Cryptanalysis of Improved and Provably Secure Three-Factor User Authentication Scheme for Wireless Sensor Networks

Jihyeon Ryu[1], Taeui Song[1], Jongho Moon[2], Hyoungshick Kim[3], and Dongho Won[3],*

[1]Department of Platform Software, Sungkyunkwan University
{jhryu, tusong}@security.re.kr
[2]Department of Electrical and Computer Engineering, Sungkyunkwan University
jhmoon@security.re.kr
[3]Department of Computer Engineering, Sungkyunkwan University
hyoung@skku.edu(H.K.); dhwon@security.re.kr(D.H.)

Abstract. Wireless sensor networks are applied in various areas like smart grid, environmental monitoring, health care, and security and surveillance. It applies to many fields, but as the utilization is higher, security becomes more important. Recently, the authentication scheme for the environment of wireless sensor network has also been studied. Wu et al. has announced a three-factor user authentication scheme claiming to be resistant to different types of attacks and maintain various security attributes. However, their proposal has several fatal vulnerabilities. First, it is vulnerable to the outsider attack. Second, it is exposed to user impersonation attack. Third, it does not satisfy user anonymity. Therefore, in this paper, we describe these vulnerabilities and prove Wu et al.'s scheme is unsafe.

Keywords: Wireless Sensor Network · Elliptic Curve Cryptosystem · Remote user authentication · Biometric

1 Introduction

A distributed network of autonomous sensors that can collect information related to environmental or physical conditions is called wireless sensor network(WSN). Thanks to its easiness and inexpensive deployment capabilities, WSN is applicable to numerous scientific and technological areas: Environmental monitoring, a smart grid, health care, security and surveillance, an earthquake, fire and other human activities and physical and environmental phenomena. For these reasons, a security of WSN is as important as its variety of applications. In particular, if user's personal information is contained, it should not be exposed to others.

* Corresponding author.

R. Alfred et al. (eds.), *Computational Science and Technology*, Lecture Notes in Electrical Engineering 481, https://doi.org/10.1007/978-981-13-2622-6_5

WSN systems consist of three entities: a user interface, sensor nodes that measure physical or environmental conditions, and gateway nodes that forward information received from sensor nodes to a central server. WSN should provide simplicity and efficiency to users and must also be secure. Even if intercepting data packets sent from the WSN, an unauthorized user should not know any private information, such as the user's identity. Furthermore, any user should not be able to be authenticated as another user. However, the problem we found is that these conditions do not hold in Wu et al.'s scheme [1].

1.1 Related Work

In 2004, Watro et al. [2] suggested a user authentication scheme using the RSA and Diffie-Hellman key exchange algorithm. In 2009, the first two-factor user authentication scheme for WSNs was introduced by Das [3]. In their scheme, to pass a gateway node's checking steps, a legitimate user should have not only a password but also a smart card. This mechanism had been applied for many years in client/server networks [4–7]. However, He et al. [8] have discovered that Das' scheme was susceptible to several attacks such as an insider attack, impersonation attack, and it had lack of mutual authentication. For these reasons, they proposed the improved scheme. Unfortunately, Kumar et al. [9] mentioned that there were several vulnerabilities such as information leakage, no session key agreement, no user anonymity, and no mutual authentication in the scheme [8] In 2011, Yeh et al. [10] suggested the first two-factor user authentication scheme for WSNs using elliptic curve cryptosystem. In addition, In 2013, Xue et al. [11] proposed a temporal-credential-based authentication scheme for WSNs. In fact, a temporal credential is a result from hashing the shared key between the user and the gateway, the user's identity, and the expiration time of the temporal credential. However, it is proved by Jiang et al. [12] that the scheme [11] was insecure to the identity guessing attack, insider and tracking attacks, and off-line password guessing attack. As a result, they proposed a new mechanism in the scheme [12].

