On the security of Xu et al.'s authentication and key agreement scheme for telecare medicine information systems

SK Hafizul Islam [a]

Department of Computer Science and Information Systems, Birla Institute of Technology and Science, Pilani, Rajasthan 333 031, India

Abstract

In 2014, Xu et al. proposed a two-factor mutual authentication and key agreement scheme for telecare medicine information system (TIMS) based on elliptic curve cryptography (ECC). However, it has been shown that Xu et al.'s scheme is not suitable for practical use as it is many problems. As a remedy, an improved scheme is proposed with better security and functionality attributes.

Keywords: Anonymity; Remote user authentication; Security; Smartcard, Cryptanalysis; Authentication; Hash function; Password.

1. Introduction

Nowadays, the paper-based medical information systems are inefficient and inconvenient to use due to the following reasons: (1) it is often insufficient in quality, error prone and poorly organized, (2) it is often not in time or not available in time or incomplete or inconsistent and cannot be accessible at anytime from anywhere and (3) the space requirement for storing, routing, archiving and maintenance of the documents are high. With the potential growth of computer networks and Internet, the historic paper-based medical information systems are now being replaced to the electronic media-based systems (e-medicine) gradually [1, 2, 3, 4].

Recently, many password authentication schemes [6, 5, 7] have been proposed in the field of the Telecare Medicine Information System (TIMS). In 2014, Xu et al. [8] presented a two-factor mutual authentication and key agreement scheme using elliptic curve cryptography (ECC) for TIMS service. However, in this paper, it has been proved that Xu et al.'s scheme is not efficient due to the following reasons: (1) it fails to achieve strong authentication in login and authentication phases; (2) it fails to update the password correctly in the password change phase; (3) it fails to provide the revocation of lost/lost smartcard; (4) it fails to protect the strong replay attack; and (5) it has the overhead of public key certificate management. An improved scheme is also proposed in this paper, that not only overcome the flaws of Xu et al.'s scheme, but also provides other attacks resilience and functionality requirements.

The paper is organized in the following ways. In Section 2, the brief introduction of the theory of elliptic curve and some computational problems are given. The brief review of Xu et al.'s scheme is given in Section 3. The cryptanalysis of Xu et al.'s scheme is given in Section 4. The improved scheme is described in Section 5. The security analysis of the proposed scheme is given in Section 6. Finally, in Section 7, some concluding remarks are given.

2. Mathematical preliminaries

This section discussed the theory of elliptic curve cryptography and some mathematical hard problems on it.

[a]Corresponding author: hafi786@gmail.com, hafizul.ism@gmail.com, hafizul@pilani.bits-pilani.ac.in, ☎: +91-8797369160

April 30, 2014

2.1. *Theory of elliptic curve*

Recently, Elliptic curve cryptography (ECC) has accepted as an efficient tool in public key cryptography (PKC) due to the computation, communication and security strengths. For example, it offers same level of security at reduced key sizes than other PKCs. Below is the brief description of ECC.

Let E/F_p be a set of elliptic curve points over a prime field F_p, defined by the following non-singular elliptic curve:

$$y^2 \bmod p \quad = \quad (x^3 + ax + b) \bmod p \tag{1}$$

where $x, y, a, b \in F_p$ and $(4a^3 + 27b^2) \bmod p \neq 0$. A point $P(x, y)$ is an elliptic curve point if it satisfies Equ. (1), and the point $Q(x, -y)$ is called the negative of P, i.e. $Q = -P$. Let $P(x_1, y_1)$ and $Q(x_2, y_2)(P \neq Q)$ be two points on (1), the line l (tangent to the curve (1) if $P = Q$) joining the points P and Q intersects the curve (1) at $-R(x_3, -y_3)$ and the reflection of it with respect to *x-axis* is the point $R(x_3, y_3)$, i.e. $P + Q = R$. The points E/F_p together with a point O, called *"point at infinity"* or *"zero point"*, makes an additive elliptic curve cyclic group G_p, i.e. $G_p = \{(x, y) : x, y \in F_p \text{ and } (x, y) \in E/F_p\} \cup \{O\}$ of prime order p. The scalar point multiplication on G_p is defined as: $k \cdot P = P + P + \cdots + P$ (k times). A generator point $P \in G_p$ has order n if n is the smallest positive integer and $n \cdot P = O$ [9].

