \models \{X\}_K
  \]
- \( R_2 \), Nonce-verification rule:
  \[
  \frac{P \models \#(X), P \models Q \models X}{P \models Q \models X}
  \]
- \( R_3 \), Jurisdiction rule:
  \[
  \frac{P \models Q \models X, P \models Q \models X}{P \models X}
  \]
- \( R_4 \), Fresh rule:
  \[
  \frac{P \models \#(X)}{P \models \#(X, Y)}
  \]
Then, using BAN logic rules, we transform our goals into the following forms. The login and authentication phase needs mutual authentication and a two-way key agreement. In addition, the password change phase needs mutual authentication and a one-way key agreement.

[Transformation of the goals of the login and authentication phase]

- $G_1 : MN \equiv FA \equiv MN \xrightarrow{SK} FA,$
- $G_2 : FA \equiv MN \equiv MN \xrightarrow{SK} FA,$
- $G_3 : MN \equiv MN \xrightarrow{SK} FA,$
- $G_4 : FA \equiv MN \xrightarrow{SK} FA.$

[Transformation of the goals of the password change phase]

- $G_5 : MN \equiv HA \equiv MN \xrightarrow{K_{MN}^{\text{new}}} HA,$
- $G_6 : HA \equiv MN \equiv MN \xleftarrow{K_{MN}^{\text{new}}} HA,$
- $G_7 : MN \equiv MN \xleftarrow{K_{MN}^{\text{new}}} HA.$

Next, the messages $M_1, M_2, M_3,$ and $M_4$ in Figure 5, and $M_5$ and $M_6$ in Figure 6 are transformed into the idealized messages as follows:

[Idealized messages]

- $M_1 : (h(K_H || ID_{MN}) \rightarrow h(K_H) < ID_{MN} \rightarrow H_{MN} \rightarrow ID_{MN}),$
- $M_2 : (h(K_H || ID_{MN}) \rightarrow h(K_H) < ID_{MN} \rightarrow H_{FA} \rightarrow H_{FA}),$
- $M_3 : (MN \xrightarrow{SK} FA \rightarrow (K_{MN}^{\text{new}}), MN \xrightarrow{SK} FA \rightarrow (K_{FA}^{\text{new}}), HA \rightarrow H_{FA},$
- $M_4 : (MN \xrightarrow{SK} FA \rightarrow (K_{MN}^{\text{new}}), MN \xrightarrow{SK} FA \rightarrow (K_{FA}^{\text{new}}), HA),$
- $M_5 : (h(K_H || ID_{MN}) \rightarrow h(K_H) < ID_{MN} \rightarrow H_{MN} \rightarrow H_{MN}),$
- $M_6 : (MN \xrightarrow{K_{MN}^{\text{new}}} HA \rightarrow (K_{MN}^{\text{new}}), HA \rightarrow H_{MN}).$

In addition, we make the following assumptions to analyze our proposed scheme.

[Assumptions]

- $A_1 : MN \equiv \#(t_X)$, where $X$ is $FA$ or $HA,$
- $A_2 : FA \equiv \#(t_X)$, where $X$ is $MN$ or $HA,$
- $A_3 : HA \equiv \#(t_X)$, where $X$ is $MN$ or $FA,$
- $A_4 : MN \equiv MN \xrightarrow{h(K_H)} HA,$
- $A_5 : HA \equiv MN \xleftarrow{h(K_H)} HA,$
- $A_6 : MN \equiv MN \xrightarrow{K_{HA}} HA,$
- $A_7 : HA \equiv MN \xleftarrow{K_{HA}} HA,$
- $A_8 : FA \equiv FA \xrightarrow{KH} HA,$
- $A_9 : HA \equiv FA \xleftarrow{KH} HA,$
- $A_{10} : MN \equiv HA \Rightarrow MN \xrightarrow{SK} FA,$
- $A_{11} : FA \equiv HA \Rightarrow MN \xleftarrow{SK} FA,$
- $A_{12} : MN \equiv MN \xrightarrow{SK} FA,$
- $A_{13} : HA \equiv MN \Rightarrow ID_{MN},$
- $A_{14} : MN \equiv HA \Rightarrow MN \xrightarrow{K_{MN}^{\text{new}}} HA,$
- $A_{15} : ID_{MN}$ is unknown for anyone except $MN.$
Using the above rules and assumptions, we analyze the idealized form of our proposed scheme. The following procedure shows how the proposed scheme meets the goals described above:

