Multimedia Systems (2017) 23:195–205
DOI 10.1007/s00530-015-0476-3

REGULAR PAPER

An improved and anonymous two-factor authentication protocol for health-care applications with wireless medical sensor networks

Fan Wu[1] · Lili Xu[2] · Saru Kumari[3] · Xiong Li[4,5]

Received: 27 March 2015 / Accepted: 20 July 2015 / Published online: 8 August 2015

Abstract Wireless sensor networks (WSNs) are fast developed and widely used in many applications. One of the most important applications is wireless medical sensor network (WMSN) which makes modern health-care more popular. The doctor can get the patient's physiological data collected by special sensors deployed on or in the patient's body in real time with the mobile devices via the wireless communication channel. The collected data are important and should be confidential. So security measures are considered in the process of communication. Recently, He et al. (Multimed Syst, 21(1), 49–60, 2015) proposed a new two-factor authentication scheme for health-care with WMSNs and claimed it to be secure. But we find that it is vulnerable to the off-line guessing attack, the user impersonation attack, and the sensor node capture attack. Moreover, we present an improved scheme to overcome the disadvantages. Through the formal verification with Proverif and the analysis presented by us, our scheme is secure. It is more practical for applications through the comparison between some recent schemes for WMSNs.

Communicated by L. Zhou.

✉ Fan Wu
conjurer1981@gmail.com

1 Department of Computer Science and Engineering, Xiamen Institute of Technology, Xiamen 361021, China

2 School of Information Science and Technology, Xiamen University, Xiamen 361005, China

3 Department of Mathematics, Ch. Charan Singh University, Meerut 250005, Uttar Pradesh, India

4 School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan 411201, China

5 Nanjing University of Information Science and Technology, Nanjing 210044, China

Keywords Wireless medical sensor network · Smart card · The off-line guessing attack · The sensor capture attack · Mutual authentication

1 Introduction

Wireless communication is now a popular issue. Many applications have appeared in recent years. Wireless sensor network is one of the most important way. WSNs include three participants: the sensors, the gateway and the users. Wireless sensors can be deployed in many fields, such as military monitoring, wildlife tracing and so forth [6, 7]. WMSN is a concrete application in medical care. Medical sensors on or in the patient's body are used for collecting optical, thermal, or other physiological signals. Doctors or health professionals can make corresponding treatments according to the real-time collected data. Researchers have discussed it [8, 27]. Usually, the gateway and the users have enough abilities for storage and computations, but the sensors are opposite. A sensor owns poor resources such as a small memory, the low battery and the weak computational ability. So it is important to use the sensors economically. The structure of the WMSN is shown in Fig. 1.

The data collected by sensors are sensitive and important. If someone obtains the patient's data illegally, the privacy of the patient is broken. If the patient's data are distorted, the health professionals will make wrong diagnoses and fateful consequences may occur. So it is important to provide a secure communicating environment. Researchers have proposed some authentication schemes for WMSNs, like [9, 19]. But they have some weaknesses like no mutual authentication or under information leakage attacks.

With the password, the smart card is widely used in the authentication process in last decades. They form the

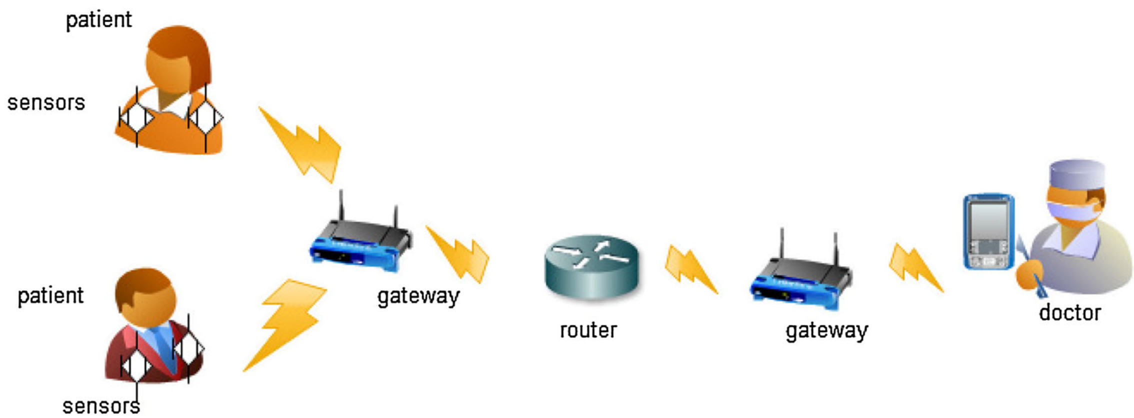

Fig. 1 Structure of WMSN

method called two-factor authentication. The smart card is a suitable storage device issued by a trusted server or a gateway and is mastered by the user. Many two-factor authentication schemes have been proposed [4, 11, 12, 17, 20–22, 31, 34]. Some researchers have tried to review such papers to improve the communication security [18, 28–30]. In 2009, Das [2] proposed a new two-factor authentication scheme for WSNs. Later, many papers [1, 10, 13, 26] pointed out that there were many weaknesses in [2], such as no mutual authentication, under the sensor node capture attack, the off-line password guessing attack, the impersonation attack, the insider attack and so on. Chen et al. [1] and Khan et al. [13] proposed their improved schemes respectively. But Yoo et al. [35] showed that the two schemes were also insecure. Chen et al.'s scheme could not withstand the replay attack and the impersonation attack and Khan et al.'s scheme could not achieve mutual authentication. In 2012, Kumar et al. [16] proposed a novel authentication scheme for WMSN and they thought their own scheme met every security requirement. However, He et al. [5] pointed out that Kumar et al.'s scheme could not resist the insider attack and the off-line guessing attack. Furthermore, it did not keep user anonymous. So He et al. presented an improvement. But Li et al. [23] pointed out that the improvement has weaknesses such as no password detection and under the de-synchronization attack. Moreover, we find that there are some other disadvantages in He et al.'s scheme. It could not withstand the off-line guessing attack, the user impersonation attack and the sensor node capture attack. We also give an improved scheme for WMSN to overcome the disadvantages.

