Journal of Medical Systems (2019) 43: 321
https://doi.org/10.1007/s10916-019-1442-y

MOBILE & WIRELESS HEALTH

Authenticated Key Agreement Scheme with Strong Anonymity for Multi-Server Environment in TMIS

Hui Qiao[1] · **Xuewen Dong**[1] · **Yulong Shen**[1]

Received: 27 April 2019 / Accepted: 28 August 2019 / Published online: 7 October 2019

Abstract

The technology of Internet of Things (IoT) has appealed to both professionals and the general public to its convenience and flexibility. As a crucial application of IoT, telecare medicine information system (TMIS) provides people a high quality of life and advanced level of medical service. In TMIS, smart card-based authenticated key agreement schemes for multi-server architectures have gathered momentum and positive impetus due to the conventional bound of a single server. However, we demonstrate that most of the protocols in the literatures can not implement strong security features in TMIS, such as Lee et al.'s and Shu's scheme. They store the identity information directly, which fail to provide strong anonymity and suffer from password guessing attack. Then we propose an extended authenticated key agreement scheme (short for AKAS) with strong anonymity for multi-server environment in TMIS, by enhancing the security of the correlation parameters stored in the smart cards and calculating patients' dynamic identities. Furthermore, the proposed chaotic map-based scheme provides privacy protection and is formally proved under Burrows-Abadi-Needham (BAN) logic. At the same, the informal security analysis attests that the AKAS scheme not only could resist the multifarious security attacks but also improve efficiency by 21% compared with Lee et al.'s and Shu's scheme.

Keywords Authenticated key agreement · Strong anonymity · Multi-server · Chaotic map · Dynamic identity

Introduction

The Internet of Things (IoT), also known as the Internet of Everything or the Industrial Internet, is a novel technology pattern conceived as a global network of machines and devices having the capability to interact with each other, which is taken for one of the most significant areas of the advanced technology in the future and is gaining the extensive concern from a wide range of industries [1]. IoT brings a good many applications [2–4], such as intelligent transportation, smart home, telecare medicine information system (TMIS), and so on. It is worth noting that TMIS as an emerging network can provide remote monitoring, remote medical treatment, and emergency medical assistance.

TMIS is becoming quite popular research topic, however, the medical information of the patient is highly sensitive, thus, malicious access could lead to terrifying security and privacy issues, which are barriers in real-time applications of TMIS. With the swift expansion of communication technology and the increasing number of hand-held devices, it becomes necessary to authenticate the legal status of users and control access to network resources. In 1981, the first password-based remote user authentication scheme for single-server environment is presented by Lamport [5]. However, password-based authenticated key agreement schemes are easy to forget and vulnerable to online/off-line password guessing attacks. Besides, in a conventional single server authenticated key agreement scheme, one server is answerable to render services to all the remote users who have registered in TMIS. If a certain patient hopes to access medical services from the different servers, he or she has to register as a legal user with these servers respectively, which is quite tedious and cumbersome and adds the costs simultaneously [6]. As a resultful and flexible method, a

This article is part of the Topical Collection on *Mobile & Wireless Health*

✉ Yulong Shen
ylshen@mail.xidian.edu.cn

Hui Qiao
qiaohui1007300405@163.com

Xuewen Dong
xwdong@xidian.edu.cn

[1] Xidian University, Xi'an, China

multi-server architecture is generally adopted. In a multi-server environment, patients just need to register once in the registry server and associate with all other application servers. In the past decade, researchers have proposed a variety of authentication protocols for multi-server environment in TMIS [7–23]. At the same time, it has been noted that the majority of the authenticated key agreement schemes still are not completely free from security attacks, especially, strong anonymity [20–23].

In this work, we study the patient privacy problem over patient medical information. Our goal is to develop a strong anonymity solution that is more secure and efficient. To this end, we propose an extended authenticated key agreement scheme(AKAS) with strong anonymity for the multi-server environment in TMIS, which combines the symmetric encryption/decryption and Chebyshev chaotic map operations. To improve our scheme's performance, we employ the lightweight security mechanism, chaotic map operation, to finish the mutual authentication between patient and server, while replacing other cryptographic mechanisms, which are costly.