In 2014, Das [13] explained that there are some significant problems in Jiang et al.'s two-factor user authentication method [12], such as vulnerability of insider attack, lack of no formal security verification, and de-synchronization attacks, so they suggested a new three-factor user authentication scheme. In 2015, Das also introduced two three-factor authentication schemes in [14, 15], individually. In 2018, however, Wu et al. [1] found that Das' schemes [13–15] are still vulnerable. The scheme [13] was susceptible to off-line password guessing and de-synchronization attacks, and schemes [14, 15] could not withstand the off-line password guessing, user impersonation attacks. Wu et al. [1] designed an improved user authentication scheme using elliptic curve cryptography(ECC) which has been applied for WSN recently.

Unfortunately, we have found that Wu et al. [1]'s scheme is still unreliable. To be specific, Wu et al.'s scheme is exposed to the outsider, user impersonation attacks and do not satisfy user anonymity.

1.2 Organization of our paper

The rest of the paper is summarized as follows. In Section 2, we provide some preliminary knowledge such as ECC, fuzzy extractor and threat model. In addition, we review Wu et al.'s scheme of [1] in Section 3. In Section 4, we specify some vulnerabilities in Wu et al.'s scheme [1]. At last, the conclusion is shown in Section 5.

2 Preliminary Knowledge

This section describes the basic backgrounds of the elliptic curves and contents of the fuzzy extractor which are used in Wu et al.'s scheme [1] and threat model.

2.1 Elliptic Curve Cryptosystem

Elliptic curve cryptosystem(ECC) is the most frequently used password in modern passwords and has strong security characteristics. The elliptic curve cryptosystem created by Victor Miller [16] and Neal Kobiltz [17] in 1985 and 1987. It has the following form:

$$y^2 = x^3 + ax + b \quad mod\ p \qquad a, b \in F_p \tag{1}$$

Equation 1 is an equation of elliptic curve cryptosystem on the field F_p. The following conditions must be met to ensure safety.

$$4a^3 + 27b^2 \neq 0 \quad mod\ p \tag{2}$$

Equation 2 guarantees non-singular of an elliptic curve. In other words, using this elliptic curve equation 2, the following safety is guaranteed. We assume that P is the point on the elliptic curve, xP is the computation of P times x, yP is the computation of P times y, and xyP is the computation of P times xy.

1. *Elliptic Curve Decisional Diffie-Hellman Problem:* Given xP, yP it is impossible to find xyP.
2. *Elliptic Curve Computational Diffie-Hellman Problem:* Given xyP, it is impossible to find xP, yP.
3. *Elliptic Curve Discrete Logarithm Problem:* Given P, xP it is impossible to find x.

2.2 Fuzzy Extractor

User's biometric information is very important and sensitive information. In general, human biometrics can be perceived as a different result. The fuzzy extractor retrieves everybody's biometrics with a random arbitrary bit stream. User can get owns a secret string using error tolerance through the fuzzy extractor. Based on Refs [18, 19], the fuzzy extractor is worked through two processes (*Gen*, *Rep*) as follows:

$$Gen(B) \rightarrow \langle \alpha, \beta \rangle \tag{3}$$

$$Rep(B^{*}, \beta) = \alpha \text{ if } BIO^{*} \text{ is reasonably close to BIO} \tag{4}$$

From above equations, *Gen* is a probabilistic generation function using biometrics B , and extracts string $\alpha \in \{0,1\}^k$ and auxiliary string $\beta \in \{0,1\}^*$. On the other hand, *Rep* is a deterministic reproduction function that recovers α from β and any vector BIO^* that is reasonably close to BIO. For further details of the fuzzy extractor, see [20].

2.3 Threat Model

In this subsection, we describe some threat model [21] and consider constructing the assumptions of the threat model are shown as follows:

1. The attacker $\mathcal{A}$ could be either a user, sensor, or gateway. Any certified user can act as an attacker.
2. $\mathcal{A}$ could intercept or snoop all communication messages in a public channel so that $\mathcal{A}$ could steal any messages communicated between a user and sensor or gateway.
3. $\mathcal{A}$ has the capability of modifying, rerouting or deleting the intercepted message.
4. Using a side channel attack, stored parameters can be drawn from the smart card.