(1) We apply $R_4$ and $A_2$ to $M_1$ to derive the following statement:

$$FA|_{\equiv \# \left( < h(K_H||ID_{MN}) >_{h(K_H)}, < ID_{MN} >_{K_M} \right)}$$  \hspace{1cm} (S_1)

(2) We apply $R_1$ and $A_9$ to $M_2$ to derive

$$HA|_{\equiv FA|_{\sim \left( < h(K_H||ID_{MN}) >_{h(K_H)}, < ID_{MN} >_{K_M}, t_{FA} \right)}}$$  \hspace{1cm} (S_2)

(3) We apply $R_2$ and $A_3$ to $S_2$ to derive

$$HA|_{\equiv FA|_{\equiv \left( < h(K_H||ID_{MN}) >_{h(K_H)}, < ID_{MN} >_{K_M} \right)}}$$  \hspace{1cm} (S_3)

(4) To break conjunctions, we apply the rule of BAN logic to $S_3$, then get

$$HA|_{\equiv FA|_{\equiv < h(K_H||ID_{MN}) >_{h(K_H)}}}$$  \hspace{1cm} (S_4)

$$HA|_{\equiv FA|_{\equiv < ID_{MN} >_{K_M}}}$$  \hspace{1cm} (S_5)

(5) We apply $R_1$ and $A_5$ to $S_4$ to derive

$$HA|_{\equiv MN|_{\equiv h(K_H||ID_{MN})}}$$  \hspace{1cm} (S_6)

(6) We apply $R_1$ and $A_7$ to $S_5$ to derive

$$HA|_{\equiv MN|_{\equiv ID_{MN}}}$$  \hspace{1cm} (S_7)

(7) We apply $R_3$ and $A_{13}$ to $S_7$ to derive

$$HA|_{\equiv ID_{MN}}$$  \hspace{1cm} (S_8)

(8) From $A_{15}$ and $S_8$, we can deduce the following rule:

$$HA|_{\equiv MN \overset{ID_{MN}}{\leftrightarrow} HA}$$  \hspace{1cm} (S_9)

(9) From $S_6$ and $S_9$, we can also deduce the following rule:

$$HA|_{\equiv MN|_{\equiv MN \overset{SK}{\leftrightarrow} FA}}$$  \hspace{1cm} (S_{10})

(10) We apply $R_1$ and $A_8$ to $M_3$ to derive

$$FA|_{\equiv HA|_{\sim \left( < MN \overset{SK}{\leftrightarrow} FA >_{(K_{MN,ID_{MN})}}, < MN \overset{SK}{\leftrightarrow} FA >_{(KFH,FA)}, t_{HA} \right)}}$$  \hspace{1cm} (S_{11})

(11) We apply $R_2$ and $A_2$ to $S_{11}$ to derive

$$FA|_{\equiv HA|_{\equiv \left( < MN \overset{SK}{\leftrightarrow} FA >_{(K_{MN,ID_{MN})}}, < MN \overset{SK}{\leftrightarrow} FA >_{(KFH,FA)} \right)}}$$  \hspace{1cm} (S_{12})

(12) To break conjunctions, we apply the rule of BAN logic to $S_{12}$, then get

$$FA|_{\equiv HA|_{\equiv < MN \overset{SK}{\leftrightarrow} FA >_{(K_{MN,ID_{MN})}}}}$$  \hspace{1cm} (S_{13})
\[ FA \equiv HA \equiv MN \leftrightarrow FA \]  \hspace{1cm} (S_{14})