Several big challenges should be solved before the TMIS system could be adopted and deployed far and wide by the public. We observe that protecting the patient's identity ID_i and password PW_i from disclosure, and verifying the legitimacy of both users and servers are key operations during the process of communication in TMIS. During these processes, the information regarding the patient's identity ID_i and password PW_i should not be revealed, because some confidential information, such as the medical record and diagnostic report could be leaked with identity information disclosed. Although some works are tackling these problems, their schemes are not satisfactory, since strong anonymity problem and password guessing attacks have remained. That means that if another legitimate server intercepts the communication, it can obtain the identity information of the patient. In this paper, our proposed scheme aims to not only could withstand the various security attacks but also support strong anonymity. That is to say, any third party, even other legitimate users and servers, cannot obtain the relevant identity information of two communication parties in the multi-server environment.

TMIS system architecture

The TMIS system architecture has been depicted minutely in Fig. 1, in which diverse entities are involved, like registration center (RC), patient (U_i), and server (S_j). The RC renders services of registration to patients and distributes smart cards to those registered patients. Almost simultaneously it also registers the other servers. The patients submit their healthcare data to a telecare server via wired/wireless terminals at their home expediently. After receiving the medical records of a certain patient, the doctors perform the diagnosis and then employ the final and best medical treatments for patients through the Internet. With this method, TMIS system is equipped to break the obstacles of location and time.

Our contributions

By addressing the challenges that TMIS faced with, in this paper, we present an authenticated key agreement scheme with strong anonymity for multi-server environment. And the contributions of our paper are listed as follows:

(1) We have examined two excellent schemes in [22, 23] respectively and demonstrated that they stored the patient identity information in the smart card directly, could not solve the strong anonymity problem of patients and withstand password guessing attack.
(2) In combination with symmetric encryption/decryption and Chebyshev chaotic map operations, the major

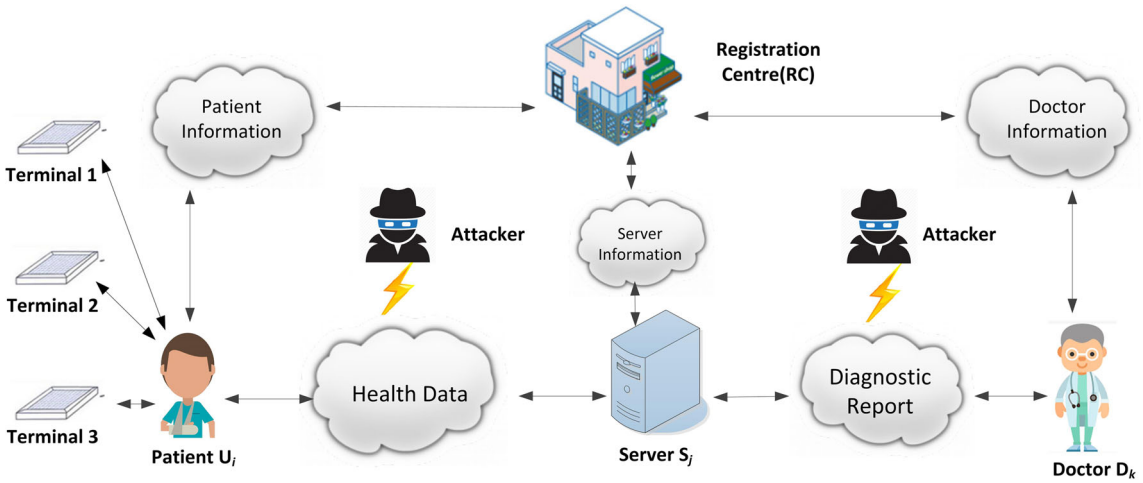

Fig. 1 TMIS system architecture

highlight of this paper is the design and analysis of a lightweight and robust authenticated key agreement scheme for multi-server environment in TMIS.

(3) We have attested that AKAS scheme can not only resist a variety of attacks, such as password guessing attack, but also achieve all outstanding security features such as user anonymity, mutual authentication and so on. Further, it deals with the problem of the strong anonymity of patients in TMIS.

(4) In order to testify the security feature of the mutual authentication, we have utilized Burrows-Abadi-Needham (BAN) logic model.

(5) We have certified that the performance of the AKAS scheme is better in terms of computation and communication costs. In addition, compared with Lee et al.'s scheme, the AKAS scheme reduces Chebyshev chaotic map operations twice and compared with Shu's scheme, the AKAS scheme reduces symmetric encryption/decryption operations twice, Chebyshev chaotic map operations twice, and improves efficiency by 21%.

