J Med Syst (2015) 39: 76
DOI 10.1007/s10916-015-0259-6

On the Security of a Two-Factor Authentication and Key Agreement Scheme for Telecare Medicine Information Systems

Hamed Arshad[1] · Vahid Teymoori[1] · Morteza Nikooghadam[1] · Hassan Abbassi[1]

Received: 31 January 2015 / Accepted: 2 June 2015 / Published online: 18 June 2015

Abstract Telecare medicine information systems (TMISs) aim to deliver appropriate healthcare services in an efficient and secure manner to patients. A secure mechanism for authentication and key agreement is required to provide proper security in these systems. Recently, Bin Muhaya demonstrated some security weaknesses of Zhu's authentication and key agreement scheme and proposed a security enhanced authentication and key agreement scheme for TMISs. However, we show that Bin Muhaya's scheme is vulnerable to off-line password guessing attacks and does not provide perfect forward secrecy. Furthermore, in order to overcome the mentioned weaknesses, we propose a new two-factor anonymous authentication and key agreement scheme using the elliptic curve cryptosystem. Security and performance analyses demonstrate that the proposed scheme not only overcomes the weaknesses of Bin Muhaya's scheme, but also is about 2.73 times faster than Bin Muhaya's scheme.

Keywords Authentication · Key agreement · Telecare Medicine Information System (TMIS) · Anonymous · Elliptic Curve Cryptosystem (ECC) · Security

This article is part of the Topical Collection on *Systems-Level Quality Improvement*

✉ Hamed Arshad
hamedarshad@imamreza.ac.ir; hamedarshad@aol.com

✉ Morteza Nikooghadam
morteza.nikooghadam@gmail.com;
m.nikooghadam@imamreza.ac.ir

Vahid Teymoori
vahidteymoori@yahoo.com

Hassan Abbassi
abbaasi@gmail.com

[1] Department of Computer Engineering and Information Technology, Imam Reza International University, Mashhad, Iran

Introduction

Nowadays, with the rapid development of information and communication technologies, telecare medicine information systems (TMISs) are widely used to provide healthcare services remotely. By using TMISs, patients (especially in hard-to-reach places and rural areas) can stay at their home and obtain healthcare services at the right time and lower cost. Patients can send their body parameters which indicate their health condition to medical servers and receive a proper treatment from doctors [1, 2]. These systems not only reduces patients expenses and problems, but also can save precious resources in hospitals, such as veteran doctors, beds, medical devices and so on. Furthermore, since hospitals and healthcare providers can share their stored patients medical records via the internet, repeated medical examinations are not needed and doctors can rapidly diagnose diseases and prescribe appropriate treatments [3].

In TMISs, medical servers maintain patients electronic medical records such as personal information, health records, and physiological parameters (e.g., blood pressure, heart rate, etc.) [4–6]. Since these data are sensitive, access to medical servers should be controlled to prevent unauthorized accesses and preserve patients' privacy [7–9]. Furthermore, the security (e.g., confidentiality, integrity, and authenticity) of data that are exchanged between users (e.g., patients and doctors) and medical

servers should be provided because these data are the basis of medical decisions, and any modification of them may cause a substantial injury to the patients [7, 10]. Access to medical servers' resources can be controlled by an authentication process and security of data that are exchanged can be provided by encrypting/authenticating them with the keys that are negotiated during a key agreement process [2, 7, 11, 12]. Therefore, a secure authentication and key agreement scheme is a proper solution to provide security and privacy in TMISs. Until now, many authentication and key agreement schemes have been proposed to provide security in TMISs.

In 2012, Wu et al. [13] proposed an authentication scheme based on the discrete logarithm problem (DLP) for TMISs. However, He et al. in [14] demonstrated that Wu et al.'s scheme [13] is insecure against the privileged insider and impersonation attacks. In order to enhance the security of Wu et al.'s scheme [13], He et al. proposed an improved DLP-based authentication scheme for TMISs [14]. Since the scheme proposed by He et al. [14] required fewer exponentiation operations than Wu et al.'s scheme [13], it was more efficient than Wu et al.'s scheme [13]. Nevertheless, Wei et al. [15] showed that both Wu et al.'s scheme [13] and He et al.'s scheme [14] are vulnerable to off-line password guessing attacks. Furthermore, in order to improve the security, Wei et al. suggested their improved DLP-based authentication scheme for TMISs [15]. Unfortunately, Zhu in [16] demonstrated that Wei et al.'s scheme [15] similar to the previous schemes is vulnerable to off-line password guessing attacks. In addition, in order to enhance the security of Wei et al.'s scheme [15], Zhu [16] proposed a new authentication scheme for TMISs.

In 2013, Khan et al. [17] demonstrated that Zhu's scheme [16] is vulnerable to online password guessing attacks and does not provide key agreement. In order to overcome the weaknesses of Zhu's scheme [16], Khan et al. proposed an improved authentication scheme for TMISs [17]. Furthermore, Lee et al. in [18] demonstrated that Zhu's scheme [16] is insecure against parallel attacks. Lee et al. [18] also proposed an improved authentication scheme and claimed that their scheme could withstand various attacks. Nevertheless, Das et al. [19] showed that since in the password change phase of Lee et al.'s scheme [18] the smart card does not verify the inputted old password, if a user mistakenly enters a wrong old password, then he/she will no longer be able to login to the medical server. Therefore, the password change process of Lee et al.'s scheme [18] does not work properly and this can lead to denial-of-services attacks [19].