2.2. *Mathematical hard problems*

This section summarizes some existing computational problems on the elliptic curve group.

Definition 1. Elliptic Curve Discrete Logarithm Problem (ECDLP): Given a tuple $(P, Q) \in G_p$, it is computationally hard by a polynomial-time bounded algorithm to find an integer $a \in Z_p^*$ such that $Q = aP$.

Definition 2. Computational Diffie-Hellman Problem (CDHP): Given a tuple $(P, aP, bP) \in G_p$ for any $a, b \in Z_p^*$, computation of abP is hard by a polynomial-time bounded algorithm.

3. Review of Xu et al.'s scheme

In this section, we reviewed Xu et al.'s two-factor authentication with key agreement scheme based on elliptic curve for telecare medical information systems [8]. The list of notations are illustrated in Table 1. Xu et al.'s scheme is composed of four phases, called registration phase, login phase, authentication phase, and password update phase.

Initially, the TIMS server S chooses an elliptic curve (1) and the group E/F_p with a base point P of order n, which is a large prime number. Then S selects a random number $s \in Z_p^*$ as the private key and computes the corresponding public key as $Y = s \cdot P$. In addition, S also chooses two one-way hash functions $h()$ and $h_1()$, respectively.

Table 1: Different notations used in this paper.

Notations	Description
ID	The identity of the patient U
PW	The password of the patient U
S	The telecare server in TMIS
s	The private key of S
Y	The public key of S, $Y = s \cdot P$
Z_p^*	The multiplicative group of Z_p
$h(), h_1()$	Two secure and one-way hash functions, $h(), h_1() : \{0, 1\}^* \rightarrow Z_p^*$
P	The base point of E/F_p
$\|\|$	The string concatenation operator
$\oplus$	The bitwise XOR operator

3.1. Registration phase

To become a legal user of TMIS server S, the patient U should performed the following operations:

(a). U chooses his/heridentity ID, password PW and a random number $r \in_R Z_p^*$. Then U sends his/her ID and $A = h(PW\|r)$ to S through a secure channel.

(b). Upon receiving ID and A from U, S computes $M = h(s\|ID)$ and $B = M \oplus A$.

(c). S then stores the parameters $\{E/F_p, P, Y, B, h(), h_1()\}$ into a new smartcard and sends it to U via a secure channel.

(d). After receiving the smartcard, U stores r into it. Finally, the smartcard contains $\{E/F_p, P, Y, B, r, h(), h_1()\}$.

3.2. Login phase

In order to get the services from S, U needs to send a login message to S. The steps should be performed as follows:

(a). U inserts the smartcard into the smart device and inputs ID and PW in to the smartcard. Then the smartcard computes $A = h(PW\|r)$, $M = B \oplus A$, $C_1 = a \cdot P$, $C_2 = a \cdot Y$, $CID = ID \oplus h_1(C_2)$, and $F = h(ID\|M\|T_1)$. Here, a is a nonce chosen by U from Z_p^* and T_1 is the current timestamp.

(b). The smartcard then sends the login message $m_1 = \{C_1, CID, F, T_1\}$ to S over a public channel.

3.3. Authentication phase

Both the U and S will execute the following operations:

(a). On receiving the login message m_1 from U, S checks whether the timestamp T_1 is valid or not. If T_1 is invalid, S quits the session. Otherwise, S computes $C_2' = s \cdot C_1$, $ID' = CID \oplus h_1(C_2')$, $M' = h(ID \oplus s)$ and $F' = h(ID'\|M'\|T_1)$. Now, S checks whether $F' = F$ holds. If it is invalid, S aborts the session. Otherwise, S authenticates U and proceeds to the next step.

(b). S computes $D_1 = c \cdot P$, $D_2 = c \cdot C_1$, $sk = h(ID'\|h_1(D_2)\|M')$, $G = h(sk\|M'\|T_2)$, where c is a nonce and T_2 is the current timestamp chosen by S. Then, S sends the authentication message $m_2 = \{D_1, G, T_2\}$ to U over a public channel.