(13) We apply \(R_1, R_2,\) and \(A_8\) to \(S_{14}\) to derive

\[ FA \equiv HA \equiv MN \leftrightarrow FA \]  \hspace{1cm} (S_{15})

(14) From \(S_{10}\) and \(S_{15},\) we can imply the following statement:

\[ FA \equiv MN \equiv MN \leftrightarrow FA \]  \hspace{1cm} (S_{16})

In this step, we achieve \(G_2.\)

(15) We apply \(R_3\) and \(A_{11}\) to \(S_{15}\) to derive

\[ FA \equiv MN \leftrightarrow FA \]  \hspace{1cm} (S_{17})

In this step, we achieve \(G_4.\)

(16) We apply \(R_2, A_1,\) and \(A_{10}\) to \(M_4\) to derive

\[ MN \equiv HA \equiv \left( < MN \leftrightarrow FA >_{(K_{MJMN})}, < MN \leftrightarrow FA >_{(SK_{JFA})} \right) \]  \hspace{1cm} (S_{18})

(17) To break conjunctions, we apply the rule of BAN logic to \(S_{18},\) then get

\[ MN \equiv HA \equiv < MN \leftrightarrow FA >_{(K_{MJMN})} \]  \hspace{1cm} (S_{19})

\[ MN \equiv HA \equiv < MN \leftrightarrow FA >_{(SK_{JFA})} \]  \hspace{1cm} (S_{20})

(18) We apply \(R_1, R_2, A_1,\) and \(A_6\) to \(S_{19}\) to derive

\[ MN \equiv HA \equiv MN \leftrightarrow FA \]  \hspace{1cm} (S_{21})

(19) We apply \(R_1, R_2, A_1,\) and \(A_{12}\) to \(S_{20}\) to derive

\[ MN \equiv FA \equiv MN \leftrightarrow FA \]  \hspace{1cm} (S_{22})

In this step, we achieve \(G_1.\)

(20) We apply \(R_3\) and \(A_{10}\) to \(S_{21}\) to derive

\[ MN \equiv MN \leftrightarrow FA \]  \hspace{1cm} (S_{23})

In this step, we achieve \(G_3.\)

(21) We apply \(R_2, A_3,\) and \(A_{14}\) to \(M_5\) to derive

\[ HA \equiv MN \equiv \left( < h(K_{HIDMN}) >_{h(K_{J})}, < ID_{MN} >_{K_{M}} \right) \]  \hspace{1cm} (S_{24})

(22) To break conjunctions, we apply the rule of BAN logic to \(S_{24},\) then get

\[ HA \equiv MN \equiv < h(K_{HIDMN}) >_{h(K_{J})} \]  \hspace{1cm} (S_{25})
HA |≡ MN| ⊑ < ID_{MN} >_K_M

(23) We apply \( R_1 \) and \( A_5 \) to \( S_{25} \) to derive

\[ HA |≡ MN| ≡ h(K_H||ID_{MN}) \]  

(27) From \( S_{27} \) and \( S_{30} \), we can also deduce the following rule:

\[ HA |≡ MN| ≡ MN \xleftarrow{K_M} HA \]  

(30) We apply \( R_1, R_2, \) and \( A_6 \) to \( S_{33} \) to derive

\[ MN |≡ HA| ≡ < MN \xleftarrow{(K_M|I_{MN})} HA >_{(K_M|I_{MN})} \]  

In this step, we achieve \( G_5 \).

7. Security Analysis

The proposed scheme guarantees anonymity, hop-by-hop authentication, untraceability, resistance against password guessing attack, resistances against impersonation and forgery attacks, resistance
against known session key attack, and fair key agreement. We define two different anonymity preservations in this paper. One is weak anonymity preservation against a passive adversary who accomplishes a passive attack, like eavesdropping. The other is strong anonymity preservation against a valid but malicious node. Clearly, a malicious node is more powerful than a passive adversary because it possesses a valid sensor. If a scheme guarantees strong anonymity, it also absolutely preserves weak anonymity. The detail analysis of our scheme is described below.