In 2014, Bin Muhaya [20] demonstrated that Zhu's scheme [16] is insecure against user impersonation attacks, off-line password guessing attacks and denial-of-service attacks. Bin Muhaya [20] also claimed that Khan et al.'s scheme [17], which is an improvement of Zhu's scheme [16], is vulnerable to user impersonation attacks and denial-of-services attacks and also does not provide user anonymity. In order to improve the security of the previous schemes, Bin Muhaya [20] proposed an improved authentication and key agreement scheme for TMISs. However, in this paper, it is demonstrated that Bin Muhaya's scheme [20] is also vulnerable to off-line password guessing attacks and does not provide perfect forward secrecy that is an important security requirement for security protocols [21]. Furthermore, in order to overcome the weaknesses of Bin Muhaya's scheme [20], a new two-factor authentication and key agreement scheme based on the elliptic curve discrete logarithm problem (ECDLP) is proposed. The proposed scheme not only could overcome the weaknesses of Bin Muhaya's scheme, but also has better performance compared to previous schemes.

The rest of the paper is organized as follows. Section "Review of Bin Muhaya's scheme" provides a brief review of Bin Muhaya's scheme. Section "Weaknesses of Bin Muhaya's scheme" presents the security weaknesses of Bin Muhaya's scheme. Our improved authentication and key agreement scheme is described in Section "The peoposed scheme". Sections "Security analysis" and "Performance analysis" analyze the security and performance of the proposed scheme. Finally, Section "Conclusion" concludes the paper.

Review of Bin Muhaya's scheme

In this section, we will briefly review Bin Muhaya's authentication scheme [20] which consists of four phases namely initialization phase, registration phase, login-authentication cum session key agreement phase, and password change phase. Definition of notations used in Bin Muhaya's scheme is summarized in Table 1. The registration and login-authentication cum session key agreement phases of Bin Muhaya's scheme are illustrated in Fig. 1.

Table 1 Notations used in Bin Muhaya's scheme

Symbol	Definition
p, q	Two prime numbers
d	The medical server's secret key
(e, n)	The medical server's public key
ID_i	The user's identity
PW_i	The user's password
$\|$	The concatenation operation
S_{key}	A shared session key
$\oplus$	The exclusive-or operation (XOR)

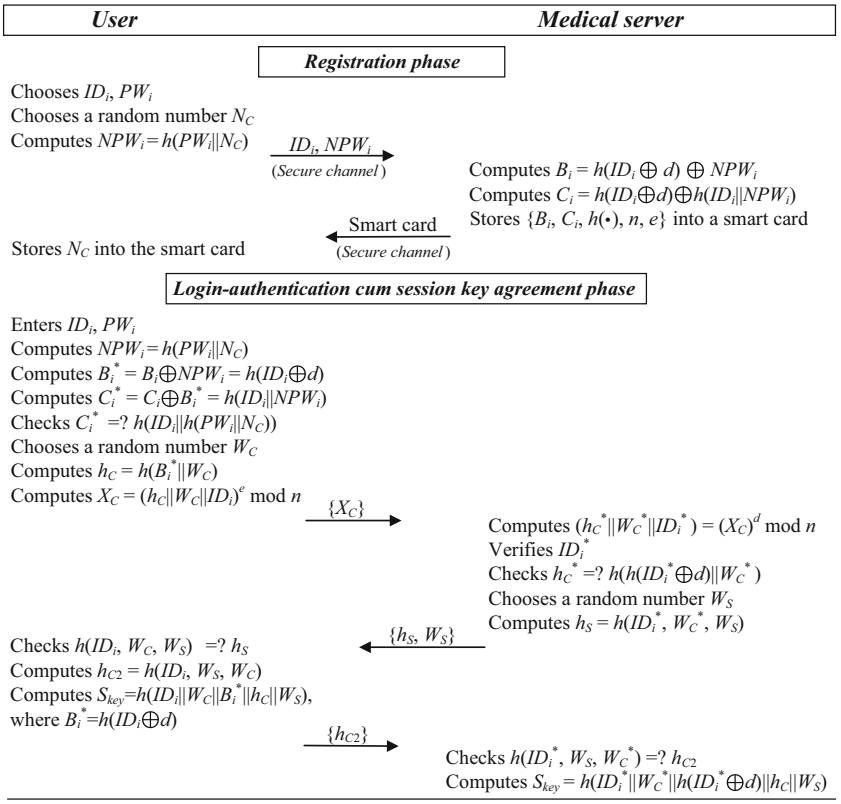

Fig. 1 Bin Muhaya's scheme

Initialization phase

In this phase, the medical server chooses two prime numbers p and q and computes $n = pq$. Then, the medical server chooses a secure one-way hash function $h(\cdot) : \{0, 1\}^* \rightarrow Z_q{}^*$, and two integers e and d such that $(de \bmod (p-1)(q-1)) = 1$. Finally, the medical server keeps d as its secret key and publishes (e, n) as its public key.

Registration phase

The user registration process in Bin Muhaya's scheme [20] involves the following steps.

Step 1: The user chooses his/her identity ID_i and password PW_i, and generates a random number N_C. Then, the user computes $NPW_i = h(PW_i \parallel N_C)$ and sends a registration request message $\{ID_i, NPW_i\}$ to the medical server through a secure channel.

Step 2: After receiving the message $\{ID_i, NPW_i\}$ from the user, the medical server computes $B_i = h(ID_i \oplus d) \oplus NPW_i$ and $C_i = h(ID_i \oplus d) \oplus h(ID_i \parallel NPW_i)$. Then, the medical server stores $\{h(\cdot), e, n, B_i, C_i\}$ in a smart card and issues the smart card to the user via the secure channel.

Step 3: After receiving the smart card, the user stores N_C in the memory of the smart card.

Login-authentication cum session key agreement phase

In this phase, the user and the medical server authenticate each other and negotiate a session key as follows.