(c). On receiving the authentication message m_2 from S, U checks whether T_2 is valid or not. If it is invalid, U aborts the session. Otherwise, U computes $D_2' = a \cdot D_1$, $sk' = h(ID\|h_1(D_2')\|M)$, and $G' = h(sk'\|M\|T_2)$. Now, U checks whether $G' = G$ holds. If it is invalid, U aborts the session, otherwise, authenticates S and accepts sk' as the session key.

The description login and authentication phases of the Xu et al.'s scheme is given in Figure 1.

3.4. Password change phase

In this phase, U freely changes his/her password without connection from the TMIS server S. This phase can be described as follows:

(a). U enters ID and PW, and then the smartcard computes $A = h(PW\|r)$, $M = B \oplus A$.

(b). The smartcard asks U to input a new password PW_{new} and the smartcard computes $A_{new} = h(PW_{new}\|r)$, $B_{new} = A_{new} \oplus M$. Then, the smartcard replaces B with B_{new}.

4. Cryptanalysis of Xu et al.'s scheme

This section identified and analyzed the weaknesses of Xu et al.'s scheme.

User U/Smartcard		**TIMS Server S**
User U: Input ID and PW **Smartcard:** $A = h(PW\|\|r)$ $M = B \oplus A$ $a \in_R Z_p^*$, T_1 $C_1 = a \cdot P$ $C_2 = a \cdot Y$ $CID = ID \oplus h_1(C_2)$ $F = h(ID\|\|M\|\|T_1)$		
	$\xrightarrow{m_1 = \{C_1, CID, F, T_1\}}$ (via a public channel)	
		If T_1 is invalid, abort Else, $C'_2 = s \cdot C_1$ $ID' = CID \oplus h_1(C'_2)$ $M' = h(ID \oplus s)$ $F' = h(ID'\|\|M'\|\|T_1)$ If $(F' = F)$, abort Else, $c \in_R Z_p^*$, T_2 $D_1 = c \cdot P$, T_2 $D_2 = c \cdot C_1$ $sk = h(ID'\|\|h_1(D_2)\|\|M')$ $G = h(sk\|\|M'\|\|T_2)$ $G_2 = H(sk\|\|C_2\|\|T_2\|\|s \cdot P)$
	$\xleftarrow{m_2 = \{D_1, G, T_2\}}$ (via a public channel)	
Smartcard: If T_2 is valid, abort Else, $D'_2 = a \cdot D_1$ $sk' = h(ID\|\|h_1(D'_2)\|\|M)$ $G' = h(sk'\|\|M\|\|T_2)$ If $(G' \neq G)$, abort Else, accept sk as session key		

Figure 1: Login and authentication phases of the Xu et al.'s scheme.

4.1. Xu et al.'s scheme fails to achieve strong authentication in login and authentication phases

In login phase of Xu et al.'s scheme, U enters his/her smartcard into the specific device and keys his/her identity and password in to the smartcard. However, the smartcard does not check whether the inputed password supplied by U is correct. Suppose that U enters his/her password incorrectly by mistake, then both the login and authentication phases still continue in their scheme. At the authentication phase, S will observed the U sends an incorrect login message m_1. This phenomena increases the burden on the communication and computational costs in the login and authentication phases. The detailed description of this attack in Xu et al.'s scheme is given below.

Suppose that U inserts the wrong password PW' instead of the correct password PW. Then the smartcard computes $A = h(PW'||r)$ and

$$\begin{aligned} M &= B \oplus A \\ &= h(s||ID) \oplus h(PW||r) \oplus h(PW'||r) \\ &\neq h(s||ID) \end{aligned}$$

Now the smartcard chooses a nonce $a \in_R Z_p^*$ and a current timestamp T_1, and computes $C_1 = a \cdot P$, $C_2 = a \cdot Y$, $CID = ID \oplus h_1(C_2)$, and

$$\begin{aligned} F &= h(ID\|M\|T_1) \\ &= h(ID\|(h(s\|ID) \oplus h(PW\|r) \oplus h(PW'\|r))\|T_1)) \\ &\neq h(ID\|h(s\|ID)\|T_1) \end{aligned}$$

Then the smartcard sends the login message $m_1 = \{C_1, CID, F, T_1\}$ to S over a public channel. In the authentication phase of Xu et al.'s scheme, the TIMS server S checks that the timestamp T_1 is valid. Now S computes $C_2' = s \cdot C_1$, $ID' = CID \oplus h_1(C_2') = ID$, $M' = h(ID \oplus s)$ and

$$\begin{aligned} F' &= h(ID'\|M'\|T_1) \\ &= h(ID\|h(s\|ID)\|T_1)) \\ &\neq F \end{aligned}$$

Therefore, S confirms that U is an illegal user and thus rejects the login message m_1. However, in practice, U is a legitimate user. Therefore, an efficient authentication scheme should be robust in providing the incorrect password detection at the smartcard's side in the login phase. However, Xu et al.'s scheme does not have such provision.