7.1. Strong Anonymity

Among the transmission messages, only $MV_4$ and $HV_5$ contain $MN$’s identity $ID_{MN}$, which is formed as:

$$MV_4 = MV_1 \oplus MV_2 \oplus t_{MN} = A_{MN} \oplus h(ID_{MN}) \oplus B_{MN} \oplus h(PW_{MN}||r) \oplus t_{MN}$$

$$= h(K_H) \oplus h(K_H||ID_{MN}) \oplus t_{MN}$$

$$HV_5 = h(K_M||ID_{MN}||t_{MN})$$

An adversary who refers to a malicious sensor node can know $h(K_H)$ and $t_{MN}$. Clearly, a valid sensor node can provide $h(K_H)$, while $M_1$ can reveal $t_{MN}$. Thus, it is easy for the adversary to know $h(K_H)$ and $t_{MN}$. However, although knowing $h(K_H)$ and $t_{MN}$, there is no way to get $ID_{MN}$ from $MV_4$ and $HV_5$. This is because $ID_{MN}$ is one of the input parameters of a one-way hash function, and it is always used with the secret key $K_H$ or $K_M$. Namely, only the entity who knows $K_H$ or $K_M$ can obtain $ID_{MN}$. As a result, the proposed scheme guarantees strong anonymity against a malicious sensor node.

7.2. Hop-by-Hop Authentication

In the login and authentication phase, each entity, $MN$, $FA$, and $HA$, needs to authenticate the others. Trusting relationships between $MN$ and $HA$, and $FA$ and $HA$ make it possible for them to check each other’s identities. First, after computing $SK = h(K_M||t_{MN}||t_{FA}||ID_{MN}||ID_{FA})$, $MN$ can authenticate $HA$, by checking $HV_1 = h(SK||K_M||t_{MN})$. Since $K_M$ is only known to $MN$ and $HA$, a successful verification of $HV_1$ implies $HA$ normally computes $SK$ and $HV_1$. $MN$ can also authenticate $FA$, by checking $FV_2 = SK \oplus h(SK||t_{FA})$ with the verified $SK$. $HA$ is another entity other than $FA$ that can compute $FV_2$, but there is no reason for $HA$ to compute $FV_2$ instead of $FA$. This means that only $FA$ can compute a valid $FV_2$. Second, by verifying $HV_3 = h(KFH||HV_1||HV_2||ID_{HA}||t_{HA})$, $FA$ can authenticate $HA$, since $KFH$ is a securely pre-shared secret key between $FA$ and $HA$. In addition, $FA$ can anonymously authenticate $MN$ through $HA$. Although $FA$ has no information related to $MN$, $FA$ can authenticate $MN$, by confirming that $HA$ ensures the identification of $MN$. Lastly, $HA$ can authenticate $FA$, through verifying $FV_4 = h(KFH||MV_4||MV_5||t_{MN}||t_{FA})$. For the same reason as $FA$, $HA$ can identify $FA$, due to $KFH$. Since $MN$ is the only entity to compute $MV_5$ using $K_M$ and $ID_{MN}$, checking $MV_5 = h(K_M||ID_{MN}||t_{MN})$ makes $HA$ authenticate $MN$. As a result, the proposed scheme provides hop-by-hop authentication among $MN$, $FA$, and $HA$, while they accomplish the login and authentication phase.