Step 1: The user inserts his/her smart card into a card reader and enters his/her ID_i and PW_i. Then, the smart card computes $NPW_i = h(PW_i \parallel N_C)$, $B_i^* = B_i \oplus NPW_i$, and $C_i^* = C_i \oplus B_i^*$ and checks whether C_i^* is equal to $h(ID_i \parallel NPW_i)$ or not. If they are not equal, it stops the process. Otherwise, it selects a random number W_C, computes $h_C = h(B_i^* \parallel W_C)$ and $X_C = (h_C \parallel W_C \parallel ID_i)^e \bmod n$, and sends a login request message $\{X_C\}$ to the medical server through a public channel.

Step 2: On receiving the message $\{X_C\}$, the medical server decrypts X_C as $(X_C)^d \bmod n = (h_C^* \parallel W_C^* \parallel ID_i^*)$, and checks whether ID_i^* is a valid identity or not. If it is not a valid identity, the medical server terminates the session. Otherwise, it checks whether h_C^* is equal to $h(h(ID_i^* \oplus d) \parallel W_C^*)$ or not. If they are not equal, the medical server terminates the session. Otherwise, it selects a random number W_S, computes $h_S = h(ID_i^*, W_C^*, W_S)$, and sends a message $\{h_S, W_S\}$ to the user through the public channel.

Step 3: Upon receiving the message $\{h_S, W_S\}$, the user checks whether h_S is equal to $h(ID_i, W_C, W_S)$ or not. If they are not equal, the user halts the process. Otherwise, the user authenticates the medical server, computes $h_{C2} = h(ID_i, W_S, W_C)$, and sends a response message $\{h_{C2}\}$ to the medical server through the public channel. Furthermore, the user computes the shared session key S_{key} as $S_{key} = h(ID_i \| W_C \| B_i \oplus h(PW_i \| N_C) \| h_C \| W_S)$.

Step 4: After receiving the message $\{h_{C2}\}$ from the user, the medical server checks whether the received h_{C2} is equal to $h(ID_i^*, W_S, W_C^*)$ or not. If they are equal, the medical server authenticates the user, accepts the login request, and computes the shared session key S_{key} as $S_{key} = h(ID_i^* \| W_C^* \| h(ID_i^* \oplus d) \| h_C \| W_S)$.

Password change phase

When a user decides to change his/her current password PW_i, he/she inserts his/her smart card into the card reader and enters his/her ID_i and PW_i, and also a new password $PW_i{}^{New}$. Then, the smart card computes $NPW_i = h(PW_i \| N_C)$, $B_i^* = B_i \oplus NPW_i$, and $C_i^* = C_i \oplus B_i^*$ and checks whether C_i^* is equal to $h(ID_i \| NPW_i)$ or not. If they are equal, the smart card computes $NPW_i{}^{New} = h(PW_i{}^{New} \| N_C)$, $B_i{}^{New} = B_i \oplus NPW_i \oplus NPW_i{}^{New}$, and $C_i{}^{New} = C_i \oplus h(ID_i \| NPW_i) \oplus h(ID_i \| NPW_i{}^{New})$ and replaces B_i and C_i with $B_i{}^{New}$ and $C_i{}^{New}$, respectively.

Weaknesses of Bin Muhaya's scheme

This section demonstrates that Bin Muhaya's scheme [20] is vulnerable to off-line password guessing attacks and does not provide perfect forward secrecy.

Off-line password guessing attack

If an adversary steals or finds a user's smart card, he/she is able to guess its password as follows:

Step 1: The adversary retrieves $\{h(\cdot), e, n, B_i, C_i, N_C\}$ from the memory of the smart card by using the methods proposed in [22, 23], where $B_i = h(ID_i \oplus d) \oplus NPW_i$, $C_i = h(ID_i \oplus d) \oplus h(ID_i \| NPW_i)$, and $NPW_i = h(PW_i \| N_C)$.

Step 2: The adversary selects a pair (ID_i^*, PW_i^*) from the Cartesian product $D_{ID} \times D_{PW}$, where D_{ID} and D_{PW} denote the identity space and the password space, respectively.

Step 3: The adversary computes $NPW_i^* = h(PW_i^* \| N_C)$, $B_i^* = B_i \oplus NPW_i^*$, and $C_i^* = C_i \oplus B_i^*$ and checks whether C_i^* is equal to $h(ID_i^* \| NPW_i^*)$ or not. If they are equal, it implies that he/she has selected the right pair (ID_i^*, PW_i^*); otherwise, he/she repeats Steps 2 and 3 until he/she succeeds.

Since the user's identity and password have low entropy, an adversary can enumerate all pairs (ID_i, PW_i) in the Cartesian product $D_{ID} \times D_{PW}$ within polynomial time [24–28]. Therefore, the presented attack is feasible and Bin Muhaya's scheme [20] is vulnerable to off-line password guessing attacks.

Lack of perfect forward secrecy

Suppose an adversary has eavesdropped and recorded the previously transmitted messages $\{X_C\}$ and $\{h_S, W_S\}$. If the adversary somehow obtains the medical server's secret key, d, he/she is able to compute the session key of each previous communication session as follows:

Step 1: The adversary decrypts X_C with the obtained secret key d as $(X_C)^d \bmod n = (h_C \| W_C \| ID_i)$.

Step 2: Then, the adversary computes the session key S_{key} as $S_{key} = h(ID_i \| W_C \| h(ID_i \oplus d) \| h_C \| W_S)$.

Therefore, since disclosure of the medical server's secret key leads to compromising the previously established session keys, it can be said that the perfect forward secrecy is not supported in Bin Muhaya's scheme [20].

The peoposed scheme

In order to overcome the weaknesses of Bin Muhaya's scheme [20], a new two-factor user anonymity preserving authentication and key agreement scheme for TMISs is proposed in this section. The proposed scheme includes four phases: system setup phase, registration phase, authentication phase, and password change phase. The definition of notations used in the proposed scheme is summarized in Table 2 and the phases are described in the following subsections.