4.2. Xu et al.'s scheme fails to update the password correctly in the password change phase

Although, Xu et al. proposed a password change phase, which could help U to change his/her old password to the new password without the assistance from the TIMS server S. However, it has been observed that their password change phase has some problem. In this phase, the verification of the correctness of the inputed old password is absent and thus, the change of old password to a new password then take place incorrectly if U inserts his/her old password PW wrongly by mistake. The description of this attack in Xu et al.'s scheme is given below.

In the password change phase, assume that U enters the wrong password PW' by mistake instead of the correct password PW. Then the smartcard computes $A = h(PW'\|r)$ and

$$\begin{aligned} M &= B \oplus A \\ &= h(s\|ID) \oplus h(PW\|r) \oplus h(PW'\|r) \\ &\neq h(s\|ID) \end{aligned}$$

Now the smartcard asked U for a fresh password. If U inputs a new password PW_{new} and the smartcard computes $A_{new} = h(PW_{new}\|r)$ and

$$\begin{aligned} B_{new} &= h(PW_{new}\|r) \oplus h(s\|ID) \oplus h(PW\|r) \oplus h(PW'\|r) \\ &\neq h(PW_{new}\|r) \oplus h(s\|ID) \end{aligned}$$

Then the smartcard replaces B with B_{new}. It is to be observed that, B_{new} is incorrectly updated by smartcard due to wrong old password. As a result, the subsequent login phase, authentication phase and password change phase will be hampered, if U wishes to execute these phases with the new password PW_{new}. This phenomena enters into an unrecoverable situation. The only possibility to overcome this situation is that U can issue a new smartcard with the fresh password and identity according to the registration phase. However, Xu et al.'s scheme also fails to propose a lost/stolen smartcard revocation phase.

4.3. Xu et al's scheme fails to provide the revocation of lost/lost smartcard

In a two-factor authentication, the assumption that the smartcard is non-temper resistance is a realistic assumption and the revocation lost/stolem smartcard is necessary. Otherwise, if the lost/stolen smartcard of an user is acquired by an adversary, then he can get the secret values using the methods proposed in [10, 11, 12]. Based on the knowledge of extracted information and with the help of some other off-line methods, the advrsary can guessed the password of the user. If the adversary finds the password and if the server unable to distinguish the new smartcard from the lost card, the adversary can impersonate the user by using the old stolen/lost smartcard. Thus, the revocation of lost/stolen smartcard is required in an authentication system in order provide the adequate security to system. However, it has been observed that, Xu et al.'s scheme fails to provide such an important feature.

4.4. Xu et al.'s scheme fails to protect the strong replay attack

In the login phase of Xu et al.'s scheme, U sends the login message $m_1 = \{C_1, CID, F, T_1\}$ to S, where $C_1 = a \cdot P$, $C_2 = a \cdot Y$, $CID = ID \oplus h_1(C_2)$, and $F = h(ID||M||T_1)$. The description of this attack in Xu et al.'s scheme is given below.

(a). Assume that a attacker $\mathcal{A}$ sniffing the communication channel and eavesdropped the message m_1 and then replayed it to S within the expected valid time interval ΔT.

(b). On receiving m_1, S verifies that the timestamp T_1 is valid and then S executes other checks according to the Xu et al.'s scheme. It is to be noted that the message $\{C_1, CID, F, T_1\}$ is correctly generated by U withe correct login identity and password and thus, it passes all the verification performed by S. Thus, the adversary $\mathcal{A}$ gets success to login to S on behalf of U with the strong replay attack.

Therefore, we can conclude that Xu et al.'s scheme fails to protect this kind of strong replay attack using the timestamp.