7.3. Untraceability

If there are transmission messages that have the same value throughout several sessions, an adversary can trace those messages, and know all the messages that originate from one sensor node. However, since they always contain different timestamps, every transmission message in the proposed scheme is unique in each session. In addition, an adversary cannot link two or more different sessions of the same sensor node. Therefore, the proposed scheme preserves the freshness and untraceability of every message in every session.
7.4. Resistance Against Password Guessing Attack

An adversary can eavesdrop any transmission messages, but there is no way to get the sensor node’s password from those messages. The reason is that no transmission messages contain the password itself, or even related information. Even if the secret parameters and a salt stored in the sensor node are revealed, it is still impossible to obtain the password. An adversary can generate a lookup table to make pre-computed hash values with candidate passwords and salt. However, changing salts in the password change phase makes it impossible to generate pre-computed hash values. Therefore, in the proposed scheme, the possibility to verify the correctness of a guessed password does not exist, and a password guessing attack is impossible.

7.5. Resistance Against Impersonation and Forgery Attacks

If an adversary can compute $MV_5$ formed as $h(K_M || ID_{MN} || t_{MN}), he or she is able to impersonate $MN$, by sending a valid login and authentication request message. However, this is absolutely impossible, since the adversary cannot know $K_M$ and $ID_{MN}$. Meanwhile, suppose that the adversary makes the following forged message $M_1 = \{MV_4', MV_5', t_{MN}', ID_{HA}\}$, and sends it to $HA$ via $FA$:

\[
MV_4' = h(K_{H'} \oplus h(K_{adv} || ID_{adv} || t_{adv}))
\]

\[
MV_5' = h(K_{adv} || ID_{adv} || t_{adv})
\]

where $ID_{adv}$ is the identity of the adversary, $t_{adv}$ is the timestamp generated by the adversary, $K_{adv}$ is the secret key of the adversary, and $K_{H'}$ is the fake secret key of $HA$. Then, after computing $MV_4' \oplus h(K_{H'} \oplus t_{adv})$, $HA$ tries to search $h(K_{H'} || ID_{adv})$ in its database. Unfortunately, $HA$ finds no matching value, and then it recognizes the fact that the adversary sent $MV_4'$.

7.6. Resistance Against Known Session Key Attack

Even if the session key established between $MN$ and $FA$ is revealed, there is no way to compute the next session key, using the exchanged messages, $HV_1$ and $HV_2$. To compute the session key, it is necessary to know the sensor node’s secret key $K_M$ or the long-term secret key $KF_H$, which is shared between $FA$ and $HA$. Clearly, an adversary cannot compute the session key, since he or she does not know $K_M$ or $KF_H$. Moreover, since every session key contains unique timestamps, they have no relation with each other. For this reason, the proposed scheme is resistant against known session key attack.

7.7. Fair Key Agreement

When $MN$, $FA$, and $HA$ perform the login and authentication phase, the session key contains two timestamps generated by $MN$ and $FA$, respectively. This implies that $MN$ and $FA$ make the same contribution to the freshness and randomness of the session key. In other words, both $MN$ and $FA$ contribute equally during the establishment of the session key. As a result, the proposed scheme achieves fair key agreement.

8. Security and Performance Comparisons

In this section, we compare security and performance of our scheme with the previous schemes of Jiang et al., Wen et al., Shin et al., Gope and Hwang, and Farash et al. To analyze security of each scheme, we apply the following security features to them:

- SF1: Weak anonymity,
- SF2: Strong anonymity,
- SF3: Hop-by-hop authentication,
- SF4: Untraceability,
- SF5: Resistance against password guessing attack,
- SF6: Resistance against impersonation and forgery attack,
- SF7: Resistance against known session key attack,
- SF8: Fair key agreement,
- SF9: No verification table.

In addition, we apply the experiment result of Li et al. [4] to analyze performance. The following notations show the execution times of each operation:

- H: Execution time of a one-way hash function ($1H \approx 0.0005$ s),
- S: Execution time of a symmetric operation ($1S \approx 0.0087$ s),
- E: Execution time of a modular exponential operation ($1E \approx 0.522$ s),

To describe concisely, we also use the following terms:

- P1: Phase of login and authentication,
- P2: Phase of password change.