System setup phase

In this phase, which runs once at the system initialization time, the medical server chooses an elliptic curve E [29] and selects a point P with the large order n over the elliptic curve as the base point. Then, the medical server selects a

Table 2 Notations used in the proposed scheme

Symbol	Definition
E	An elliptic curve with order n
P	The base point of the elliptic curve E
x	The secret key of the medical server
Y	The public key of the medical server, where $Y = xP$
ID_i	The identity of the user
PW_i	The password of the user
MID_i	The masked identity of the user
T_1, T_2	Two timestamps
ΔT	The maximum transmission delay
SK	A shared session key
$\|$	The concatenation operation
$\oplus$	The exclusive-or operation (XOR)

random integer $x \in_R Z_p{}^*$ as its secret key and computes its public key $Y = xP$. Moreover, the medical server chooses a secure one-way hash function $h(\cdot) : \{0,1\}^* \rightarrow \{0,1\}^l$, where l is the length of the output. Finally, the medical server publishes $(E, n, P, Y, h(\cdot))$ as system parameters and keeps x securely.

Registration phase

Figure 2 shows the registration process of the proposed scheme. In this phase, which runs once for each user, the following steps are performed over a secure channel to register the user with the medical server.

Step 1: The user chooses an identity ID_i and sends a registration request message $\{ID_i\}$ to the medical server.

Step 2: Upon receiving the registration request message $\{ID_i\}$, the medical server checks whether ID_i exists in its database or not. If it does not exist, the medical server selects a random number N_S and computes the user's masked identity $MID_i = h(ID_i \| N_S)$ and the user's authenticator $A_i = h(ID_i \| x \| MID_i)$. Finally, the medical server stores $\{MID_i, ID_i\}$ in its database and $\{A_i, E, MID_i, n, P, Y, h(\cdot)\}$ in a smart card and then sends the smart card to the user.

Step 3: Upon receiving the smart card, the user chooses a password PW_i, computes $B_i = A_i \oplus h(ID_i \| PW_i)$, and replaces A_i with B_i in the smart card. Finally, the smart card contains $\{B_i, E, MID_i, n, P, Y, h(\cdot)\}$.

Authentication phase

Figure 3 shows the mutual authentication and key agreement process of the proposed scheme. In this phase, which runs frequently, the following steps are performed over a public channel.

Step 1: The user inserts his/her smart card into a card reader and enters his/her ID_i and PW_i. Then, the smart card selects a random integer $d_C \in_R Z_p{}^*$ and computes $A_i = B_i \oplus h(ID_i \| PW_i)$, $Q_C = d_C P$, $V_1 = h(ID_i \| A_i \| Q_C \| T_1)$, $K_1 = d_C Y = d_C(xP)$, $V_1{}' = V_1 \oplus h(K_1 \| T_1)$, and $MID_i{}' = MID_i \oplus h(K_1)$, where T_1 is the current timestamp and $A_i = h(ID_i \| x \| MID_i)$. At last, it sends a login request message $\{MID_i{}', V_1{}', Q_C, T_1\}$ to the medical server.

Step 2: Upon receiving the message $\{MID_i{}', V_1{}', Q_C, T_1\}$, the medical server checks the validity of the timestamp T_1 by checking the condition $T_2 - T_1? \leq \Delta T$, where T_2 is the current timestamp and ΔT denotes the predetermined maximum transmission delay. If the condition does not hold, the medical server terminates the session. Otherwise, the medical server computes $K_1^* = xQ_C = x(d_C P)$, $V_1 = V_1{}' \oplus h(K_1^* \| T_1)$, and $MID_i =$