4.5. Xu et al's scheme has the overhead of public key certificate management

It has been noticed that Xu et al.'s scheme, S has the private-public key pair $\{s, Y = s \cdot P\}$, this is a public key algorithm based on public key infrastructure (PKI). Hoverer, PKI requires a certificate authority (CA) to issue a certificate for the authentication of the user's private-public key pair. In addition, in PKI-based system, user must have additional capability to verify the public key certificates of other users. Therefore, to maintain the certificate framework, PKI incurs a nontrivial level of system complexity and implementation costs.

5. The Proposed scheme

5.1. Initialization phase

(a). S selects a security parameter k and a k-bit prime number p. Then S determine the tuple $\{F_p$, E/F_p, $P\}$.

(b). S chooses a number $s \in Z_p^*$ as his/her secret key and an one-way collision-resistant secure hash function $H() : \{0, 1\}^* \rightarrow Z_p^*$.

(c). S publishers the system parameters, $\Omega = \{F_q$, E/F_p, $H()$, P, $p\}$ and keeps s secret.

5.2. Registration phase

(a). U chooses his/her identity ID, password PW and a random number $r \in_R Z_p^*$, and then sends ID and $l = H(PW||r)$ to S through a secure channel.

(b). Upon receiving ID and l, S checks the registration details of U and whether ID is already in the database or not. If ID already exists in the database, S asks U to provide a fresh identity.

(c). S then checks the registration record of U and if U is a new user then S sets $N = 0$, otherwise if U is registering next time in the system, then S sets $N = N + 1$ and stores values (ID, N) in the database.

(d). S chooses a random number $b \in_R Z_p^*$ and computes $\sigma = \frac{(b+s)}{l}$ mod p, $B = b \cdot P$ and $u = H(s \cdot P||l)$. S stores $\{E/F_p$, P, u, B, σ, $H()$, $p\}$ into a smartcard and sends it to U via a secure channel.

(e). After receiving the smartcard, U stores r into it. Finally, the smartcard contains the information $\{E/F_p$, P, u, B, r, σ, $H()$, $p\}$.

5.3. Login phase

(a). U inserts the smartcard into the smart device and inputs his/her ID and PW, and then the smartcard computes $l = H(PW||r)$ and $s \cdot P = (\sigma l) \cdot P - B$. Also the smartcard computes $u^* = H(s \cdot P||l)$ and checks whether $u^* = u$ holds. If it is invalid, the smartcard abort the session, otherwise, proceeds to the next step.

(b). The smartcard also chooses a nonce $a \in_R Z_p^*$, a current timestamp T_1 and computes $C_1 = a \cdot (s \cdot P)$, $CID = ID \oplus H(s \cdot P||T_1)$, $G_1 = H(ID||C_1||T_1||s \cdot P)$. The smartcard then sends the login message $m_1 = \{CID$, C_1, G_1, $T_1\}$ to S over a public channel.

5.4. Authentication phase

(a). On receiving m_1, S checks whether T_1 is valid or not. If T_1 is invalid, S aborts the session. Otherwise, S computes $ID' = CID \oplus H(s \cdot P||T_1)$ and $G_1' = H(ID'||C_1||T_1||s \cdot P)$. Then, S checks whether $G_1' = G_1$ holds. If it is invalid, S aborts the session, otherwise, accepts U as a legal user.

(b). S chooses a nonce $c \in_R Z_p^*$, a current timestamp T_2. Then, S computes $C_2 = c \cdot (s \cdot P)$, the session key $sk = H(ID'||C_1||C_2||k||s \cdot P)$ and $G_2 = H(sk||C_2||T_2||s \cdot P)$, where $k = c \cdot (C_1) = c \cdot a \cdot s \cdot P$. Then, S sends $m_2 = \{C_2$, G_2, $T_2\}$ to U over a public channel. In order to protect the strong replay attack and to facility the lost smartcard revocation, S incorporates the tuple (ID, N, T_1) in the database [13, 14, 15]. If S will receive the next login message, say $m_1' = \{CID'$, C_1', G_1', $T_1'\}$ from U, S rejects the login request if $T_1' = T_1$. If this condition holds, S gets confirmation that it is a replay message with in the valid timestamp ΔT.