1.1 Contribution of our paper

1. We point the weaknesses of He et al.'s authentication scheme for Health-care Applications with Wireless Medical Sensor Networks.
2. We present an improved scheme under the environment of Wireless Medical Sensor Networks.
3. To prove the security of our scheme, we use the tool Proverif to give a formal verification, and analyze the security properties concretely.

1.2 Organization of our paper

The remainder of the paper is organized as follows. We show some notations and discussions about the off-line guessing and sensor node capture attack in Sect. 2. Section 3 shows He et al.'s scheme and its disadvantages. In Sects. 4, 5 and 6, we list our scheme, its formal verification and security analysis respectively. Performance comparison is in Sect. 7 and at last the conclusion is given in Sect. 8.

2 Preliminaries

2.1 Notations

- U_i, ID_i, PW_i: The i-th user, his identity and password
- GW, ID_g: The gateway node and its identity
- S_n: The n-th sensor
- ΔT: A permitted time threshold
- SK: The session key
- A: A malicious adversary
- $E_k(.)/D_k(.)$: The symmetric encryption/decryption algorithm with key k
- $h(.)$: The one-way hash function
- $a\|b$: The conjunction of strings a and b
- $a \oplus b$: The X-OR operation of a and b

2.2 Model of the attacker

Based on Dolev-Yao model [3] and the papers [5, 15, 18, 24, 25], we use the model of the attacker as follows and we only consider that the attacker A has the above abilities but do not care about how to achieve them.

1. In polynomial time, the attacker A cannot crack the secret keys in the gateway, the results of the hash functions and the random numbers.
2. In polynomial time, A can guess the user U_i's password by retrieving one element from a finite set on and on. Similar guessing can be finished for U_i's identity. This item is based on [5, 28–30].
3. A can control the public channel. In other words, A can eavesdrop, intercept, modify or generate the messages between the participants in the public channel.
4. A can get the stored data from U_i's smart card.
5. A can compromise some but not all sensors in the WSN.

2.3 Discussion of the off-line guessing attack

In the last decade, researchers have concentrated on the off-line password guessing attack in authentication schemes with smart cards. According to [15, 24, 25], A can get data from a smart card. Also, it is a common view that the passwords of the users are stored in a finite set where the elements can be tested in polynomial time. Then A can guess a password to try to break the privacy of the user with the data stolen from the smart card and eavesdropped from the public channel. If the user's identity is anonymous, it is usually considered to be hard to get. But in He et al.'s paper [5], the authors think that the users' identities are in a small dictionary, too. And it should be easy to remember. Based on the thought, they consider Kumar et al.'s scheme [16] to be insecure. Here we expand the notion as "off-line guessing attack" and we use it to analyze He et al.'s scheme and ours.

2.4 Discussion of the sensor node capture attack

In wireless sensor networks, sensors are considered to be weak and they may be compromised. The process of this attack is: if one sensor is captured by A, A will use its stored data to forge the other sensors. Obviously many sensors are in the WMSN, actually on or in one patient's body. So this attack should be noticed.

2.5 Discussion of the de-synchronization attack

Li et al. [23] showed that He et al.'s scheme was under the de-synchronization attack. This attack occurs if there is some inconsistency in a legal users normal authentication process. For example, if there is no checking mechanism in password change phase, a user may input a wrong old password by mistake. This will lead to the failure of subsequent authentication.

3 Review and analysis of He et al.'s scheme

3.1 Outline of He et al.'s scheme

He et al.'s scheme contains four phases: professional registration, patient registration, login and authentication and password change. Since Li et al. [23] criticized the password change phase, which is under the de-synchronization attack. We only use their conclusion and omit the phase here.

First there are three premises:

1. The registration center is trusted.
2. The gateway node has three 256 bits secret keys: J, K and Q.
3. The sensor node and the gateway share a key $SK_{gs} = h(ID_g||Q)$.

3.1.1 Professional registration

1. U_i chooses ID_i, PW_i and a nonce r_i and sends ID_i with $h(PW_i||r_i)$ to GW confidentially.
2. GW generates a nonce r_g and computes $C_{ig} = E_J(r_g||ID_i||ID_g)$ and $N_i = h(ID_i||ID_g||K) \oplus h(PW_i||r_i)$. GW stores $h(.)$, C_{ig} and N_i into a smart card and sends it to U_i via a secure channel.
3. U_i injects r_i into the smart card.

3.1.2 Patient registration

1. The patient sends his name to the registration center.
2. The registration center selects a suitable sensor kit and appoints professionals.
3. The registration center gives corresponding professionals the patient's identity ID_{pt} and information about medical sensors.

3.1.3 Login and authentication

A health professional U_i can access the data of the patients in the WMSN. The steps are listed as follows:

1. U_i inserts his smart card into the terminal and inputs ID_i and PW_i. The card selects two random numbers M and N with a timestamp T_1, computes

$$R_i = N_i \oplus h(PW_i||r_i)$$
$$h_1 = h(ID_i||C_{ig}||S_n||M||N||T_1)$$
$$CID_i = E_{R_i}(h_1||S_n||M||N)$$

and sends $m_1 = \{C_{ig}, CID_i, T_1\}$ to GW.