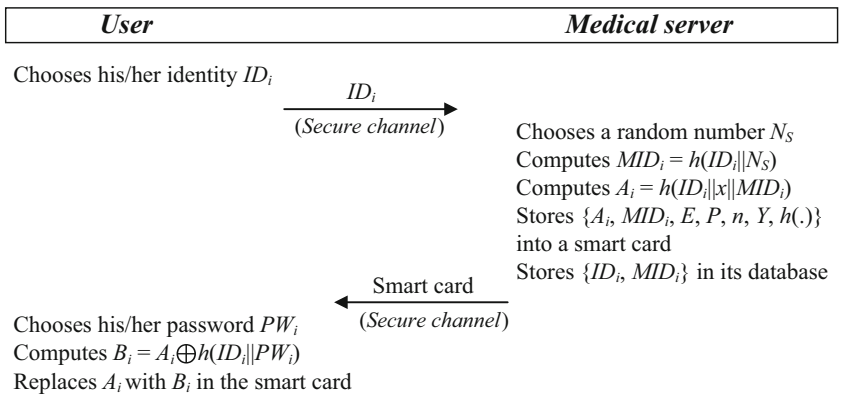

Fig. 2 Rgistration phase of the proposed scheme

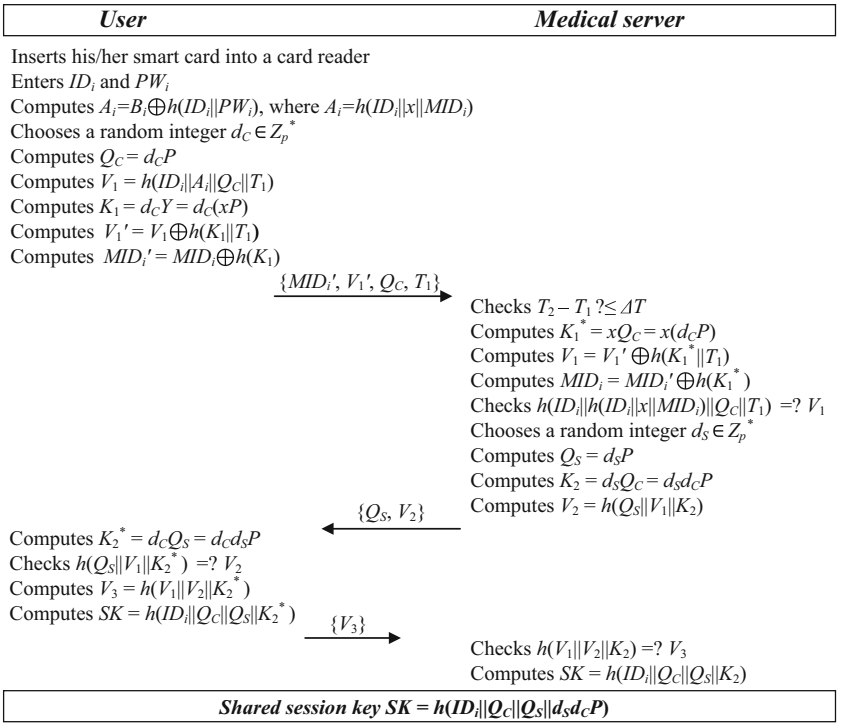

Fig. 3 Authentication phase of the proposed scheme

$MID_i' \oplus h(K_1^*)$, retrieves ID_i from its database based on the masked identity MID_i, and checks whether $h(ID_i \parallel h(ID_i \parallel x \parallel MID_i) \parallel Q_C \parallel T_1)$ is equal to V_1 or not. If they are not equal, the medical server terminates the session. Otherwise, the medical server selects a random integer $d_S \in_R Z_p^*$, computes $Q_S = d_S P$, $K_2 = d_S Q_C = d_S d_C P$, and $V_2 = h(Q_S \parallel V_1 \parallel K_2)$, and sends a challenge message $\{Q_S, V_2\}$ to the user.

Step 3: After receiving the message $\{Q_S, V_2\}$ from the medical server, the user computes $K_2^* = d_C Q_S = d_C d_S P$ and checks whether $h(Q_S \parallel V_1 \parallel K_2^*)$ is equal to the received V_2 or not. If they are not equal, the user stops the process. Otherwise, he/she authenticates the medical server, computes $V_3 = h(V_1 \parallel V_2 \parallel K_2^*)$ and sends a response message $\{V_3\}$ to the medical server. Finally, the user computes the shared session key SK as $SK = h(ID_i \parallel Q_C \parallel Q_S \parallel K_2^*)$.

Step 4: Upon receiving the message $\{V_3\}$, the medical server checks whether $h(V_1 \parallel V_2 \parallel K_2)$ is equal to the received V_3 or not. If they are not equal, the medical server terminates the session. Otherwise, the medical server authenticates the user and computes the shared session key SK as $SK = h(ID_i \parallel Q_C \parallel Q_S \parallel K_2)$.

Password change phase

When a user wants to change his/her password, he/she inserts his/her smart card into the card reader and enters his/her identity ID_i and password PW_i, and a new password PW_i^{New}.

Step 1: This step is the same as Step 1 in Section "Authentication phase".

Step 2: This step is the same as Step 2 in Section "Authentication phase".

Step 3: On receiving the message $\{Q_S, V_2\}$, the smart card computes $K_2^* = d_C Q_S = d_C d_S P$ and checks whether $h(Q_S \parallel V_1 \parallel K_2^*)$ is equal to the received V_2 or not. If they are not equal, the smart card stops the process. Otherwise, it computes $B_i^{New} = h(ID_i \parallel PW_i^{New}) \oplus h(ID_i \parallel PW_i) \oplus B_i = h(ID_i \parallel PW_i^{New}) \oplus h(ID_i \parallel PW_i) \oplus h(ID_i \parallel PW_i) \oplus A_i = h(ID_i \parallel PW_i^{New}) \oplus A_i$ and then replaces B_i with B_i^{New}.

Security analysis

In this section, it is demonstrated that the proposed scheme not only could withstand replay attacks, password guessing attacks, server impersonation attacks, user impersonation attacks, man-in-the-middle attacks, privileged insider attacks, and modification attacks, but also could provide

perfect forward secrecy, known-key security, and user anonymity.

Replay attacks

In the proposed scheme, if an adversary sends an old login request message $\{MID_i', V_1', Q_C, T_1\}$ to the medical server, the medical server can detect a replay attack by checking the condition $T_2 - T_1? \leq \Delta T$, where T_2 is the time when the medical server receives the login request message and ΔT denotes the maximum transmission delay. The adversary may replay the eavesdropped login request message $\{MID_i', V_1', Q_C, T_1\}$ within the expected valid time interval. However, since the adversary does not know d_C, he/she cannot produce a valid response message $\{V_3\}$ after receiving the medical server's challenge message $\{Q_S, V_2\}$, where $V_3 = h(V_1' \oplus h(d_C Y \parallel T_1) \parallel V_2 \parallel d_C Q_S)$. Therefore, the proposed scheme is secure against replay attacks.

Password guessing attacks

Suppose an adversary steals or finds a user's smart card. The adversary may retrieve $\{B_i, MID_i\}$ from the memory of the smart card, guess a pair (ID_i^*, PW_i^*), and compute $h(ID_i^* \parallel PW_i^*)$. However, since the adversary does not know the medical server's secret key, x, he/she is not able to check the correctness of his/her guessed pair (ID_i^*, PW_i^*) as $h(ID_i^* \parallel PW_i^*) =? B_i \oplus h(ID_i^* \parallel x \parallel MID_i)$. Even if the adversary has all the previously transmitted login and authentication messages, he/she is still not able to check the correctness of his/her guessed ID_i^* and PW_i^*. Because, he/she cannot relate the stolen smart card with its corresponding login and authentication messages. Therefore, the proposed scheme can resist the off-line password guessing attacks.