Table 2 denotes the security comparison of each scheme. Table 2 shows that our proposed scheme provides more enhanced security than previous schemes do. However, the schemes of Wen et al. and Shin et al. and our scheme need to maintain a verification table. The verification table stored in $HA$ contains information for user authentication. Namely, it contains identity/counter pairs in Wen et al.’s scheme, identity/password pairs in Shin et al.’s scheme, and identity/secret key pairs in our scheme. Looking up this information takes time. However, considering $HA$’s strong computational power, it is negligible.

| Scheme           | SF1 | SF2 | SF3 | SF4 | SF5 | SF6 | SF7 | SF8 | SF9 |
|------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Jiang et al.     | Yes | Yes | No  | Yes | No  | Yes | Yes | Yes | Yes |
| Wen et al.       | Yes | Yes | No  | Yes | No  | Yes | No  | No  | No  |
| Shin et al.      | Yes | Yes | No  | Yes | No  | Yes | Yes | No  | No  |
| Gope and Hwang   | Yes | Yes | Yes | Yes | No  | Yes | Yes | Yes | Yes |
| Farash et al.    | Yes | No  | No  | Yes | No  | Yes | Yes | Yes | Yes |
| Ours             | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No  | No  |

Meanwhile, Table 3 compares the computation cost in the login and authentication phase. Our proposed scheme, which is based only on low-cost functions, needs the lowest computation cost among all schemes. Table 4 shows the performance comparison in the password change phase. In the schemes of Jiang et al., Wen et al., and Gope and Hwang, $MN$ changes his or her password without any help of $HA$. Whereas, the schemes of Shin et al., Farash et al., and our scheme require that both $MN$ and $HA$ participate while updating $MN$’s password. Clearly, the schemes that $MN$ performs the password change phase alone have a little bit better efficiency. However, the computation cost of the password change phase in each scheme is slightly different, and thus it does not affect the performance of all over the scheme. Table 5 shows the total computation cost of each scheme. Consequently, as shown in Table 5, our proposed scheme runs the fastest and has the highest efficiency.

| Scheme           | MN   | FA   | HA   | Total           |
|------------------|------|------|------|-----------------|
| Jiang et al.     | 4H + 1E | 4H | 5H + 1E | 13H + 2E ≈ 1.0505 s |
| Wen et al.       | 4H + 1E | 4H + 1E | 5H + 2E | 13H + 4E ≈ 2.0945 s |
| Shin et al.      | 5H | 1H + 2S | 3H + 2S + 1E | 9H + 4S + 1E ≈ 0.5613 s |
| Gope and Hwang   | 6H + 1E | 3H + 1S | 7H + 1S + 1E | 16H + 2S + 2E ≈ 1.0694 s |
| Farash et al.    | 6H | 1H + 2S | 5H + 2S | 12H + 4S ≈ 0.0408 s |
| Ours             | 6H | 4H | 7H | 17H ≈ 0.0085 s |
Table 4. Performance comparison in password change phase.

| Scheme            | MN | FA  | HA  | Total       |
|------------------|----|-----|-----|-------------|
| Jiang et al.     | 2H | N/A | N/A | 2H          | ≈ 0.0010 s   |
| Wen et al.       | 2H | N/A | N/A | 2H          | ≈ 0.0010 s   |
| Shin et al.      | 4H | N/A | 1H  | 5H + 1E     | ≈ 0.5245 s   |
| Gope and Hwang   | 2H | N/A | N/A | 2H          | ≈ 0.0010 s   |
| Farash et al.    | 6H | N/A | 5H  | 11H         | ≈ 0.0055 s   |
| Ours             | 6H | N/A | 5H  | 11H         | ≈ 0.0055 s   |

Table 5. Total computation cost comparison.