2. GW chooses the timestamp T_2 and checks if $T_2 - T_1 < \Delta T$. If not, GW rejects the session. Otherwise, GW calculates

$$r'_g||ID'_i||ID'_g = D_J(C_{ig})$$
$$R'_i = h\left(ID'_i||ID'_g||K\right)$$
$$h'_1||S'_n||M'||N' = D_{R'_i}(CID_i)$$

and checks $h_1? = h(ID'_i||C_{ig}||S'_n||M'||N'||T_1)$. If not, GW rejects the session. Otherwise, GW produces a nonce r'_g, picks up the timestamp T_3, computes

$$C'_{ig} = E_J(r'_g||ID'_i||ID_g)$$
$$h_2 = h\left(C'_{ig}||R'_i\right)$$
$$B_i = E_{N'}\left(C'_{ig}||h_2\right)$$
$$A_i = E_{SK_{gs}}(ID'_i||S'_n||M'||B_i||T_1||T_3)$$

and sends $m_2 = \{A_i, T_3\}$ to S_n. We should say that there is no T_1 in A_i in the original paper. But that does not make sense because S_n can not use T_1 to construct the session key below if T_1 is not sent.

3. After S_n receives m_2, it picks up the timestamp T_4 and checks if $T_4 - T_3 < \Delta T$. If not, it stops the session. Then it decrypts A_i with SK_{gs} and gets ID''_i, S''_n, M'', B'_i, T'_1 and T'_3. S_n checks the correctness of S''_n and T'_3. If they are right, S_n selects the timestamp T_5, calculates $SK = h(ID''_i||S_n||M''||T_1||T_5)$ and $L = E_{SK}(S_n||B'_i||T_5)$ and sends $m_3 = \{L, T_5\}$ to U_i.
4. U_i checks T_5, computes $SK = h(ID_i||S_n||M||T_1||T_5)$ and decrypts L to get S'''_n, B''_i and T'_5. U_i checks $S_n? = S'''_n$ and $T'_5? = T_5$. If not, the session will be aborted. Then U_i decrypts B''_i with N, gets $C''_{ig}||h'_2$ and checks $h'_2? = h(C''_{ig}||R_i)$. If it is right, U_i replaces C_{ig} with C''_{ig}.

3.2 Weaknesses in He et al.'s scheme

3.2.1 Off-line guessing attack and user impersonation attack

According to Sects. 2.2 and 2.3, A guesses ID^* and PW^* for candidates and gets C^{old}_{ig}, CID_i, T_1, A_i, T_3, L and T_5 from channel and C_{ig}, r_i and N_i from the smart card. Then A computes $R^* = N_i \oplus h(PW^*||r_i)$, decrypts CID_i, gets h^*_1, S^*_n, M^* and N^* and checks if $h^*_1 = h(ID^*||C^{old}_{ig}||S^*_n||M^*||N^*||T_1)$. Finally A can gain the correct ID_i and PW_i and impersonate U_i to start a session. The same thing happens on Kumar et al.'s scheme [16].

We should note that scheme in [14] cannot resist the two attacks and we only list the corresponding attack details here. In [14], if A gets the data C_{ug}, N_u, PK_u and PK_g from the smart card and C_{u1} from the channel, A guesses ID^* and PW^* to calculate $K^*_u = PK_u \oplus (ID^*||PW^*)$, $K^*_g = PK_g \oplus (PW^*||ID^*)$ and checks if $N_u = h(ID^*||PW^*||K^*_u)$ and $C_{u1} = C_{ug} \oplus h(K^*_g)$. A may try all possible identities and passwords in dictionaries and at last succeed. Like He et al.'s scheme, it is also vulnerable to the user impersonation attack.

3.2.2 Sensor node capture attack

According to Sect. 2.4, if A compromises a sensor S_p, he could get SK_{gs}, which is distributed in all sensors. He can use SK_{gs} to decrypt all A_i sent to any other sensors and then forge the corresponding reply. Moreover, this disadvantage also appears on schemes in [14, 16].

4 Outline of our scheme

Our scheme also includes four phases as He et al.'s. But the premises are a little different. The login and authentication phase is shown in Fig. 2.

1. The registration center is trusted.
2. The gateway node has only one 256 bits secret key K.
3. Each sensor node S_n and the gateway share a key $SK_{sn} = h(S_n||K)$. That means different sensors have different shared keys.

4.1 Professional registration

1. U_i chooses ID_i, PW_i and a nonce r_i, computes $HPW_i = h(PW_i||r_i)$ and sends $\{ID_i, HPW_i\}$ to GW via a secure channel.
2. GW generates a random number r_g, computes $C_{ig} = E_K(r_g||ID_i||ID_g)$ and $B_1 = h(r_g||ID_i||ID_g||K) \oplus HPW_i$, stores $\{C_{ig}, B_1, h(.)\}$ into a smart card and issues the card to U_i via a secure channel. GW stores ID_i for auditing.
3. U_i computes $B_2 = h(ID_i||PW_i) \oplus r_i$ and injects B_2 into the smart card.

4.2 Patient registration

This phase is the same as He et al.'s. So we omit it here.

4.3 Login and authentication

1. U_i inserts the smart card into the terminal and inputs ID_i with PW_i. Then the card computes $r'_i = B_2 \oplus h(ID_i||PW_i)$ and $HPW_i = h(PW_i||r'_i)$.
2. The card selects random numbers M_i, N_i and V_i and a sensor S_n, computes

U_i	GW	S_n

input ID_i,PW_i
compute $r_i^{'} = B_2 \oplus h(ID_i||PW_i)$,
$HPW_i = h(PW_i||r_i^{'})$
generate M_i, N_i, V_i, select S_n
compute:
$C_1 = B_1 \oplus HPW_i \oplus M_i$
$C_2 = h(ID_i||C_{ig}||S_n||M_i||N_i||V_i)$
$CID_i = E_{M_i}(C_2||S_n||M_i||N_i||V_i)$