Attack model

Since the authentication protocol is implemented in the public and insecure channel, the adversary has several advantages or capabilities. The authenticated key agreement scheme in TMIS should satisfy the following assumptions:

(1) An attacker *Eve* can detect energy consumption by using reverse engineering techniques to extract the patient private information which is stored in the smart card [24, 25]. Besides, the attacker *Eve* can eavesdrop on all messages exchanged in the public channel. At the same time, the malicious attacker *Eve* can not intercept messages transmitted in the secure channel.

(2) According to [26], an attacker *Eve* has capability of guessing identity ID and low entropy password PW of the registered patient individually easily but guessing two secret parameters (e.g. ID, PW) is computationally infeasible in polynomial time.

(3) An attacker *Eve* can resend, reroute, modify, delete the messages which derive from patients and remote servers in the TMIS system.

(4) An attacker *Eve* may be a legitimate but malicious user or server in the TMIS system.

The remainder of this work is organized as follows. In Section "Related works", we take a review of the related works, and Section "Preliminaries" deals with the preliminaries, briefly introducing the basic concepts. Section "Analysis of Lee et al.'s and Shu's scheme" relates to the analysis of Lee et al's and Shu's scheme, whereas Section "AKAS scheme" presents the AKAS scheme in detail. Section "Security analysis" depicts the security analysis, which includes security properties and formal analysis under BAN logic. Subsequently, the performance is evaluated through extensive simulations in Section "Performance analysis". Concluding remarks are found in the last Section.

Related works

In this section, we provide a summary of mutually authenticated key agreement protocols for a multi-server environment. In 2009, Liao et al. [13] put forward a dynamic identity-based remote user authenticated key agreement scheme in the multi-server environment. The same year, Hsiang et al. [11] revealed that the scheme proposed by Liao & Wang was found vulnerable to spoofing attacks on servers and registration centers, and impersonation attacks, then they come up with an improved mutual authentication scheme without verification table. In 2011, Sood et al. [17] demonstrated that Hsiang et al.'s scheme also easily suffered from a series of attacks such as impersonation attacks, smart card theft attacks, and replay attacks. Then, they repaired security flaws in Hsiang's protocol and presented a protocol that had the different trust levels between the two servers. In 2012, a dynamic identity-based authentication protocol was proposed by Li et al. [12], who declared that their scheme can overcome the above security attacks. However, Xue et al. [18] certified that protocols such as Li et al.'s protocol still can not withstand security attacks such as impersonation attacks, eavesdropping attacks and denial of service attacks. Then they designed a dynamic pseudo-identity based mutual authentication and key agreement scheme that did not depend on the verification table in the multi-server environment. Wan et al. [19] argued that Li et al.'s protocol could not resist the session key leak attack and the user impersonation attacks, and proposed a novel scheme to overcome the drawbacks that Li et al.'s scheme exists.

In 2012, Tsaur et al. [20] proposed an efficient authentication protocol in the multi-server environment, which did not rely on clock synchronization to resist replay attacks. Li et al. [21]showed that Tsaur et al.'s scheme was insecure against impersonation attack and multi-user login attack, and proposed a simple and improved scheme. At the same time, they declared that their scheme could withstand the multi-user login attack and impersonation attack. In 2014, Lee et al. [22] pointed out that both Tsaur et al.'s and Li et al.'s scheme failed to withstand privileged-insider attacks, and two protocols also failed to achieve the perfect forward secrecy and user anonymity. Therefore, they presented an improved protocol and claimed that their proposed new scheme achieves user anonymity and the

perfect forward secrecy. In 2016, Shu [23] disclosed that Lee et al.'s scheme failed to provide strong anonymity. If other legitimate servers intercept the communication information, they will get the remote users' identity information. In order to surmount the weaknesses, Shu proposed a key agreement scheme based on the extended chaotic map. However, by analyzing Lee et al.'s and Shu's scheme, we find that both Lee et al.'s and Shu's scheme stored the patients' identity information in smart cards directly, still could not satisfy the strong anonymity of remote users and withstand the password guessing attack.

Preliminaries

We briefly introduce some background knowledge in this section, including Chebyshev polynomial and intractability problems.

Chebyshev polynomial

Firstly, we review some basic concepts about Chebyshev polynomial briefly. For more details, please refer to [27].