(c). On receiving m_2, U checks whether T_2 is valid or not. If it is invalid, U aborts the session, otherwise, computes $k' = a \cdot (C_2) = c \cdot a \cdot s \cdot P$, $sk' = H(ID||C_1||C_2||k'||s \cdot P)$, $G_2' = H(sk'||C_2||T_2||s \cdot P)$ and checks whether $G_2' = G_2$ holds. If it is invalid, U aborts the session, otherwise, authenticats S and accepts sk' as the correct session key.

The description login and authentication phases of the proposed scheme scheme is given in Figure 2.

5.5. Password change phase

(a). U inserts his/her smartcard into the smartcard reader and then enters ID and PW into the smartcard.

(b). The smartcard computes $l = h(PW||r)$, $s \cdot P = (\sigma \cdot l)P - B$ and $u^* = H(s \cdot P||l)$. The smartcard then and checks whether $u^* = u$ holds. If it is invalid, the smartcard abort the password change request. Otherwise, the smartcard asks U for new password.

(c). U chooses a new number $r_{new} \in_R Z_p^*$, a new password PW_{new} and enters them into the smartcard. The smartcard then computes $l_{new} = H(PW_{new}||r_{new})$, $\sigma_{new} = \frac{l\sigma}{l_{new}} = \frac{(s+b)}{l_{new}}$ and $u_{new} = H(s \cdot P||l_{new})$. Then, the smartcard replaces $\{E/F_p, P, u, B, r, \sigma, H(), p\}$ with $\{E/F_p, P, u_{new}, B, r_{new}, \sigma_{new}, H(), p\}$.

5.6. Stolen/lost smartcard revocation phase

In the proposed scheme, if the smartcard of U is lost or stolen, U then requests S for its revocation. S firstly checks the registration credentials of U, e.g. driver's licence card, national identity, date of birth, etc. After checking the credential, S updates N as $N = N + 1$ for the tuple (ID, N, T_1) to revoke the smartcard. In every revocation, N is incremented by one. For each revocation, U is encouraged to use fresh password and random number, otherwise, the adversary, who has the lost smartcard, can masqurade U by using the same credentials previously stored in the lost/stolen smartcard.

6. Security analysis of the proposed scheme

6.1. User anonymity

In the proposed scheme, U's identity is changed in each session and kept secret from the adversary, i.e., U's anonymity is achieved during login phase. In our scheme, an anonymous identity $CID = ID \oplus H(s \cdot P||T_1)$ for the user U is calculated and this identity will changed in each session, since it is generated using the timestamp T_1. Only the TIMS server S can recover original identity ID from CID, however, an adversary cannot do the same.

User U/Smartcard		**TIMS Server S**
User U: Input ID and PW **Smartcard:** $l = H(PW\|\|r)$ $s \cdot P = (\sigma l) \cdot P - B$ $u^* = H(s \cdot P\|\|l)$ If $(u^* \neq u)$, abort Else, $a \in_R Z_p^*, T_1$ $C_1 = a \cdot (s \cdot P)$ $CID = ID \oplus H(s \cdot P\|\|T_1)$ $G_1 = H(ID\|\|C_1\|\|T_1\|\|s \cdot P)$.	$\xrightarrow{m_1 = \{CID, C_1, G_1, T_1\}}$ (via a public channel)	
		If T_1 is invalid, abort Else, $ID' = CID \oplus H(s \cdot P\|\|T_1)$ $G_1' = H(ID'\|\|C_1\|\|T_1\|\|s \cdot P)$ If $(G_1' = G_1)$, abort Else, $c \in_R Z_p^*, T_2$ $C_2 = c \cdot (s \cdot P)$ $k = c \cdot (C_1)$ $sk = H(ID'\|\|C_1\|\|C_2\|\|k\|\|s \cdot P)$ $G_2 = H(sk\|\|C_2\|\|T_2\|\|s \cdot P)$ Store (ID, N, T_1)
	$\xleftarrow{m_2 = \{C_2, G_2, T_2\}}$ (via a public channel)	
Smartcard: If T_2 is valid, abort Else, $k' = a \cdot (C_2)$ $sk' = H(ID\|\|C_1\|\|C_2\|\|k'\|\|s \cdot P)$ $G_2' = H(sk'\|\|C_2\|\|T_2\|\|s \cdot P)$ If $(G_2' \neq G_2)$, abort Else, accept the session key sk'		

Figure 2: Login and authentication phases of the proposed scheme.