3 Review of Wu et al.'s scheme

In this section, we review Wu et al.'s scheme [1] to do the cryptanalysis on their scheme. The scheme consists of four phases as follows: registration phase, login phase, authentication phase, and password change phase. As schemes in [19], the scheme employs the ECC. GWN, first, produces G on $E(F_p)$ using a generator P and a large prime order n. GWN, then, chooses a private key x of which length is the security length l_s and two cryptographic hash functions $h(\cdot)$ and $h_1(\cdot)$. They are considered that the all the random generated numbers should reach the length l_s. The notations used in Wu et al.'s scheme are written in Table 1.

3.1 Registration Phase

This phase consists of two parts: user registration and sensor registration.

Table 1. Notations used in Wu et al.'s scheme.

Notations	Description
U_i	The i-th user
S_j, SID_j	The j-th sensor and its identity
ID_i	U_i's identity
PW_i	U_i's Password
B_i	U_i's biometric information
$\mathcal{A}$	The malicious attacker
x	Private key of GWN
r_i	U_i's randomly generated number
$h(\cdot)$, $h_1(\cdot)$	One-way hash function
$X\|\|Y$	Concatenation operation
$\oplus$	Bitwise XOR operation
$E(F_p)$	A collection of points on an elliptic curve over a finite field F_p
P	A point generator in F_p with a large prime order n
G	A cyclic addition group with point generator P
sk_u, sk_s	The session key generated by U_i and S_j respectively.
l_s	Security length variable

User registration

1. An user U_i, first, decides his/her identity ID_i and password PW_i with a randomly generated number r_i, imprints B_i over a device for biometrics collection, and calculates $Gen(B_i) = (R_i, P_{bi})$, $DID_i = h(ID_i \parallel r_i)$ and $HPW_i = h(PW_i \parallel r_i \parallel R_i)$. He/she, then, transmits the registration request $\{ID_i, DID_i\}$ to the gateway node GWN in the secure channel.
2. After obtaining the registration request from the U_i, GWN computes $B_1' = h(DID_i \parallel x)$ where the value x is a secret key of GWN, produces a smart card for U_i holding $h(\cdot)$, $h_1(\cdot)$, P, and stores ID_i in its database. GWN then delivers the smart card with B_1' to the U_i secretly.
3. After taking the smart card with B_1' from the GWN, U_i computes $B_1 = B_1' \oplus HPW_i$ and $B_2 = h(ID_i \parallel R_i \parallel PW_i) \oplus r_i$ with storing B_1, B_2, P and P_{bi} into the smart card.

Sensor registration

1. GWN picks an identity $SIDj$ for each new sensor node S_j, calculates $c_j = h(SID_j \parallel x)$, and sends $\{SID_j, c_j\}$ to S_j.
2. S_j stores P, SID_j and c_j, and follows the WSN.

3.2 Login Phase

1. U_i enters ID_i, PW_i and B_i'. The smart card generates $Rep(B_i', P_{bi}) = R_i$, $r_i = B_2 \oplus h(ID_i \parallel R_i \parallel PW_i)$, $HPW_i = h(PW_i \parallel r_i \parallel R_i)$ and $DID_i = h(ID_i \parallel r_i)$.

2. The smart card produces randomly generated numbers r_i^{new}, e_i and $\alpha \in [1, n-1]$, and chooses a special sensor SID_j. The smart card then computes $DID_i^{new} = h(ID_i \parallel r_i^{new})$, $C_1 = B_1 \oplus HPW_i \oplus e_i$, $C_2 = \alpha P$, $C_3 = h(e_i) \oplus DID_i^{new}$, $Z_i = ID_i \oplus h(e_i \parallel DID_i)$ and $C_4 = h(ID_i \parallel e_i \parallel DID_i \parallel DID_i^{new} \parallel C_2 \parallel SID_j)$. The value C_4 is used for checking the identities' integrity and the user side's new data and verifying the source of the message M_1.
3. U_i sends the login request messages $M_1 = \{C_1, C_2, C_3, C_4, Z_i, DID_i, SID_j\}$ to GWN.