Definition 1 Chebyshev Polynomial:

Let n be an integer, and x be a variable taking value over the interval $x \in [-1, 1]$, then the Chebyshev polynomial $T_n(x)$: $[-1, 1] \rightarrow [-1, 1]$ above is defined as follows:

$$T_n(x) = \cos(n \arccos(x)) \tag{1}$$

A recurrent relation can be used for defining Chebyshev polynomial map $T_n : R \rightarrow R$ of degree n, by specifying the following equation.

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x) \tag{2}$$

Given $n \geq 2$, $T_0(x) = 1$, and $T_1(x) = x$.

The first few Chebyshev polynomials are listed as below:

$$T_2(x) = 2x^2 - 1 \tag{3}$$
$$T_3(x) = 4x^3 - 3x \tag{4}$$
$$T_4(x) = 8x^4 - 8x^2 + 1 \tag{5}$$

Definition 2 Semi-group Feature:

The semi-group feature of Chebyshev polynomial can be defined on an interval $(-\infty, +\infty)$ as below:

$$T_r(T_t(x)) = T_{rt}(x) = T_{tr}(x) = T_t(T_r(x)) \tag{6}$$

Work by [27] proved that the semi-group feature also apply to the Chebyshev polynomial defined in the interval $(-\infty, +\infty)$.

Definition 3 Extended Chebyshev Polynomial:

We presume n as an integer, and x as a variable with the interval $(-\infty, +\infty)$, then the extended Chebyshev polynomial is defined as follows:

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x) (\text{mod } p) \tag{7}$$

Given $n \geq 2$, $T_0(x) = 1$, $T_1(x) = x$, and p is a big prime. Obviously, the semi-group feature also apply to the extended Chebyshev polynomial.

$$T_r(T_t(x)) = T_{rt}(x) = T_{tr}(x) = T_t(T_r(x)) (\text{mod } p) \tag{8}$$

Intractability problems

Definition 4 Chaotic Map Discrete Logarithm Problem (CMDLP): Given $\langle x, T_u(x)\rangle$, it is computationally infeasible to find an proper integer u such that $T_u(x) = y$.

Definition 5 Chaotic Map Computational Diffie-Hellman Problem (CMCDHP): Given $\langle x, T_u(x), T_v(x)\rangle$, it is computationally infeasible to compute $T_{uv}(x) = y$.

Analysis of Lee et al.'s and Shu's scheme

In this section, similar to Lee et al.'s scheme in user anonymity, Shu's scheme and several others [20–23] will not be introduced in the following again, and we take a brief review of Lee et al.'s excellent multi-server authentication scheme. Simultaneously, we will analyze both Lee et al.'s and Shu's scheme in detail. Two phases comprising Lee et al.'s scheme are the registration phase, and the login and authentication phase. Some of the notations utilized in this paper are summarized in Table 1. The registration centre RC first selects a random number X and randomly selects two number r, s as the master key of RC. It then, calculates $\omega = h(r \parallel s)$ and sends message ω to the server S_j via a secure channel.

Registration phase

This phase is performed in the registration centre RC. Firstly the patient must get registered in the RC as a legitimate user to gain the servers' service rendered by $S = \{S_1, S_2, \cdots, S_n\}$. It performs the following steps for registration.

Step R1: The patient first selects its unique identity ID_i, secret password PW_i and generates a random number N. It then sends a registration request message $(ID_i, h(PW_i) \oplus N_i)$ to the registration centre RC.

Step R2: Registration centre RC calculates $v_i = h(ID_i \parallel P_i \parallel \omega)$, $u_i = v_i \oplus h(PW)_i \oplus N$ and stores $\{ID_i,$

Table 1 Notation description

Symbol	Description
U_i	Users in TMIS
S_j	Telecare servers in TMIS
Eve	Attacker
ID_i	The identity of the patient U_i
PW_i	The password of the patient U_i
RC	Registration Centre
SC	Smart Card
$Sym.Enc_{(mk)}(M)$	Symmetric key encryption algorithm using mk
$Sym.Dec_{(mk)}(M)$	Symmetric key decryption algorithm using mk
$h(\cdot)$	Secure one-way hash function
$\oplus$	Exclusive-or operation
$\|$	String concatenation operation
$T_s(x)$	Chebyshev chaotic map operation

$u_i, P_i, RPUB, X, h(), p\}$ in smart card and sends it to the patient U_i. Given P_i is a valid period of service for users.