$\xrightarrow{m_1=\{C_{ig},CID_i,C_1\}}$

decrypt C_{ig} to get $r_g^{'}$, $ID_i^{'}$ and $ID_g^{'}$
check $ID_g^{'}$ and $ID_i^{'}$
compute $M_i^{'} = C_1 \oplus h(r_g^{'}||ID_i^{'}||ID_g||K)$
decrypt CID_i to get $C_2^{'}$, $S_n^{'}$, $M_i^{''}$, $N_i^{'}$ and $V_i^{'}$
check $M_i^{'}? = M_i^{''}$
$C_2^{'}? = h(ID_i^{'}||C_{ig}||S_n^{'}||M_i^{'}||N_i^{'}||V_i^{'})$
generate r_g^{new}
compute $C_{ig}^{'} = E_K(r_g^{new}||ID_i^{'}||ID_g)$
$C_3 = h(C_{ig}^{'}||N_i^{'})$
$C_4 = E_{N_i^{'}}(C_{ig}^{'}||C_3)$
$C_5 = h(r_g^{new}||ID_i^{'}||ID_g||K) \oplus h(N_i^{'}||C_2^{'})$
$C_6 = E_{SK_{sn}}(ID_i^{'}||S_n^{'}||V_i^{'}||C_4)$
$C_7 = h(ID_i^{'}||C_{ig}^{'}||S_n^{'}||M_i^{'}||N_i^{'}||V_i^{'}||C_5)$

$\xrightarrow{m_2=\{C_5,C_6,C_7\}}$

decrypt C_6 to obtain $ID_i^{''}$, $S_n^{''}$, $V_i^{''}$ and $C_4^{'}$
check if $S_n^{''}$ is right
produce V_n
compute $SK = h(ID_i^{''}||S_n||V_i^{''}||V_n)$
$C_8 = V_n \oplus h(C_7||V_i^{''})$
$C_9 = C_4^{'} \oplus h(C_7||V_n)$
$C_{10} = h(ID_i^{''}||S_n||V_i^{''}||V_n||SK||C_8||C_9)$

$\xleftarrow{m_3=\{C_5, C_7, C_8, C_9, C_{10}\}}$

compute $V_n^{'} = C_8 \oplus h(C_7||V_i)$
$SK = h(ID_i||S_n||V_i||V_n^{'})$
check $C_{10}? = h(ID_i||S_n||V_i||V_n^{'}||SK||C_8||C_9)$
compute $C_4^{''} = C_9 \oplus h(C_7||V_n^{'})$
decrypt $C_4^{''}$ to get $C_{ig}^{''}$ with $C_3^{'}$
check $C_3^{'}? = h(C_{ig}^{''}||N_i)$
$C_7? = h(ID_i||C_{ig}^{''}||S_n||M_i||N_i||C_5)$
compute $B_1^{new} = C_5 \oplus HPW_i \oplus h(N_i||C_2)$
replace (B_1, C_{ig}) with $(B_1^{new}, C_{ig}^{''})$ respectively

Fig. 2 Login and authentication phase

$$C_1 = B_1 \oplus HPW_i \oplus M_i$$
$$C_2 = h(ID_i||C_{ig}||S_n||M_i||N_i||V_i)$$
$$CID_i = E_{M_i}(C_2||S_n||M_i||N_i||V_i)$$

and sends $m_1 = \{C_{ig}, CID_i, C_1\}$ to GW.

3. GW decrypts C_{ig} to get r_g', ID_i' and ID_g', checks if ID_g' and ID_i' are valid. If either is not, it rejects the session. Then it computes $M_i' = C_1 \oplus h(r_g'||ID_i'||ID_g||K)$ and decrypts CID_i to get C_2', S_n', M_i'', N_i' and V_i'. GW checks $M_i'? = M_i''$ and $C_2'? = h(ID_i'||C_{ig}||S_n'||M_i'||N_i'||V_i')$. If

either of them is wrong, the session will be terminated.

4. GW generates a nonce r_g^{new}, computes

$$C'_{ig} = E_K\left(r_g^{new}||ID'_i||ID_g\right)$$
$$C_3 = h\left(C'_{ig}||N'_i\right)$$
$$C_4 = E_{N'_i}\left(C'_{ig}||C_3\right)$$
$$C_5 = h\left(r_g^{new}||ID'_i||ID_g||K\right) \oplus h\left(N'_i||C'_2\right)$$
$$C_6 = E_{SK_{sn}}\left(ID'_i||S'_n||V'_i||C_4\right)$$
$$C_7 = h\left(ID'_i||C'_{ig}||S'_n||M'_i||N'_i||V'_i||C_5\right)$$

and sends $m_2 = \{C_5, C_6, C_7\}$ to S_n.

5. After S_n receives m_2, it decrypts C_6 to obtain ID''_i, S''_n, V''_i and C'_4. S_n checks if S''_n is right. If it is right, S_n produces a nonce V_n, calculates

$$SK = h\left(ID''_i||S_n||V''_i||V_n\right)$$
$$C_8 = V_n \oplus h\left(C_7||V''_i\right)$$
$$C_9 = C'_4 \oplus h(C_7||V_n)$$
$$C_{10} = h\left(ID''_i||S_n||V''_i||V_n||SK||C_8||C_9\right)$$

and sends $m_3 = \{C_5, C_7, C_8, C_9, C_{10}\}$ to U_i.

6. The card calculates

$$V'_n = C_8 \oplus h(C_7||V_i)$$
$$SK = h(ID_i||S_n||V_i||V'_n)$$

and checks $C_{10}? = h(ID_i||S_n||V_i||V'_n||SK||C_8||C_9)$. If not, the session is rejected. Then U_i computes $C''_4 = C_9 \oplus h(C_7||V'_n)$, decrypts C''_4 and gets $C''_{ig}||C'_3$, and checks $C'_3? = h(C''_{ig}||N_i)$ and $C_7? = h(ID_i||C''_{ig}||S_n||M_i||N_i||C_5)$. If either of them fails, the session is stopped. Finally U_i computes $B_1^{new} = C_5 \oplus HPW_i \oplus h(N_i||C_2)$ and replaces (B_1, C_{ig}) with (B_1^{new}, C''_{ig}) respectively.