3.3 Authentication Phase

1. After accepting the login request messages M_1 from the user U_i, GWN first computes $e_i = C_1 \oplus h(DID_i \parallel x)$, $DID_i^{new} = C_3 \oplus h(e_i)$ and $ID_i = Z_i \oplus h(e_i \parallel DID_i)$, and checks the validity of ID_i and $C_4 \stackrel{?}{=} h(ID_i \parallel e_i \parallel DID_i \parallel DID_i^{new} \parallel C_2 \parallel SID_j)$. If either fails, GWN terminates the session. If authentication attempts fail three times in a row in a defined time span, GWN will freeze the U_i's account; otherwise, GWN calculates $c_j = h(SID_j \parallel x)$ and $C_5 = h(c_j \parallel DID_j \parallel SID_j \parallel C_2)$ and sends $M_2 = \{C_2, C_5, DID_i\}$ to the sensor node S_j. The value C_5 is used for checking the integrity of the strings including c_j and the data that can make the sensor S_j to obtain the correct data for computing the session key. In addition, C_5 is used for verifying the source of M_2.
2. S_j checks $C_5 \stackrel{?}{=} h(c_j \parallel DID_i \parallel SID_j \parallel C_2)$ with its identity SID_j. If this does not hold, S_j will disconnect the session. S_j, then, selects $\beta \in [1, n-1]$, and computes $C_6 = \beta P$, $sk_s = \beta C_2$, $C_7 = h_1(C_2 \parallel C_6 \parallel sk_s \parallel DID_i \parallel SID_j)$ and $C_8 = h(DID_i \parallel SID_j \parallel c_j)$. The major role of C_7 is to check the session key's integrity and C_6's integrity, which is the part used by U_i to compute the session key. Furthermore, both C_7 and C_8 are used to verifying the source of M_3. In the end, S_j transmits $M_3 = \{C_6, C_7, C_8\}$ to GWN.
3. GWN checks $C_8 \stackrel{?}{=} h(DID_i \parallel SID_j \parallel c_j)$. If this does not satisfy, GWN disconnect the session; otherwise, GWN computes $C_9 = h(DID_i^{new} \parallel x) \oplus h(DID_i \parallel e_i)$ and $C_{10} = h(ID_i \parallel SID_j \parallel DID_i \parallel DID_i^{new} \parallel e_i \parallel C_9)$. The value C_{10} is to verify the source of the message M_4. Finally, GWN sends the message $M_4 = \{C_6, C_7, C_9, C_{10}\}$ to U_i.
4. U_i checks $C_{10} \stackrel{?}{=} h(ID_i \parallel SID_j \parallel DID_i \parallel DID_i^{new} \parallel e_i \parallel C_9)$. U_i then computes the session key $sk_u = \alpha C_6$, and checks $C_7 \stackrel{?}{=} h_1(C_2 \parallel C_6 \parallel sk_u \parallel DID_i \parallel SID_j)$. If this does not satisfy, U_i terminates the session. After that, U_i calculate $HPW_i^{new} = h(PW_i \parallel r_i^{new} \parallel R_i)$, $B_1^{new} = C_9 \oplus h(DID_i \parallel e_i) \oplus HPW_i^{new}$ and $B_2^{new} = h(ID_i \parallel R_i \parallel PW_i) \oplus r_i^{new}$, and replaces (B_1, B_2) with (B_1^{new}, B_2^{new}) in the smart card individually.