Step R3: After receiving the smart card SC, the patient U_i computes $u_i' = u_i \oplus N$ and u_i will be replaced with u_i'. Finally, the patient U_i stores $\{ID_i, u_i', P_i, RPUB, X, h(), p\}$ in the smart card SC.

Login and authentication phase

In the login phase, the patient U_i performs the following steps:

Step L1: The patient U_i inserts its smart card into a specific card reader and then inputs its unique identity ID_i and its secret password PW_i.

Step L2: The smart card SC first computes $v_i = u_i' \oplus h(PW_i)$, chooses a random number ri, and then calculates $C_1 \equiv T_{ri}(X) \bmod p$, $C_2 \equiv T_{ri}(R) \bmod p$, $UID_i = ID_i \oplus h(C_1 \| C_2)$,and $M_{ij} = h(ID_i \| UID_i \| P_i \| v_i \| C_1 \| C_2)$.

Step L3: The patient U_i sends the login request message $M_1 = \{M_{ij}, UID_i, C_1, P_i\}$ to the telecare server S_j via an insecure and public channel.

In order to satisfy mutual authentication and key agreement between two entities, the patient and the server S_j, when the server S_j received the message M_1 , the following steps are performed:

Step A1: the server S_j verifies $h(ID_i' \| UID_i \| P_i \| v_i' \| C_1 \| C_2')? = M_{ij}$. First, the server S_j calculates $C_2' \equiv T_\omega(C_1) \bmod p$, decrypts the message UID_i from the patient U_i to retrieves $ID_i' = UID_i \oplus h(C_1 \| C_2')$ by using C_2' above. Check the validity of P_i, if we prove that P_i is invalid, the server stop providing the service. Otherwise, it calculates $v_i{}' = h(ID_i{}' \| P_i \| \omega)$, and verifies whether $h(ID_i' \| UID_i \| P_i \| v_i' \| C_1 \| C_2')? = M_{ij}$. If not equal, server S_j terminates the session, otherwise, the server S_j updates the service period P_i with $P_i^{new} = P_i - 1$, and then computes the new secrete information $V_i = v_i' \oplus v_i^{new}$ to protect the parameter v_i^{new} and generates an random integer rj. By utilizing the new random number rj, the server S calculates $C_3 \equiv T_{rj}(X) \bmod p$, and the session key $SK \equiv T_{rj}(C_1) \equiv T_{rjri}(X) \bmod p$. Then, the telecare server computes $M_{ji} = h(ID_i' \| v_i' \| v_i^{new} \| P_i^{new} \| C_2' \| C_3 \| SK)$. Finally, the server S_j sends the message $M_2 = \{M_{ji}, C_3, V_i\}$ to the patient through the public and insecure channel.

Step A2: After receiving the message M_2 sent by the server S_j, the smart card SC checks whether the equation $h(ID_i \| v_i \| v_i^{new\prime} \| P_i^{new} \| C_2 \| C_3 \| SK')? = M_{ji}$ holds or not. First, the smart card computes $v_i^{new\prime} = V_i \oplus v_i$, $P_i^{new} = P_i - 1$, $SK' \equiv T_{ri}(C_3) \equiv T_{rirj}(X) \bmod p$. If the aforesaid equation holds, the smart card calculates $u_i^{new} = v_i^{new\prime} \oplus h(PW_i)$ and replaces $\{u_i', P_i\}$ with $\{u_i^{new}, P_i^{new}\}$; otherwise, this session then will be stopped. Finally, the smart card calculates $M_{sk} = h(C_2 \| SK')$ and transmits message $M_3 = \{M_{sk}\}$ to the server S_j via the public and insecure channel.

Step A3: After receiving the message M_3, the server S_j examines the correctness of the session key SK by confirming if the equation $h(C_2' \| SK)? = M_{sk}$ holds. If not equal, the server S_j terminates the session. Otherwise, both S_j and U_i can make full use of SK for ensuring safety of the subsequent session.

Security analysis of Lee et al.'s and Shu's scheme

In this subsection, the security analysis of Lee et al.'s and Shu's scheme will be carried out. Lee et al. claimed that their scheme can provide user anonymity, as well as Shu pointed out that his scheme satisfies strong anonymity. However, after analyzing the Lee et al.'s and Shu's scheme, we find that they fail to provide user anonymity and resist password guessing attack. And we make a concrete analysis as follows:

The identity of the patient cannot be stored in a smart card directly

If the attacker launches the side-channel attack, all the information may be extracted from the smart card. At the registration phase in Lee et al.'s and Shu's Scheme, the identity of each patient is stored in the smart card directly. However, the security of patient identity information is so important that patient identity is limited and immutable. So

the identity information can not be stored directly in a smart card.