4.4 Password change

1. This step is same as step 1 of last phase.
2. The card selects random numbers M_i and N_i, computes C_1, $C_{11} = h(ID_i||C_{ig}||M_i||N_i)$ and $TID_i = E_{M_i}(C_{10}||M_i||N_i)$ and sends C_{ig}, C_1 and TID_i, with a password change request to GW.
3. GW decrypts C_{ig} and computes M'_i as in last phase. Then it decrypts TID_i to get C'_1, M''_i and N'_i and checks $M'_i? = M''_i$, $C_1? = C'_1$ and $C'_{11}? = h(ID'_i||C_{ig}||M'_i||N'_i)$. If either of them is wrong, the session is rejected.
4. GW produces a nonce r_g^{new}, computes

$$C'_{ig} = E_K(r_g^{new}||ID'_i||ID_g)$$
$$C_{12} = h(C'_{ig}||N'_i)$$
$$C_{13} = E_{N'_i}(C'_{ig}||C_{11})$$
$$C_{14} = h(r_g^{new}||ID'_i||ID_g) \oplus h(N'_i||ID'_i)$$
$$C_{15} = h(C_{12}||C_{13}||ID'_i||M'_i||N'_i)$$

and sends C_{13}, C_{14}, C_{15} and a grant to U_i.

5. The card decrypts C_{13} to obtain C''_{ig} and C'_{12} and checks $C'_{12}? = h(C''_{ig}||N_i)$ and $C_{15}? = h(C_{13}||C_{14}||ID_i||M_i||N_i)$. If either is wrong, U_i rejects it. Then U_i is asked to input a new password PW_i^{new}. The card generates a nonce r_i^{new}, computes

$$HPW_i^{new} = h\left(PW_i^{new}||r_i^{new}\right)$$
$$B_1^{new2} = C_{14} \oplus h(N_i||ID_i) \oplus HPW_i^{new}$$
$$B_2^{new} = h\left(ID_i||PW_i^{new}\right) \oplus r_i^{new}$$

and substitutes $(C''_{ig}, B_1^{new2}, B_2^{new})$ for (C_{ig}, B_1, B_2).

5 Formal verification with Proverif

Proverif is a popular tool to verify the security properties of cryptographical schemes. The protocol analysis with Proverif is related to some sessions with some message space. The attack can be reconstructed by Proverif: if a property cannot be proved, the execution trace which is for the failed property is shown. We illustrate the code for our scheme.

First three channels are defined for the communications. ch is for the public channel while the other two are the secure channels.

```
(*———channels————-*)
free ch: channel.
free sch1: channel [private].
free sch2: channel [private].
```

sku and sks are session keys formed in the authentication.

```
(*——-shared keys———*)
free sku: bitstring [private].
free sks: bitstring [private].
```

According to the scheme, K is the gateway's secret key and xsksn is shared between the gateway and the sensor Sn.

```
(*——GW's secret key——*)
free K:bitstring [private].
free xsksn:bitstring [private].
```

Then some constants are listed below:

```
(*——constants————*)
free IDi:bitstring [private].
free PWi:bitstring [private].
const Sn:bitstring.
const IDg:bitstring.
table d(bitstring).
```

Referred functions and rules are listed below:

```
(*———-functions———-*)
fun h(bitstring):bitstring. (*hash function*)
fun senc(bitstring,bitstring):bitstring. (*symmetric encryption*)
fun xor(bitstring,bitstring):bitstring.
fun con(bitstring,bitstring):bitstring. (*string concatenation*)

(*———reduction————*)
reduc forall m:bitstring, n:bitstring; sdec(senc(m,n),n)=m.
(*symmetric decryption*)

(*———equations————*)
equation forall m:bitstring,n:bitstring; xor(xor(m,n),n)=m.
```

The aims of the verification are to prove the following three queries. The third is about the correct authentication process of the author and the corresponding two events are listed.

```
(*————-queries———-*)
query attacker(sku).
query attacker(sks).
query id:bitstring; inj-event(UserAuth(id))==>
inj-event(UserStart(id)).

(*—————event————-*)
event UserStart(bitstring).
event UserAuth(bitstring).
```