It is well known that online password guessing attacks can be defeated by limiting the number of continuous failed login requests. For example, the medical server blocks the user account for a certain amount of time (e.g. 15 minutes) after a certain amount of continuous failed login requests (e.g. three times) within a certain amount of time (e.g. one hour). In order to hold the account accessible for the legal user, the system can be designed in a way that the legal user can reactivate his/her blocked account using a Private Unblocking Key (PUK). In addition, it can be implemented in a way that instead of blocking the user account (after a certain amount of continuous failed login requests), the medical server asks some security questions in addition to the password. Furthermore, CAPTCHA [30] can also be used to prevent automated attacks. The system can also be designed in a way that upon entering the right password by the user, the medical server sends a code to the user through another communication channel; the user is then requested to enter the code to complete the authentication process.

Server impersonation attacks

In the proposed scheme, if an adversary wants to impersonate a legal medical server, he/she has to compute a proper verification message V_2 that corresponds to the user's login request message $\{MID_i', V_1', Q_C, T_1\}$. Since the adversary does not know the medical server's secret key, x, he/she is not able to derive V_1 from V_1' as $V_1 = V_1' \oplus h(x Q_C \parallel T_1)$. Hence, the adversary cannot compute a proper value V_2 as $V_2 = h(Q_S \parallel V_1 \parallel K_2)$. Therefore, since the adversary cannot produce a proper challenge message $\{Q_S, V_2\}$, he/she fails to impersonate the medical server.

User impersonation attacks

In the proposed scheme, if an adversary wants to impersonate a legal user, he/she has to forge the messages $\{MID_i', V_1', Q_C, T_1\}$ and $\{V_3\}$. Even if the adversary steals the user's smart card and retrieves $\{B_i, E, MID_i, n, P, Y, h(\cdot)\}$ from the smart card's memory, since he/she does not know PW_i and ID_i, he/she is not able to compute the user's authenticator A_i as $A_i = B_i \oplus h(ID_i \parallel PW_i)$. Hence, the adversary cannot compute $V_1 = h(ID_i \parallel A_i \parallel Q_C \parallel T_1)$ and thus cannot produce a valid login request message $\{MID_i', V_1', Q_C, T_1\}$, where $V_1' = V_1 \oplus h(d_C Y \parallel T_1)$. Therefore, the proposed scheme could withstand user impersonation attacks.

Man-in-the-middle attacks

Since the proposed scheme is secure against the user and server impersonation attacks, the mutual authentication is provided in our proposed scheme and man-in-the-middle attacks cannot succeed in our scheme.

Privileged insider attacks

Since in the registration phase of the proposed scheme, the user only submits his/her identity ID_i to the medical server and does not send his/her password, the privileged user of the medical server has no way to obtain the user's password. Therefore, the proposed scheme is immune from privileged insider attacks.

Modification attacks

In the proposed scheme, an adversary is not able to modify the login and authentication messages, because the user and the medical server can detect any unauthorized modification by verifying the verification messages V_1, V_2, and V_3. If an

Table 3 Performance comparison

Comparison criteria			Scheme			
			Zhu [16]	Khan et al. [17]	Bin Muhaya [20]	The proposed
Computational cost	Registration phase	Cost	$2h_t + 2x_t$	$4h_t + 1me_t$	$3h_t + 3x_t$	$3h_t + 1x_t$
		Time	$1ms$	$524ms$	$1.5ms$	$1.5ms$
	Login and authentication phase	Cost	$2me_t + 8h_t + 2x_t$	$5me_t + 10h_t$	$2me_t + 12h_t + 3x_t$	$6pm_t + 12h_t + 5x_t$
		Time	$1048ms$	$2615ms$	$1050ms$	$384.45ms$
Security properties	Resist password guessing attacks		No	Yes	No	Yes
	Resist denial-of-services attacks		No	No	Yes	Yes
	Resist user impersonation attacks		No	No	Yes	Yes
	Resist server impersonation attacks		Yes	Yes	Yes	Yes
	Resist man-in-the-middle attacks		Yes	Yes	Yes	Yes
	Provide perfect forward secrecy		N/A	Yes	No	Yes
	Provide user anonymity		No	No	Yes	Yes
	Provide mutual authentication		Yes	Yes	Yes	Yes
	Provide key agreement		No	Yes	Yes	Yes
	Provide known-key security		N/A	Yes	Yes	Yes

N/A: Not Available or Not Applicable

adversary wants to modify the login and authentication messages, he/she has to compute a proper verification message V_1, V_2, or V_3. However, since the values of $A_i = h(ID_i \parallel x \parallel MID_i)$ and $K_2 = d_C d_S P$ are required to compute the verification messages V_1, V_2, and V_3 and the adversary does not know them, the adversary is not able to compute a proper verification message V_1, V_2, or V_3 for his/her modified messages. Therefore, the user and the medical server can detect any unauthorized modification by verifying V_1, V_2, and V_3.

Perfect forward secrecy

In the proposed scheme, $SK = h(ID_i \parallel Q_C \parallel Q_S \parallel d_C d_S P)$ is a shared session key between the user and the medical server. Even if an adversary obtains the medical server's secret key, x, or the user's password, PW_i, he/she is still not able to compute old session keys, because without knowing d_C or d_S it is impossible to compute $d_C d_S P$. It should be noted that due to the hardness of ECDLP [29] the adversary is not able to derive d_C and d_S from Q_C and Q_S, respectively. Therefore, the perfect forward secrecy is supported in the proposed scheme.