3.4 Password and Biometrics Change Phase

1. This step is same as the first step of Login phase.

2. The smart card produces random generated numbers r_i^{new} and e_i, calculates DID_i^{new}, C_1, C_3, Z_i and $C_{11} = h(ID_i \parallel e_i \parallel DID_i \parallel DID_i^{new})$, and sends $M_5 = \{C_1, C_3, Z_i, C_{11}, DID_i\}$ with a password change request to GWN. The value C_{11} is similar to C_4 and it is used for checking the integrity of the identities and verifying the source of M_5.
3. GWN acquires e_i, ID_i and DID_i^{new} as first step of the authentication phase, and determines ID_i and $C_{11} \stackrel{?}{=} h(ID_i \parallel e_i \parallel DID_i \parallel DID_i^{new})$. If this does not satisfy, GWN disconnects the session; otherwise, GWN generates $C_9 = h(DID_i^{new} \parallel x) \oplus h(DID_i \parallel e_i)$ and $C_{12} = h(ID_i \parallel DID_i \parallel DID_i^{new} \parallel e_i \parallel C_9)$ and sends $M_6 = \{C_9, C_{12}\}$ and a grant to U_i. Here C_{12} is to verify the source of M_6.
4. U_i checks $C_{12} \stackrel{?}{=} h(ID_i \parallel DID_i \parallel DID_i^{new} \parallel e_i \parallel C_9)$. If it is incorrect, U_i disconnects this session; otherwise, U_i inputs a new password PW_i^{new} and a new biometric information B_i^{new}. The smart card then computes $Gen(B_i^{new}) = (R_i^{new}, P_{bi}^{new})$, $HPW_i^{new2} = h(PW_i^{new} \parallel r_i^{new} \parallel R_i^{new})$, $B1^{new2} = C_9 \oplus h(DID_i \parallel e_i) \oplus HPW_i^{new2}$ and $B_2^{new2} = h(ID_i \parallel R_i^{new} \parallel PW_i^{new}) \oplus r_i^{new}$. Finally, U_i substitutes $(B_1^{new2}, B_2^{new2}, P_{bi}^{new2})$ for (B_1, B_2, P_{bi}) in the smart card individually.

4 Security Weaknesses of Wu et al.'s scheme

In this section, we prove that Wu et al.'s scheme [1] has some security exposure. The following issues have been found and their specific descriptions are given below.

4.1 Outsider Attack

1. An attacker $\mathcal{A}$ who is the legitimate user and owns a his/her own smart card can extract the $\{B_{1\mathcal{A}}, B_{2\mathcal{A}}, P, P_{b\mathcal{A}}\}$ from his/her smart card.
2. $\mathcal{A}$ can thus get $h(DID_{\mathcal{A}} \parallel x) = B_{1\mathcal{A}} \oplus HPW_{\mathcal{A}}$, and use this value for other attacks. Because, this value is an important value that identifies the user on the gateway node side. $h(DID_A \parallel x)$ will be used in Section 4.2 and Section 4.3.

4.2 User Impersonation Attack

An attacker $\mathcal{A}$ can pretend any user using his/her information and other user's identity alone. We assume that the victim is user U_i at this time. The specific method is shown as follows in detailed.

1. The attacker $\mathcal{A}$ selects any identity ID_i.
2. $\mathcal{A}$ generates random numbers $r_{\mathcal{A}}^{new}$, $e_{\mathcal{A}}$, and $\alpha_{\mathcal{A}} \in [1, n-1]$, and chooses a special sensor SID_j. $\mathcal{A}$ then computes $DID_{\mathcal{A}}^{new} = h(ID_{\mathcal{A}} \parallel r_{\mathcal{A}}^{new})$, $C_{1\mathcal{A}} = B_{1\mathcal{A}} \oplus HPW_{\mathcal{A}} \oplus e_{\mathcal{A}}$, $C_{2\mathcal{A}} = \alpha_{\mathcal{A}} P$, $C_{3\mathcal{A}} = h(e_{\mathcal{A}}) \oplus DID_{\mathcal{A}}^{new}$, $Z_{\mathcal{A}} = ID_i \oplus h(e_{\mathcal{A}} \parallel DID_{\mathcal{A}})$ and $C_{4\mathcal{A}} = h(ID_i \parallel e_{\mathcal{A}} \parallel DID_{\mathcal{A}} \parallel DID_{\mathcal{A}}^{new} \parallel C_{2\mathcal{A}} \parallel SID_j)$. $C_{4\mathcal{A}}$ is used for checking the integrity of the identities and the new data produced on the user side and verifying the source of $M_{1\mathcal{A}}$.