The processes of the user, the sensor and the gateway are demonstrated as follows:

```
(*———User's process———*)
let User=
new ri:bitstring;
let HPWi=h(con(PWi , ri)) in
out(sch1,(IDi,HPWi));
in(sch1,(Cig:bitstring,B1:bitstring));
let B2 = xor(h(con(IDi,PWi)),ri) in
!
(
event UserStart(IDi);
let ri' = xor(B2,h(con(IDi,PWi))) in
let HPWi' = h(con(PWi , ri')) in
new Mi:bitstring;
new Ni:bitstring;

new Vi:bitstring;
let C1 = xor(xor(B1,HPWi'),Mi) in
let C2 = h(con(con(con(con(con(IDi,Cig),Sn),Mi),Ni),Vi))
in
let CIDi = senc(con(con(con(con(C2,Sn),Mi),Ni),Vi),Mi) in
let M1 = (Cig,CIDi,C1) in
out(ch,M1);
in (ch,(C5:bitstring,C7:bitstring,C8:bitstring,C9:bitstring,
C10:bitstring));
let Vn'= xor(C8,con(C7,Vi)) in
let sku= h(con(con(con(IDi,Sn),Vi),Vn')) in
if C10 = h(con(con(con(con(con(con(IDi,Sn),Vi),Vn'),sku)
,C8),C9)) then
let C4"= xor(C9,h(con(C7,Vn'))) in
let (Cig":bitstring,C3':bitstring)= sdec(C4",Ni) in
if C3'=h(con(Cig",Ni)) then
if C7 = h(con(con(con(con(con(IDi,Cig"),Sn),Mi),Ni),C5))
then
let B1new = xor(xor(C5,h(con(Ni,C2))),HPWi) in
let B1 = B1new in
let Cig = Cig" in
0
).
```

```
(*———Sensor's process—-*)
let Sensor =
in(sch2,xsksn:bitstring);
!
(
in(ch,(xxC5:bitstring,xxC6:bitstring,xxC7:bitstring));
let (IDi":bitstring,Sn":bitstring,Vi":bitstring,C4':bitstring)=
sdec(xxC6,xsksn) in
if Sn" = Sn then
new Vn:bitstring;
let sks = h(con(con(con(IDi",Sn),Vi"),Vn)) in
let C8 = xor(Vn,h(con(xxC7,Vi"))) in
let C9 = xor(C4',h(con(xxC7,Vn))) in
let C10 = h(con(con(con(con(con(con(IDi",Sn),Vi"),Vn),sks),
C8),C9)) in
out(ch,(xxC5,xxC7,C8,C9,C10));
0
).
```

```
(*——GW's process——-*)
let GWNReg1 =
in(sch1,(rIDi:bitstring,rHPWi:bitstring));
new rg:bitstring;
let Cig = senc(con(con(rg,rIDi),IDg),K) in
let xB1 = xor(rHPWi,h(con(con(con(rg,IDi),IDg),K))) in
insert d(rIDi);
out (sch1,xB1).
let GWNReg2 =
let sksn = h(con(Sn,K)) in
out(sch2,sksn).
let GWNAuth =
in (ch,(xCig:bitstring,xCIDi:bitstring,xC1:bitstring));
let (rg':bitstring,IDi':bitstring,IDg':bitstring)= sdec (xCig,K)
in
if IDg = IDg' then
get d(=IDi') in
let Mi'= xor(xC1,h(con(con(con(rg',IDi'),IDg),K))) in
let (C2':bitstring,Sn':bitstring,Mi":bitstring,Ni':bitstring,
Vi':bitstring) = sdec(xCIDi,Mi') in
if Mi'=Mi" then
if C2'=h(con(con(con(con(con(IDi',xCig),Sn'),Mi'),Ni'),Vi'))
then
event UserAuth(IDi');
new rgnew:bitstring;
let Cig' = senc(con(con(rgnew,IDi'),IDg),K) in
let C3 = h(con(Cig',Ni')) in
let C4 = senc(con(Cig',C3),Ni') in
let C5 = xor(h(con(con(con(rgnew,IDi'),IDg),K)),
h(con(Ni',C2'))) in
let C6 = senc(con(con(con(IDi',Sn'),Vi'),C4),xsksn) in
let C7 = h(con(con(con(con(con(con(IDi',Cig'),Sn'),Mi'),Ni')
,Vi'),C5)) in
out(ch,(C5,C6,C7)).
```

$$let\ GWN = GWNReg1|GWNReg2|GWNAuth.$$

The whole execution is below:

$$process\ !User|!GWN|!Sensor$$

We list the results of the code as follows:

– Query inj-event(UserAuth(id)) ==> inj-event(UserStart(id))
RESULT inj-event(UserAuth(id)) ==> inj-event(UserStart(id)) is true.
This result means that the execution of the event UserStart(id) is preceded by the execution of the event UserAuth(id). And UserStart(id) is independent from UserAuth(id).

– Query not attacker(sks[])
RESULT not attacker(sks[]) is true.
The result means there is no trace for the adversary to reconstruct sks. Or sks is secure to resist cracking.

– Query not attacker(sku[])
RESULT not attacker(sku[]) is true.
The result means there is no trace for the adversary to reconstruct sku. Or sku is secure to resist cracking.

So we can see that our scheme is secure from the above results.

6 Security analysis of our scheme

In this section we discuss the security of our scheme. We also list some recent schemes [5, 14, 16] for comparison. The results are demonstrated in Table 1. Some weaknesses about schemes in [5, 14, 16] have already been referred in Sects. 1 and 3.2.

6.1 Resistant to the insider attack

In professional registration phase, $HPW_i = h(PW_i||r_i)$ is submitted to GW. PW_i is protected by a random number r_i

Table 1 The comparison of security characters

	[5]	[14]	[16]	Ours
Resistant to the insider attack	No	Yes	Yes	Yes
Resistant to the off-line guessing attack	No	No	No	Yes
Resistant to the user impersonation attack	No	No	No	Yes
Resistant to the *GW* spoofing attack	Yes	Yes	Yes	Yes
Resistant to the sensor spoofing attack	Yes	Yes	Yes	Yes
Resistant to the de-synchronization attack	Yes	No	Yes	Yes
Resistant to the sensor capture attack	No	No	No	Yes
User anonymity	No	Yes	Yes	Yes
Mutual authentication	Yes	Yes	Yes	Yes

and a one-way hash function $h(.)$. The malicious administrator A could not get the real PW_i.

6.2 Resistant to the off-line guessing attack

Suppose A gets B_1, B_2 and C_{ig} from U_i's smart card and $\{m_1^{old}, m_2^{old}, m_3^{old}\}$ from the last session, he guesses (ID^*, PW^*), and calculates $HPW^* = h(PW^*||B_2 \oplus h(ID^*||PW^*))$. Only $B_1 = C_5^{old} \oplus h(N_i^{old}||C_2^{old}) \oplus HPW^*$ can be used to check the guessing result. N_i^{old} varies in every session and is protected in CID_i^{old} which is encrypted by the random number M_i^{old}. Unfortunately, M_i^{old} can only be calculated by the equation $h(r_g^{old}||ID_i||ID_g||K) \oplus M_i^{old} = C_1^{old}$. Since A can not get K from the trusted gateway and r_g^{old} is also a random number which cannot be tracked, M_1^{old} cannot be calculated. So ID_i and PW_i could not be guessed simultaneously in our scheme.