| Scheme            | P1     | P2   | Total       |
|------------------|--------|------|-------------|
| Jiang et al.     | 13H + 2E | 2H   | 15H + 2E    | ≈ 1.0515 s   |
| Wen et al.       | 13H + 4E | 2H   | 15H + 4E    | ≈ 2.0955 s   |
| Shin et al.      | 9H + 4S + 1E | 5H + 1E | 14H + 4S + 2E | ≈ 1.0858 s   |
| Gope and Hwang   | 16H + 2S + 2E | 2H   | 18H + 2S + 2E | ≈ 1.0704 s   |
| Farash et al.    | 12H + 4S | 11H  | 23H + 4S    | ≈ 0.0463 s   |
| Ours             | 17H    | 11H  | 28H         | ≈ 0.0140 s   |

9. Conclusions

In this paper, we first prove that Farash et al.’s scheme fails to guarantee strong anonymity, foreign agent authentication, or password replacement. To remedy these weaknesses, we propose an enhanced security authentication scheme. The secret key for each sensor node and the password by hashing with a different salt enhance the security of our scheme. By comparing our scheme with other recent schemes, we show that it is more secure from various aspects. In addition, to reduce the computation time, our scheme only uses low-cost functions. Performance comparison shows that, as compared with the previous ones, our proposed scheme provides better lightness. This means that it provides better efficiency. Consequently, the proposed scheme is more suitable for battery-powered sensors and wireless sensor networks.

Acknowledgments: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Science, ICT, and Future Planning (2014R1A1A2002775).

Author Contributions: Y.C. conceived the main idea, designed the scheme and wrote the paper; S.C. conducted the protocol analysis and assisted the analysis of related works; Y.L. contributed to the initial idea and the motivation of conducting analysis on several attacks; N.P. analyzed the security and performance of the scheme and assisted in revising the paper; D.W. supervised the work and revised versions.