3. A transmits the login request $M_{1\mathcal{A}} = \{C_{1\mathcal{A}}, C_{2\mathcal{A}}, C_{3\mathcal{A}}, C_{4\mathcal{A}}, Z_{\mathcal{A}}, DID_{\mathcal{A}}, SID_j\}$ to the gateway node GWN.
4. After obtaining the login request from the $\mathcal{A}$, GWN, first, calculates $e_{\mathcal{A}} = C_{1\mathcal{A}} \oplus h(DID_{\mathcal{A}} \parallel x)$, $DID_{\mathcal{A}}^{new} = C_{3\mathcal{A}} \oplus h(e_{\mathcal{A}})$ and $ID_i = Z_{\mathcal{A}} \oplus h(e_{\mathcal{A}} \parallel DID_{\mathcal{A}})$, and checks the validity of ID_i and $C_{4\mathcal{A}} \stackrel{?}{=} h(ID_i \parallel e_{\mathcal{A}} \parallel DID_{\mathcal{A}} \parallel DID_{\mathcal{A}}^{new} \parallel C_{2\mathcal{A}} \parallel SID_j)$. GWN proceeds the scheme without any detection. Unfortunately, the GWN misunderstand that he/she is communicating with the valid victim U_i.

As a result, the attacker A will be verified as user U_i by user GWN. Therefore, the user impersonation attack is succeed.

4.3 No User Anonymity

The attacker $\mathcal{A}$ can extract the identity of U_i from the login request message M_i of U_i. Assume that $\mathcal{A}$ eavesdrops the login request message $M_1 = \{C_1, C_2, C_3, C_4, Z_i, DID_i, SID_j\}$ of U_i. The details are as follows.