6.3 Resistant to the user impersonation attack

To forge a message m_1, $h(r_g||ID_i||ID_g||K)$ is needed to generate C_1. However, K and ID_i are unknown to A. So our scheme can resist this attack.

6.4 Resistant to the *GW* or sensor spoofing attack

K stored in GW and SK_{sn} shared between GW and S_n all have secure lengths. K is 256 bits and SK_{sn} is a hash result. A can not forge messages sent by GW or S_n since he can not guess the critical secret data which are needed to produce the corresponding messages.

6.5 Resistant to the de-synchronization attack

If a user wants to change his password, the old password and the identity must be checked in the password change phase via the gateway node. So once a legal user inputs the old password by mistake, it cannot be passed. The mechanism blocks the attack.

6.6 Resistant to the sensor capture attack

In our scheme, every sensor has its own secret key $h(S_n||K)$. The hash result prevents A from guessing it. So if A compromises one sensor node, sessions between the user and the other sensors will not be affected.

6.7 User anonymity

ID_i is always protected in the transmitted elements C_{ig} and C_6. Every time C_{ig} and C_6 are changed to be new encrypted strings, so A cannot track the concrete user from the messages.

6.8 Mutual authentication

1. GW can authenticate U_i by checking $M_i'? = M_i''$ and the correctness of C_2' since M_i is hidden in C_1 and CID_i, which are generated by U_i.
2. GW and S_n share the common secret key SK_{sn}, so S_n can verify GW by checking the validity of S_n'' and trust GW. For this reason, S_n could trust U_i because GW has authenticated U_i.
3. U_i can think S_n is true by checking if C_{10} is right. Moreover, U_i can authenticate GW by checking the correctness of C_3' and C_7 since only GW can calculate data with M_i and N_i by decrypting C_{ig} and CID_i successively.

So our scheme satisfies mutual authentication among the three participants.

7 Performance comparison

In this section, we compare the performance of the schemes referred in Table 1. We use T_s for time of one symmetric encryption/decryption operation and T_h for one hash operation. We use AES and SHA1 to judge the length and the time cost in corresponding calculations. According to [32, 33], $T_s = 0.1303$ ms and $T_h = 0.0004$ ms. Also, we

Table 2 The performance comparison among our scheme and other three recent schemes also using WMSN for health-care

	[16]	[5]	[14]	Ours
Time cost in login and authentication (ms)	$U_i : 2T_s + 3T_h = 0.2618$	$U_i : 3T_s + 4T_h = 0.3925$	$U_i : T_s + 5T_h = 0.1323$	$U_i : 2T_s + 10T_h = 0.2646$
	$GW : 3T_s + T_h = 0.3913$	$GW : 5T_s + 3T_h = 0.6527$	$GW : 2T_s + 9T_h = 0.2642$	$GW : 5T_s + 6T_h = 0.6539$
	$S_n : 2T_s + T_h = 0.261$	$S_n : 2T_s + T_h = 0.261$	$S_n : 7T_h = 0.0028$	$S_n : T_s + 4T_h = 0.1319$
Storage in smart card (bits)	800	832	704	832
Communication cost (bits)	2592	4192	1440	3968
Formal verification	No	No	No	Yes
Security	No	No	No	Yes

define the lengths of user's identity and random numbers are 160 bits. The comparison of time cost in login and authentication phase, storage in the smart card, communication cost, existence of formal verification and security are shown in Table 2.

We analyze the results as follows:

- Time cost of login and authentication in our scheme is the highest on the gateway side, but is very close to [5]. Such time cost in our scheme is lower than [5] on the user side and lower than [5, 16] on the sensor side.
- Our scheme costs the same as [5] in the smart card storage. It is higher than the other two schemes.
- The communication cost for our scheme is only lower than [5].
- The most important item is the security. Our scheme has a formal verification while others do not have. Moreover, our scheme is secure while others are all with disadvantages through the analysis in Sect. 6.

Because our scheme overcomes the common attacks, the indices are justified to be a bit higher in comparison. So it is clear to see that our scheme is the best among the four schemes.

8 Conclusion

In this paper, we first describe He et al.'s scheme and discuss its weaknesses including the off-line guessing attack, the user impersonation attack and the sensor capture attack. Then we propose a new authentication scheme for WMSN. Through the formal verification by Proverif and the concrete analysis, our scheme is fit for the requirements of security. Compared with some recent authentication scheme, it is more practical for the applications of healthcare with WMSN.