Known-key security

Due to the randomness of d_C and d_S in the proposed scheme, the produced session key in each session is different and independent of other session keys. Therefore, knowing a session key does not help an adversary to compute other session keys. Hence, it can be said that the known-key security is supported in the proposed scheme.

User anonymity

In the authentication phase of the proposed scheme, the user sends $MID_i' = MID_i \oplus h(d_C Y) = h(ID_i \parallel N_S) \oplus h(x Q_C)$ instead of his/her real identity ID_i to the medical server. Hence, since the adversary does not know the medical server's secret key, x, and the random number N_S, he/she is not able to obtain the user's real identity ID_i. Therefore, the proposed scheme provides user anonymity.

Performance analysis

In this section, the efficiency of the proposed scheme is evaluated and the proposed scheme is compared with Bin Muhaya's scheme [20] and the related schemes [16, 17] in terms of the computational costs, running times, and security properties. For the computation cost evaluation, the following notations are employed.

- me_t: the time complexity of performing a modular exponentiation.
- pm_t: the time complexity of performing an elliptic curve point multiplication.
- h_t: the time complexity of a hash function operation.
- x_t: the time complexity of a bit-wise exclusive-or (XOR) operation.

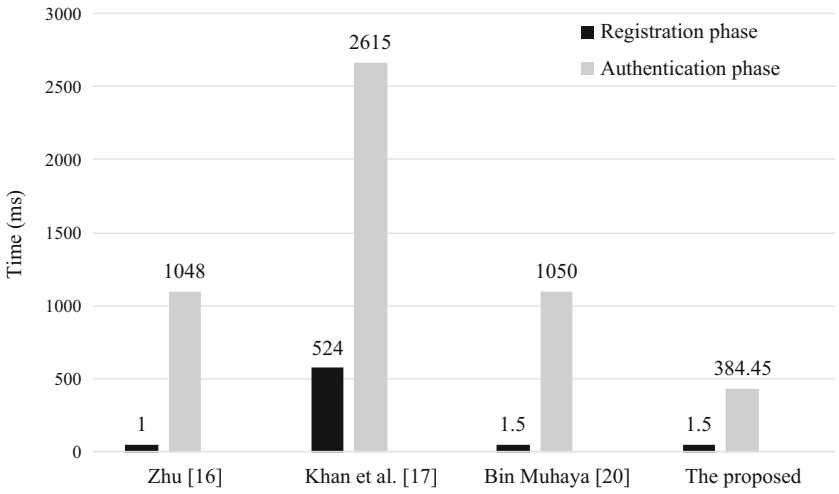

Fig. 4 Running times of different schemes

The proposed scheme in the registration phase requires three hash function operations and one exclusive-or operation; therefore, the computational cost of the registration phase is $3h_t + 1x_t$. During the authentication phase of the proposed scheme, six elliptic curve point multiplication operations, twelve hash function operations, and five exclusive-or operations are required. Therefore, the computational cost of the authentication phase of the proposed scheme is $6pm_t + 12h_t + 5x_t$.

In order to provide a precise performance comparison, the experiment data reported in[31, 32] are used. As reported in [31, 32] the average execution time of a modular exponentiation, an elliptic curve point multiplication and a hash function operation is 522 ms, 63.075 ms, and 0.5 ms, respectively. Moreover, it is assumed that the time of performing an exclusive-or operation is negligible. Therefore, the running time of the registration and authentication phases of the proposed scheme is 1.5 ms and 384.45 ms, respectively. Table 3 summarizes the comparisons among the proposed scheme and the related schemes [16, 17, 20] in terms of the computational costs and security properties. Furthermore, the proposed scheme and the related schemes [16, 17, 20] are compared in Fig. 4 in terms of their running times in the registration and authentication phases.

According to Table 3, the proposed scheme in the authentication phase is about 2.72 times faster than the schemes of Bin Muhaya [20] and Zhu [16]. Furthermore, not only the proposed scheme in the authentication phase is about 6.8 times faster than Khan et al.'s scheme [17], but also Khan et al.'s scheme in the registration phase is about 349.3 times slower than the proposed scheme. Moreover, Zhu's scheme [16] is vulnerable to password guessing attacks, parallel attacks, and user impersonation attacks and also does not provide key agreement [17, 18, 20]. Khan et al.'s scheme [17] is vulnerable to user impersonation attacks and denial-of-services attacks and also does not provide user anonymity [20]. Furthermore, Bin Muhaya's scheme [20] is vulnerable to off-line password guessing attacks and does not provide perfect forward secrecy. The proposed scheme has a better performance than the related schemes because the security of the proposed scheme is based on the elliptic curve discrete logarithm problem (ECDLP). Since ECDLP is significantly more difficult than the integer factorization problem and the discrete logarithm problem (DLP) [33], the elliptic curve cryptosystems need shorter keys than the other asymmetric cryptosystems to achieve the same security level. For instance, a 160-bit ECC key is as secure as a 1024-bit RSA key. Obviously, this means ECC has the advantages of higher speed and lower power consumption [34]. Hence, the proposed scheme is more efficient than the previous schemes. Therefore, since the proposed scheme provides more security and efficiency than the previous schemes, the proposed scheme is more suitable for TMISs.

Conclusion

In this paper, we have shown that Bin Muhaya's authentication and key agreement scheme for telecare medicine information systems (TMISs) is insecure against off-line password guessing attacks and does not support perfect forward secrecy. In order to improve Bin Muhaya's scheme, we have proposed an ECC-based anonymous authentication and key agreement scheme for TMISs. Detailed analyses confirm that the proposed scheme is more secure and efficient than the previous schemes. Therefore, the proposed scheme is an eligible authentication and key agreement scheme for TMISs.