6.2. Strong replay attack

The proposed scheme can eliminate the strong replay attack [13, 14, 15]. In the authentication phase, if S receives the next login message, say $m_1' = \{CID', C_1', G_1', T_1'\}$, then S retrieves the tuple (ID, N, T_1) and compares T_1' with T_1. If $T_1' = T_1$, then S rejects m_1' because it simply implies that the received message is a replay one. Otherwise, S updates the tuple (ID, N, T_1) to (ID, N, T_1') in the database.

6.3. Off-line password guessing attack from the lost smartcard

The proposed scheme could protect the off-line password guessing attack from the lost smartcard. Suppose that U's smartcard was stolen and the adversary breaches collects the the secret information $\{E/F_p, P, u, B, r, \sigma, H(), p\}$ from it, where $l = H(PW\|r)$, $\sigma = \frac{(b+s)}{l}$, $B = b \cdot P$ and $u = H(s \cdot P\|l)$. Although, the random number r is revealed, the adversary is still unable to compute U's password PW without the secret key s of the TIMS server S. Hence, the proposed scheme can eliminate this attack.

6.4. Mutual authentication

In the proposed scheme, the mutual authentication between U and S is achieved in order to avoid the user's impersonation attack and server's spoofing attack. In our scheme, S first validates U's message $m_1 = \{CID, C_1, G_1, T_1\}$, by checking whether the timestamp T_1 and the condition $G_1' = G_1$ are valid. On the other hand, U validates S through the verification of the timestamp T_2 and the condition $G_2' = G_2$ hold.

6.5. *Session key agreement*

In the proposed scheme, the common and secret session key agreement during the authentication phase is also provided between U and S. A session key $sk = H(ID||C_1||C_2||k||s \cdot P)$, where $k = c \cdot a \cdot s \cdot P$ is shared between U and S. It is to noted that the session key sk will be different for each session and cannot be replayed or reused after the expiration of session as it is depended on C_1, C_2 and k. Thus, both of U and S can transfer some confidential message through the encryption process using the session key sk.

6.6. *Session key forward secrecy*

In the proposed scheme, even if the secret key s of S is compromised, an adversary cannot compute the session key $sk = H(ID||C_1||C_2||k||s \cdot P)$, where $k = c \cdot a \cdot s \cdot P$, from the public messages $m_1 = \{CID, C_1, G_1, T_1\}$ and $m_2 = \{C_2, G_2, T_2\}$. Since the adversary cannot compute k from the pair $(C_1, C_2) = (a \cdot s \cdot P, c \cdot s \cdot P)$ due to the difficulties of solving the CDH problem. Thus, the proposed scheme provides the forward secrecy of the session key.

6.7. *Privileged-insider attack*

In real environment, user generally uses the common login identity and password for his/her convenience and accesses a number of applications provided by different servers. Note that if the privileged-insider of the TIMS server S has obtains the plaintext password of U, then of course he may try to masquerade U by accessing other servers where U resisters by the same login identity and password. However, in the proposed scheme, U registers to S with ID and $l = H(PW||r)$ instead of plaintext password PW. In addition, the random number r is kept secret from the privileged-insider of S, therefore, he cannot apply the offline procedure on l to get PW as the probability of guessing or r is $\frac{1}{p}$, which very small. As a result, the privileged-insider attack is hard in the proposed scheme.

6.8. *Unknown-key share attack*

In the unknown key-share attack, U finishes the session by believing that he/she shares the session key sk correctly with S, however, S mistakenly believes that sk is instead shared with an adversary. In the proposed scheme, S authenticates U by validating the time stamp T_1 and the condition $G'_1 = G_1$, respectively. Then S commutes the session key as $sk = H(ID||C_1||C_2||k||s \cdot P)$, where $k = c \cdot a \cdot s \cdot P$ and the authentication value $G_2 = H(sk||C_2||T_2||s \cdot P)$, and sends the authentication message $m_2 = \{C_2, G_2, T_2\}$ to U. On receiving the message m_2, U computes sk and $G'_2 = H(sk||C_2||T_2||s \cdot P)$. U authenticates S and accepts sk as the correct session key if $G'_2 = G_2$ holds, otherwise, abort the session. Thus, the proposed scheme resists the the unknown key-share attack.