1. The attacker $\mathcal{A}$ first generates randomly generated numbers $r_{\mathcal{A}}^{new}$, $e_{\mathcal{A}}$, and $\alpha_{\mathcal{A}} \in [1, n-1]$, and chooses a special sensor SID_j. $C_{1\mathcal{A}} = B_{1\mathcal{A}} \oplus HPW_{\mathcal{A}} \oplus e_{\mathcal{A}}$, $C_{2\mathcal{A}} = \alpha_{\mathcal{A}} P$, $C_{3\mathcal{A}} = h(e_{\mathcal{A}}) \oplus DID_i$, $Z_{\mathcal{A}} = ID_{\mathcal{A}} \oplus h(e_{\mathcal{A}} \parallel DID_{\mathcal{A}})$ and $C_{4\mathcal{A}} = h(ID_{\mathcal{A}} \parallel e_{\mathcal{A}} \parallel DID_{\mathcal{A}} \parallel DID_i \parallel C_{2\mathcal{A}} \parallel SID_j)$.
2. A sends the login request message $M_{1\mathcal{A}} = \{C_{1\mathcal{A}}, C_{2\mathcal{A}}, C_{3\mathcal{A}}, C_{4\mathcal{A}}, Z_{\mathcal{A}}, DID_{\mathcal{A}}, SID_j\}$ to the gateway node GWN.
3. After getting the login request message from the $\mathcal{A}$, GWN calculates $e_{\mathcal{A}} = C_{1\mathcal{A}} \oplus h(DID_{\mathcal{A}} \parallel x)$, $DID_i = C_{3\mathcal{A}} \oplus h(e_{\mathcal{A}})$ and $ID_{\mathcal{A}} = Z_{\mathcal{A}} \oplus h(e_{\mathcal{A}} \parallel DID_{\mathcal{A}})$, and checks the validity of $ID_{\mathcal{A}}$ and $C_{4\mathcal{A}} \stackrel{?}{=} h(ID_{\mathcal{A}} \parallel e_{\mathcal{A}} \parallel DID_{\mathcal{A}} \parallel DID_i \parallel C_{2\mathcal{A}} \parallel SID_j)$. GWN then computes $c_j = h(SID_j \parallel x)$ and $C_{5\mathcal{A}} = h(c_j \parallel DID_j \parallel SID_j \parallel C_{2\mathcal{A}})$ and sends $M_{2\mathcal{A}} = \{C_{2\mathcal{A}}, C_{5\mathcal{A}}, DID_{\mathcal{A}}\}$ to the sensor node S_j.
4. S_j checks $C_{5\mathcal{A}} \stackrel{?}{=} h(c_j \parallel DID_{\mathcal{A}} \parallel SID_j \parallel C_{2\mathcal{A}})$ with its identity SID_j. If it is incorrect, S_j terminates the session. S_j then selects $\beta_{\mathcal{A}} \in [1, n-1]$ and computes $C_{6\mathcal{A}} = \beta_{\mathcal{A}} P$, $sk_s = \beta_{\mathcal{A}} C_{2\mathcal{A}}$, $C_{7\mathcal{A}} = h_1(C_{2\mathcal{A}} \parallel C_{6\mathcal{A}} \parallel sk_s \parallel DID_{\mathcal{A}} \parallel SID_j)$ and $C_{8\mathcal{A}} = h(DID_{\mathcal{A}} \parallel SID_j \parallel c_j)$. S_j sends $M_{3\mathcal{A}} = \{C_{6\mathcal{A}}, C_{7\mathcal{A}}, C_{8\mathcal{A}}\}$ to GWN.
5. GWN checks $C_{8\mathcal{A}} \stackrel{?}{=} h(DID_{\mathcal{A}} \parallel SID_j \parallel c_j)$. If this does not hold, GWN terminates the session; otherwise, GWN calculates $C_{9\mathcal{A}} = h(DID_i \parallel x) \oplus h(DID_{\mathcal{A}} \parallel e_{\mathcal{A}})$ and $C_{10\mathcal{A}} = h(ID_{\mathcal{A}} \parallel SID_j \parallel DID_{\mathcal{A}} \parallel DID_i \parallel e_{\mathcal{A}} \parallel C_{9\mathcal{A}})$. Finally GWN sends the message $M_{4\mathcal{A}} = \{C_{6\mathcal{A}}, C_{7\mathcal{A}}, C_{9\mathcal{A}}, C_{10\mathcal{A}}\}$ to attacker $\mathcal{A}$.
6. $\mathcal{A}$ computes $h(DID_i \parallel x) = h(DID_{\mathcal{A}} \parallel e_{\mathcal{A}}) \oplus C_{9\mathcal{A}}$. Now $\mathcal{A}$ can compute $e_i = C_1 \oplus h(DID_i \parallel x)$. Finally, $\mathcal{A}$ can find $ID_i = h(e_i \parallel DID_i) \oplus Z_i$.

As a result, this result shows that Wu et al.'s scheme does not satisfy user anonymity.

5 Conclusions

In this paper, we reviewed Wu et al.'s three-factor user authentication scheme for WSN and demonstrated that outsider attack is still possible in Wu et al.'s scheme. The outsider attack could be used to pull out security-critical information. As a result, It brings about exposure of session key, user impersonation attack and no user anonymity. For these reasons, it is not secure to use their authentication scheme. Especially, ID must not exposed as an XOR to prevent user impersonation attack. Future research will need to be done in a way that will complement it. Finally, our further research would be focused on proposing an advanced user authentication scheme which can handle with these problems.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (NRF-2010-0020210)