Both schemes fail to provide the users identity anonymity

Lee et al. argued that their scheme meets user anonymity in [22], and in [23] Shu claimed that his scheme provides strong anonymity. That is to say, in addition to meeting the anonymity which required in a single-server environment, in a session, it should satisfy that the other servers will not be able to obtain the identity information of both parties in this session. In order to protect patients' privacy, their identities must be anonymous.

However, in the registration phase, obviously, all parameters stored in the smart card SC include $\{ID_i, u_i', P_i, RPUB, X, h(), p\}$ and $\{ID_i, U_{ij}', P_{ij}, x, T_{\omega_j}(x), h(), P\}$, in light of this, the registration centre RC stores the identity information in the smart card SC directly, and the attacker Eve can extract the secret information from the smart card SC easily. Finally, an attacker can obtain patient identity information. Therefore, both Lee et al.'s and Shu's scheme fails to provide user anonymity.

Both schemes cannot resist password guessing attack with the smart card

In this subsection, we attest that Lee et al.'s and Shu's scheme suffer from password guessing attack respectively.

In Lee et al.'s scheme, we assume that an attacker Eve can compromise the secret information $\{ID_i, u_i', P_i, RPUB, X, h(), p\}$ stored in the smart card and then tries to guess the patients password. The attacker Eve can compute $v_i = h(ID_i \parallel P_i \parallel \omega)$ easily. Since the secret information $u_i' = u_i \oplus N$ and $u_i = v_i \oplus h(PW_i) \oplus N$, the attacker Eve can deduce $u_i' = v_i \oplus h(PW_i)$, get u_i' from smart card and v_i above, then the attacker can get the correct password of the patient U_i.

In Shu's scheme, for the same reason we analysis above, we assume that an attacker Eve can compromise the secrete information $\{ID_i, U_{ij}', P_{ij}, x, T_{\omega_j}(x), h(), P\}$ stored in the smart card and then tries to guess the patients password. The attacker can compute $V_{ij} = h(ID_i \parallel P_{ij} \parallel \omega_j)$ easily. Since the secret information $U_{ij}' = U_{ij} \oplus N_i$ and $U_{ij} = V_{ij} \oplus PW_i \oplus N_i$, the attacker Eve can deduce $U_{ij}' = V_{ij} \oplus PW_i$, get U_{ij}' and V_{ij} above, then it can compute $PW_i = V_{ij} \oplus U_{ij}'$.

To sum up, the attacker has ability of launching the password guessing attack to acquire the patient real password. Because the above-mentioned password guessing attack does not have need to communicate with the server S_j, this attack is easy to achieve.

AKAS scheme

In this section, we describe an improved chaotic map-based authenticated key agreement scheme with strong anonymity in detail to deal with the security problems that trouble Lee et al.'s and Shu's scheme. It is by enhancing the security of the related parameters stored in the smart card and calculating users dynamic identity that an improved authenticated and key agreement scheme(AKAS) in the multi-server environment is presented. Our new scheme contains four phases: the initialization phase, the registration phase, the login phase, and the authentication phase. All the steps are performed as follows:

Initialization phase

The registration centre RC first selects a random number x, and randomly selects a number sr as the master key of RC in this phase. It then, calculates $\omega_j = h(sr \parallel SID_j)$ and sends message ω_j to the server S_j via a secure channel.

Registration phase

This phase is performed in the registration centre RC, which includes the user registration phase and the server registration phase. First the patient must get registered with the RC as a legitimate user to gain the servers' service rendered by $S = \{S_1, S_2, \cdots, S_n\}$. The minutia designs of the registration phase are portrayed as follows and illustrated in Fig. 2.

Step R1: The patient U_i selects its unique identity ID_i, secret password PW_i and generates a random number N_i, O_i. It then sends a registration request message $(ID_i, PW_i \oplus N_i)$ to the RC through the secure channel.