6.9. *Known-key attack*

The known-key attack includes that the authentication scheme should give the ability to the the user and server to agree on a common and unique secret session key in each session. If any of the session key is compromised, however, other session keys should be secured. In the proposed scheme, due to the one-way property of the hash function H and the randomness of the nonce a and c, the session key sk differs in every session. Therefore, the adversary has no ability to compromise none of the previous and further session keys from the disclosed session key. As a result, the proposed scheme can protect known-key attack.

7. Conclusion

In this paper, the Xu et al.'s two-factor mutual authentication and key agreement scheme is shown to be inefficient for practical use. We have proved that (1) it fails to achieve strong authentication in login and authentication phases; (2) it fails to update the password correctly in the password change phase; (3) it fails to provide the revocation of lost/lost smartcard; (4) it fails to protect the strong replay attack; and (5) it has the overhead of public key certificate management. In order to manage the problem of Xu et al.'s scheme, an improved scheme is proposed with better security features.

References

[1] He, D., Chen, J., and Zhang, R., A more secure authentication scheme for telecare medicine information systems. Journal of Medical Systems 36(3): 1989-1995, 2012.

[2] Wei, J., Hu, X., and Liu,W., An improved authentication scheme for telecare medicine information systems. J. Med. Syst. 36(6): 3597-3604, 2012.

[3] Li, S. H., Wang, C. Y., Lu,W. H., Lin, Y. Y., and Yen, D. C., Design and implementation of a telecare information platform. Journal of Medical Systems 36(3):1629-1650, 2012.

[4] Chen, H. M., Lo, J. W., and Yeh, C. K., An efficient and secure dynamic ID-based authentication scheme for telecare medical information systems. Journal of Medical Systems 36(6): 3907-3915, 2012.

[5] Wu, Z. Y., Lee, Y. C., Lai, F., Lee, H. C., and Chung, Y., A secure authentication scheme for telecare medicine information systems. Journal of Medical Systems 36(3):1529-1535, 2012.

[6] Kumari, S., and Khan, M. K., Kumar, R., Cryptanalysis and Improvement of 'A Privacy Enhanced Scheme for Telecare Medical Information Systems'. Journal of Medical Systems 37: 9952-9962, 2013.

[7] Das, A., and Goswami, A., A Secure and Efficient Uniqueness-and-Anonymity-Preserving Remote User Authentication Scheme for Connected Health Care. Journal of Medical Systems 37: 9948-9963, 2013.

[8] Xu, X., Zhu, P., Wen, Q., Jin, Z., Zhang, H., He, L., A Secure and Efficient Authentication and Key Agreement Scheme Based on ECC for Telecare Medicine Information Systems. Journal of Medical Systems 38:9994, 2014.

[9] Islam, SK. H., and Biswas, G. P., A more efficient and secure IDbased remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem. Journal of Systems and Software 84: 1892-1898, 2011.

[10] Messerges, T. S., Dabbish, E. A., and Sloan, R. H., Examining smartcard security under the threat of power analysis attacks. IEEE Transactions on Computers 51(5): 541-552, 2002.

[11] Joye, M., and Olivier, F., Side-channel analysis, Encyclopedia of Cryptography and Security. Kluwer Academic Publishers, pp. 571-576, 2005.

[12] Kocher, P., Jaffe, J., and Jun, B., Differential power analysis. In: Proceedings of Advances in Cryptology (Crypto'99), LNCS, pp. 388-397, 1999.

[13] Das, A. K., Analysis and improvement on an efficient biometric-based remote user authentication scheme using smart cards. IET Information Security 5(3): 145-151, 2011.

[14] Das, A. K., Goswami. A., An Improved and Effective Secure Password-Based Authentication and Key Agreement Scheme Using Smart Cards for the Telecare Medicine Information System. Journal of Medical Systems 37(3): 1-16, 2013.

[15] Das, A. K., and Bruhadeshwar, B., An Improved and Effective Secure Password-Based Authentication and Key Agreement Scheme Using Smart Cards for the Telecare Medicine Information System. Journal of Medical Systems, 37(5), 1-17, 2013.