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

References

1. Zhu, J.; Ma, J. A new authentication scheme with anonymity for wireless environments. *IEEE Trans. Consum. Electron.* 2004, 50, 231–235.
2. Lee, C.C.; Hwang, M.S.; Lio, I.E. Security enhancement on a new authentication scheme with anonymity for wireless environments. *IEEE Trans. Ind. Electron.* 2006, 53, 1683–1687. [CrossRef]
3. Wu, C.C.; Lee, W.B.; Tsaur, W.J. A secure authentication scheme with anonymity for wireless communications. *IEEE Commun. Lett.* 2008, 12, 722–723.
4. Li, C.; Hwang, M.; Chung, Y. A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular ad hoc networks. *Comput. Commun.* 2008, 31, 2803–2814. [CrossRef]
5. Lee, J.S.; Chang, J.H.; Lee, D.H. Security flaw of authentication scheme with anonymity for wireless communications. *IEEE Commun. Lett.* 2009, 13, 292–293.
6. Xu, J.; Feng, D. Security flaws in authentication protocols with anonymity for wireless environments. *ETRI J.* 2009, 31, 460–462. [CrossRef]
7. Kun, L.; Anna, X.; Fei, H.; Lee, D.H. Anonymous authentication with unlinkability for wireless environments. *IEICE Electron. Express* 2011, 8, 536–541. [CrossRef]
8. Tsai, J.L.; Lo, N.W.; Wu, T.C. Secure anonymous authentication protocol with unlinkability for mobile wireless environment. In Proceedings of the IEEE International Conference on Anti-Counterfeiting, Security and Identification, Taipei, Taiwan, 24–26 August 2012; pp. 1–5.
9. Mun, H.; Han, K.; Lee, Y.S.; Yeun, C.Y.; Choi, H.H. Enhanced secure anonymous authentication scheme for roaming service in global mobility networks. *Math. Comput. Model.* 2012, 55, 214–222. [CrossRef]
10. Zhao, D.; Peng, H.; Li, L.; Yang, Y. A secure and effective anonymous authentication scheme for roaming service in global mobility networks. *Wirel. Pers. Commun.* 2013, 78, 247–267. [CrossRef]
11. Jeon, W.; Kim, J.; Nam, J.; Lee, Y.; Won, D. An enhanced secure authentication scheme with anonymity for wireless environments. *IEICE Trans. Commun.* 2012, 95, 2505–2508. [CrossRef]
12. Nam, J.; Choo, K.K.; Han, S.; Kim, M.; Paik, J.; Won, D. Efficient and anonymous two-factor user authentication in wireless sensor networks: Achieving user anonymity with lightweight sensor computation. *PLoS ONE* 2015, 10, e0116709. [CrossRef] [PubMed]
13. Zhou, T.; Xu, J. Provable secure authentication protocol with anonymity for roaming service in global mobility networks. *Comput. Netw.* 2011, 55, 205–213. [CrossRef]
14. Jiang, Q.; Ma, J.; Li, G.; Yang, L. An enhanced authentication scheme with privacy preservation for roaming service in global mobility networks. *Wirel. Pers. Commun.* 2013, 68, 1477–1491. [CrossRef]
15. Wen, F.; Susilo, W.; Yang, G. A secure and effective anonymous authentication scheme for roaming service in global mobility networks. *Wirel. Pers. Commun.* 2013, 73, 993–1004.
16. Shin, S.; Yeh, H.; Kim, K. An efficient secure authentication scheme with user anonymity for roaming user in ubiquitous networks. *Peer-to-Peer Netw. Appl.* 2013, 8, 674–683. [CrossRef]
17. Gope, P.; Hwang, T. Enhanced secure mutual authentication and key agreement scheme preserving user anonymity in global mobile networks. *Wirel. Pers. Commun.* 2015, 82, 2231–2245. [CrossRef]
18. Farash, M.S.; Chaudhry, S.A.; Heydari, M.; Sadough, S.M.S.; Kumari, S.; Khan, M.K. A lightweight anonymous authentication scheme for consumer roaming in ubiquitous networks with provable security. *Int. J. Commun. Syst.* 2015, 28. [CrossRef]
19. Chang, C.C.; Lee, C.Y.; Chiu, Y.C. Enhanced authentication scheme with anonymity for roaming service in global mobility networks. *Comput. Commun.* 2009, 32, 611–618. [CrossRef]
20. Youn, T.Y.; Park, Y.H.; Lim, J. Weaknesses in an anonymous authentication scheme for roaming service in global mobility networks. *IEEE Commun. Lett.* 2009, 13, 471–473. [CrossRef]
21. He, D.; Chan, S.; Chen, C.; Bu, J.; Fan, R. Design and validation of an efficient authentication scheme with anonymity for roaming service in global mobility networks. *Wirel. Pers. Commun.* 2011, 61, 465–476. [CrossRef]
22. Choi, Y.; Nam, J.; Lee, D.; Kim, J.; Jung, J.; Won, D. Security enhanced anonymous multi-server authenticated key agreement scheme using smart cards and biometrics. *Sci. World J.* 2014, 2014, 281305. [CrossRef] [PubMed]
23. Kim, J.; Lee, D.; Jeon, W.; Lee, Y.; Won, D. Security analysis and improvements of two-factor mutual authentication with key agreement in wireless sensor networks. *Sensors* 2014, 14, 6443–6462. [CrossRef] [PubMed]
24. Gope, P.; Hwang, T. Lightweight and energy-efficient mutual authentication and key agreement scheme with user anonymity for secure communication in global mobility networks. *IEEE Syst. J.* 2015. [CrossRef]
25. Moon, J.; Choi, Y.; Kim, J.; Won, D. An Improvement of robust and efficient biometrics based password authentication scheme for telecare medicine information systems using extended chaotic maps. *J. Med. Syst.* 2016, 40, 1–11. [CrossRef] [PubMed]
26. Burrows, M.; Abadi, M.; Needham, R. A logic of authentication. *ACM Trans. Comput. Syst.* 1990, 8, 18–36. [CrossRef]