References

1. Wu, F., Xu, L., Kumari, S., Li, X.: An Improved and Provably Secure Three-Factor User Authentication Scheme for Wireless Sensor Networks. Peer-to-Peer Networking and Applications **11**(1), 1–20 (2018)
2. Watro, R., Kong, D., Cuti, Sf., Gardiner, C., Lynn, C., Kruus, P.: Tinypk: Securing Sensor Networks with Public Key Technology. In: Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks. ACM, 59–64 (2004)
3. Das, M.: Two-Factor User Authentication in Wireless Sensor Networks. IEEE transactions on wireless communications **8**(3), 1086-1090 (2009)
4. Choi, Y., Lee, Y., Won, D.: Security Improvement on Biometric Based Authentication Scheme for Wireless Sensor Networks Using Fuzzy Extraction. International Journal of Distributed Sensor Networks **2016**, 1-16 (2016)
5. Kim, J., Moon, J., Jung, J., Won, D.: Security Analysis and Improvements of Session Key Establishment for Clustered Sensor Networks. Journal of Sensors **2016**, 1-17 (2016)
6. Kang, D., Jung, J., Mun, J., Lee, D., Choi, Y., Won, D.: Efficient and Robust User Authentication Scheme that Achieve User Anonymity with a Markov Chain. Security and Communication Networks **9**(11), 1462-1476 (2016)
7. Jung, J., Kim, J., Choi, Y., Won, D.: An Anonymous User Authentication and Key Agreement Scheme Based on a Symmetric Cryptosystem in Wireless Sensor Networks. Sensors **16**(8), 1-30 (2016)
8. He, D., Gao, Y., Chan, S., Chen, C., Bu, J.: An enhanced two-factor user authentication scheme in wireless sensor networks. Ad hoc and Sensor wireless network **10**(4), 361-371 (2010)
9. Kumar, P., Lee, H. J.: Cryptanalysis on two user authentication protocols using smart card for wireless sensor networks. IEEE Wireless advanced (WiAd), 241-245 (2011)

10. Yeh, H. L., Chen, T. H., Liu, P. C., Kim, T. H., Wei, H. W.: A secured authentication protocol for wireless sensor networks using elliptic curves cryptography. Sensors, **11**(5), 4767-4779 (2011)
11. Xue, K., Ma, C., Hong, P., Ding, R.: A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks. Journal of Network and Computer Applications, **36**(1), 316-323 (2013)
12. Jiang, Q., Ma, J., Lu, X., Tian, Y.: An efficient two-factor user authentication scheme with unlinkability for wireless sensor networks. Peer-to-peer Networking and Applications, **8**(6), 1070-1081 (2015)
13. Das, A. K.: A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks. Peer-to-peer Networking and Applications, **9**(1), 223-244 (2016)
14. Das, A. K.: A secure and effective biometricbased user authentication scheme for wireless sensor networks using smart card and fuzzy extractor. International Journal of Communication Systems, **30**(1) (2017)
15. Das, A. K.: A secure and efficient user anonymity-preserving three-factor authentication protocol for large-scale distributed wireless sensor networks. Wireless Personal Communications, **82**(3), 1377-1404 (2015)
16. Miller, V.: Uses of Elliptic Curves in Cryptography. In: Advances in Cryptology Crypto **218**, 417–426 (1986)
17. Koblitz, N.: Elliptic curve cryptosystems. Mathematics of computation **48**, 203–209 (1987)
18. Dodis, Y., Kanukurthi, B., Katz, J., Smith, A.: Robust fuzzy extractors and authenticated key agreement from close secrets. IEEE Transactions on Information Theory **58**, 6207-6222 (2013)
19. Das, A.: A Secure and Effective Biometric-based User Authentication Scheme for Wireless Sensor Networks using Smart Card and Fuzzy Extractor. International Journal of Communication Systems **2015**, 1-25 (2015)
20. Dodis, Y., Reyzin, L., Smith, A.: Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data. In: Proceedings on the International Conference on the Theory and Applications of Cryptographic Techniques, 523–540 (2004)
21. Moon, J., Choi, Y., Jung, J., Won, D.: An Improvement of Robust Biometrics-based Authentication and Key Agreement Scheme for Multi-Server Environments using Smart Cards. PLoS One **10**, 1–15 (2015)