References

1. Lin, T.H., and Lee, T.F., Secure verifier-based three-party authentication schemes without server public keys for data exchange in telecare medicine information systems. *J. Med. Syst.* 38(5):1–9, 2014.
2. Arshad, H., and Nikooghadam, M., Three-Factor anonymous authentication and key agreement scheme for telecare medicine information systems. *J. Med. Syst.* 38(12):1–12, 2014.
3. Xie, Q., Zhang, J., Dong, N., Robust anonymous authentication scheme for telecare medical information systems. *J. Med. Syst.* 37(2):1–8, 2013.
4. Wu, F., and Xu, L., Security analysis and improvement of a privacy authentication scheme for telecare medical information systems. *J. Med. Syst.* 37(4):1–9, 2013.
5. Mishra, D., Srinivas, J., Mukhopadhyay, S., A secure and efficient chaotic map-based authenticated key agreement scheme for telecare medicine information systems. *J. Med. Syst.* 38(10):1–10, 2014.
6. Mishra, D., Mukhopadhyay, S., Kumari, S., Khan, M.K., Chaturvedi, A., Security enhancement of a biometric based authentication scheme for telecare medicine information systems with nonce. *J. Med. Syst.* 38(5):1–11, 2014.
7. Kim, K.W., and Lee, J.D., On the security of two remote user authentication schemes for telecare medical information systems. *J. Med. Syst.* 38(5):1–11, 2014.
8. Mishra, D., Mukhopadhyay, S., Chaturvedi, A., Kumari, S., Khan, M.K., Cryptanalysis and improvement of Yan et al.'s biometric-based authentication scheme for telecare medicine information systems. *J. Med. Syst.* 38(6):1–12, 2014.
9. Mishra, D., Understanding security failures of two authentication and key agreement schemes for telecare medicine information systems. *J. Med. Syst.* 39(3):1–8, 2015.
10. Mishra, D., On the security flaws in id-based password authentication schemes for telecare medical information systems. *J. Med. Syst.* 39(1):1–16, 2015.
11. Mishra, D., A study on ID?based authentication schemes for telecare medical information system, arXiv:1311.0151, 2013.
12. He, D., Kumar, N., Chilamkurti, N., Lee, J.H., Lightweight ECC based RFID authentication integrated with an ID verifier transfer protocol. *J. Med. Syst.* 38(10):1–6, 2014.
13. Wu, Z.Y., Lee, Y.C., Lai, F., Lee, H.C., Chung, Y., A secure authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(3):1529–1535, 2012.
14. He, D., Chen, j., Zhang, R., A more secure authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(3):1989–1995, 2012.
15. Wei, J., Hu, X., Liu, W., An improved authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(6):3597–3604, 2012.
16. Zhu, Z., An efficient authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(6):3833–3838, 2012.
17. Khan, M.K., and Kumari, S., An authentication scheme for secure access to healthcare services. *J. Med. Syst.* 37(4):1–12, 2013.
18. Lee, T.F., and Liu, C.M., A secure smart-card based authentication and key agreement scheme for telecare medicine information systems. *J. Med. Syst.*, 2013. doi:10.1007/s10916-013-9933-8.
19. 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. *J. Med. Syst.* 37(5):1–17, 2013.
20. Bin Muhaya, F.T., Cryptanalysis and security enhancement of Zhu's authentication scheme for telecare medicine information system. *Security and Communication Networks* 8:149–158, 2015. doi:10.1002/sec.967.
21. Arshad, H., and Nikooghadam, M., An efficient and secure authentication and key agreement scheme for session protocol using ECC. *Multimedia Tools and Applications*, 2014. doi:10.1007/s11042-014-2282-x.
22. Kocher, P., Jaffe, J., Jun, B., Differential power analysis.In: *Proceedings of Advances in Cryptology*, Vol. 1666, pp. 788–797. Santa Barbara, 1999.
23. 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.
24. Wang, D., and 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.
25. Ma, C.-G., Wang, D., Zhao, S.-D., Security flaws in two improved remote user authentication schemes using smart cards. *Int. J. Commun. Syst.* 27:2215–2227, 2014. doi:10.1002/dac.2468.
26. Klein, D.V. Foiling the cracker: a survey of, and improvements to, password security. In: *Proceedings of the 2nd USENIX Security Workshop*. Anaheim, 1990.
27. Bonneau, J. The science of guessing: Analyzing an anonymized corpus of 70 million passwords. In: *33th IEEE Symposium on Security and Privacy (S&P 2012), IEEE Computer Society*, pp. 538–552. San Francisco, 2012.
28. Islam, S.H., Design and analysis of an improved smartcard-based remote user password authentication scheme. *Int. J. Commun. Syst.*, 2014. doi:10.1002/dac.2793.
29. Hankerson, D., Menezes, A., Vanstone, S. *Guide to elliptic curve cryptography*. New York: Springer, 2004.
30. Von Ahn, L., Blum, M., Langford, J., Telling humans and computers apart automatically. *Commun. ACM* 47(2):56–60, 2004.
31. Jiang, Q., Ma, J., Li, G., Yang, l., An Efficient Ticket Based Authentication Protocol with unlinkability for wireless access networks. *Wirel. Pers. Commun.* 77(2):1489–1506, 2014.
32. Hsieh, W.-B., and Leu, J.-S., Anonymous authentication protocol based on elliptic curve DiffieHellman for wireless access networks. *Wirel. Commun. Mob. Comput.* 14:995–1006, 2014. doi:10.1002/wcm.2252.
33. Vanstone, S.A., Elliptic curve cryptosystem-the answer to strong, fast public-key cryptography for securing constrained environments. *Inf. Secur. Tech. Rep.* 12:78–87, 1997.
34. Stallings, W. *Cryptography and Network Security: Principles and Practice*. 4th edition. Upper Saddle River: Prentice Hall, 2005.