Acknowledgments The authors thank the valuable work of the editors and the anonymous reviewers. This research is supported by Fujian Education and Scientific Research Program for Young and Middle-aged Teachers under Grant No. JA14369, the National Natural Science Foundation of China under Grant No. 61300220, and it is also supported by PAPD and CICAEET.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Chen, T.H., Shih, W.K.: A robust mutual authentication protocol for wireless sensor networks. Etri J. **32**(5), 704–712 (2010)
2. Das, M.L.: Two-factor user authentication in wireless sensor networks. Wirel. Commun. IEEE Trans. **8**(3), 1086–1090 (2009)
3. Dolev, D., Yao, A.C.: On the security of public key protocols. Inf. Theory IEEE Trans. **29**(2), 198–208 (1983)
4. He, D., Zeadally, S.: Authentication protocol for an ambient assisted living system. Commun. Mag. IEEE **53**(1), 71–77 (2015)
5. He, D., Kumar, N., Chen, J., Lee, C.C., Chilamkurti, N., Yeo, S.S.: Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks. Multimed. Syst. **21**(1), 49–60 (2015). doi:10.1007/s00530-013-0346-9
6. He, D., Kumar, N., Chilamkurti, N.: A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks. Inf. Sci. **321**, 236–277 (2015). doi:10.1016/j.ins.2015.02.010
7. He, D., Zeadally, S., Wu, L.: Certificateless public auditing scheme for cloud-assisted wireless body area networks. IEEE Syst. J. (2015). doi:10.1109/JSYST.2015.2428620
8. Hsiao, T.C., Liao, Y.T., Huang, J.Y., Chen, T.S., Horng, G.B.: An authentication scheme to healthcare security under wireless sensor networks. J. Med. Syst. **36**(6), 3649–3664 (2012). doi:10.1007/s10916-012-9839-x
9. Hu, F., Jiang, M., Wagner, M., Dong, D.C.: Privacy-preserving telecardiology sensor networks: toward a low-cost portable wireless hardware/software codesign. Inf. Technol. Biomed. IEEE Trans. **11**(6), 619–627 (2007)
10. Huang, H.F., Chang, Y.F., Liu, C.H.: Enhancement of two-factor user authentication in wireless sensor networks. In: Intelligent information hiding and multimedia signal processing (IIH-MSP), sixth International Conference on, IEEE, pp. 27–30 (2010)
11. Karuppiah, M., Saravanan, R.: A secure remote user mutual authentication scheme using smart cards. J. Inf. Secur. Appl. **19**(4), 282–294 (2014)
12. Karuppiah, M., Saravanan, R.: A secure authentication scheme with user anonymity for roaming service in global mobility networks. Wirel. Pers. Commun. (2015). doi:10.1007/s11277-015-2524-x
13. Khan, M.K., Alghathbar, K.: Cryptanalysis and security improvements of two-factor user authentication in wireless sensor networks. Sensors **10**(3), 2450–2459 (2010)
14. Khan, M.K., Kumari, S.: An improved user authentication protocol for healthcare services via wireless medical sensor networks. Int. J. Distrib. Sens. Netw. (2014)
15. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Advances in Cryptology-CRYPTO99. Springer, pp. 388–397 (1999)
16. Kumar, P., Lee, S.G., Lee, H.J.: E-sap: Efficient-strong authentication protocol for healthcare applications using wireless medical sensor networks. Sensors **12**(2), 1625–1647 (2012)
17. Kumari, S., Gupta, M.K., Khan, M.K., Li, X.: An improved timestamp-based password authentication scheme: comments, cryptanalysis, and improvement. Secur. Commun. Netw. **7**(11), 1921–1932 (2014)
18. Kumari, S., Khan, M.K., Atiquzzaman, M.: User authentication schemes for wireless sensor networks: a review. Ad Hoc Netw. **27**, 159–194 (2015)
19. Le, X.H., Khalid, M., Sankar, R., Lee, S.: An efficient mutual authentication and access control scheme for wireless sensor networks in healthcare. J. Netw. **6**(3), 355–364 (2011)
20. Li, X., Xiong, Y., Ma, J., Wang, W.: An efficient and security dynamic identity based authentication protocol for multi-server architecture using smart cards. J. Netw. Comput. Appl. **35**(2), 763–769 (2012)
21. Li, X., Ma, J., Wang, W., Xiong, Y., Zhang, J.: A novel smart card and dynamic id based remote user authentication scheme for multi-server environments. Math. Comput. Model. **58**(1), 85–95 (2013)
22. Li, X., Niu, J., Khan, M.K., Liao, J.: An enhanced smart card based remote user password authentication scheme. J. Netw. Comput. Appl. **36**(5), 1365–1371 (2013)
23. Li, X., Niu, J., Kumari, S., Liao, J., Liang, W., Khan, M.K.: A new authentication protocol for healthcare applications using

wireless medical sensor networks with user anonymity. Secur. Commun. Netw. (2015)

24. Mangard, S., Oswald, E., Standaert, F.X.: One for all-call for one: unifying standard differential power analysis attacks. IET Inf. Secur. **5**(2), 100–110 (2011)
25. Messerges, T.S., Dabbish, E.A., Sloan, R.H.: Examining smart-card security under the threat of power analysis attacks. IEEE Trans. Comput. **51**(5), 541–552 (2002)
26. Nyang, D., Lee, M.K.: Improvement of das's two-factor authentication protocol in wireless sensor networks. IACR Cryptol. ePrint Arch. **2009**, 631 (2009)
27. Raja, K.N., Beno, M.M.: On securing wireless sensor network-novel authentication scheme against dos attacks. J. Med. Syst. **38**(10), 1–5 (2014). doi:10.1007/s10916-014-0084-3
28. Wang, D., Wang, P.: On the anonymity of two-factor authentication schemes for wireless sensor networks: attacks, principle and solutions. Comput. Netw. **73**, 41–57 (2014)
29. Wang, D., Wang, P.: Understanding security failures of two-factor authentication schemes for real-time applications in hierarchical wireless sensor networks. Ad Hoc Netw. **20**, 1–15 (2014)
30. Wang, D., He, D., Wang, P., Chu, C.: Anonymous two-factor authentication in distributed systems: Certain goals are beyond attainment. IEEE Trans. Dependable and Secure Computing (2014). doi:10.1109/TDSC.2014.2355850
31. Wu, F., Xu, L.: Security analysis and improvement of a privacy authentication scheme for telecare medical information systems. J. Med. Syst. **37**(4), 1–9 (2013)
32. Wu, F., Xu, L., Kumari, S., Li, X.: A novel and provably secure biometrics-based three-factor remote authentication scheme for mobile client-server networks. Comput Elect. Eng. (2015). doi:10.1016/j.compeleceng.2015.02.015
33. Xu, L., Wu, F.: Cryptanalysis and improvement of a user authentication scheme preserving uniqueness and anonymity for connected health care. J. Med. Syst. **39**(2), 1–9 (2015)
34. Xu, L., Wu, F.: An improved and provable remote user authentication scheme based on elliptic curve cryptosystem with user anonymity. Secur. Commun. Netw. **8**(2), 245–260 (2015). doi:10.1002/sec.977
35. Yoo, S.G., Park, K.Y., Kim, J.: A security-performance-balanced user authentication scheme for wireless sensor networks. Int. J. Distrib. Sens. Netw. (2012)