Step R2: Upon receiving the registration request message from the patient U_i, registration centre RC calculates $\omega_j = h(sr \parallel SID_j)$, $T_{\omega_j}(x)$, $V_{ij} = h(ID_i \parallel P_{ij} \parallel \omega_j)$, $U_{ij} = V_{ij} \oplus PW_i \oplus N_i$. Given P_{ij} is a valid period of service for users. Then, registration centre RC calculates the users dynamic identity NID, $NID{=}E_{T_{\omega_j}(x)}(ID_i \parallel P_{ij})$ and stores $\{NID, U_{ij}, P_{ij}, x, h(), p\}$ in the smart card. Finally, registration centre RC sends the smart card SC stored patients privacy data to the patient U_i.

Step R3: Upon receiving the smart card SC from the registration centre RC, the user U_i computes $U_{ij}' = U_{ij} \oplus N_i$ and U_{ij} will be replaced with U_{ij}'. Finally, the user U_i stores $\{NID, U_{ij}', P_{ij}, x, h(), p\}$ in the smart card SC.

Second, similar to the user registration phase, the server must get registered with the RC as a legitimate server to provide the convenient service for patients. The server sends message SID_j to the RC via a secure channel. After receiving

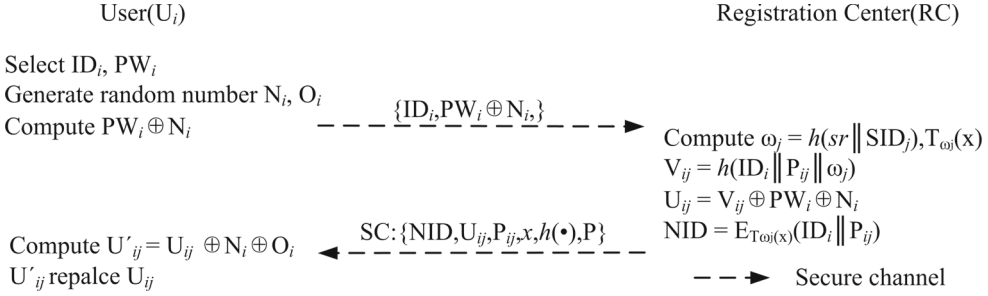

Fig. 2 Registration phase

the registration request message from the server S_j, the registration centre RC calculates $\omega_j = h(sr \parallel SID_j)$ and sends message ω_j to the server S_j via a secure channel.

Login phase

As shown in Fig. 3, the following steps will be performed by the patient U_i in the login phase.

Step L1: The patient U_i inserts its smart card into a specific card reader and then inputs its unique identity ID_i and its secret password PW_i.

Step L2: The smart card first computes $V'_{ij} = U_{ij} \oplus PW'_i$, chooses a random number r, and then calculates $C_1 = T_r(x) \bmod P$, $C_2 = h(ID_i \parallel P_{ij}) \oplus V'_{ij}$.

Step L3: The patient U_i sends the login request message $M_1 = \{NID, x, C_1, C_2\}$ to the server S_j via a public and insecure channel.

Authentication phase

In order to satisfy mutual authentication and key agreement between two entities, the patient and the server S_j, when the server S_j received the message M_1 , the following steps are performed as shown in Fig. 3.

Step A1: First, the server S_j calculates $T_{\omega_j}(x)$ and decrypts the message NID from the patient U_i to retrieve $(ID_i \parallel P_{ij})$ by using $T_{\omega_j}(x)$ above, that is $ID_i \parallel P_{ij} = D_{T_{\omega_j}(x)}(NID)$. Check the validity of ID_i and P_{ij}, if we prove that the patient identity ID_i does not meet, the server S_j terminates the session, or if we prove that P_{ij} is invalid, the server S_j stop providing the service to patient. Otherwise, it calculates $V'_{ij} = h(ID_i \parallel P_{ij}) \oplus C_2$, and verifies whether V_{ij} and V'_{ij} are equal or not. If not equal, the server S_j rejects the session, otherwise, the server S_j generates a random integer s, and calculates $C_3 = T_s(x) \bmod P$, $SK = T_s(C_1) \bmod P$. Then, the telecare server S_j computes $C_4 = h(ID_i \parallel SID_j) \oplus C_3$. Finally, the server S_j sends the message $M_2 = \{C_4\}$ to the patient through the public and insecure channel.

Step A2: Upon receiving the message M_2 sent by the server S_j, the patient calculates $C_3 = h(ID_i \parallel SID_j) \oplus C_4$ using the received message M_2. Check the validity of ID_i and S_j, if invalid, then the patient quit the session. Otherwise, the patient U_i believes that the server S_j is

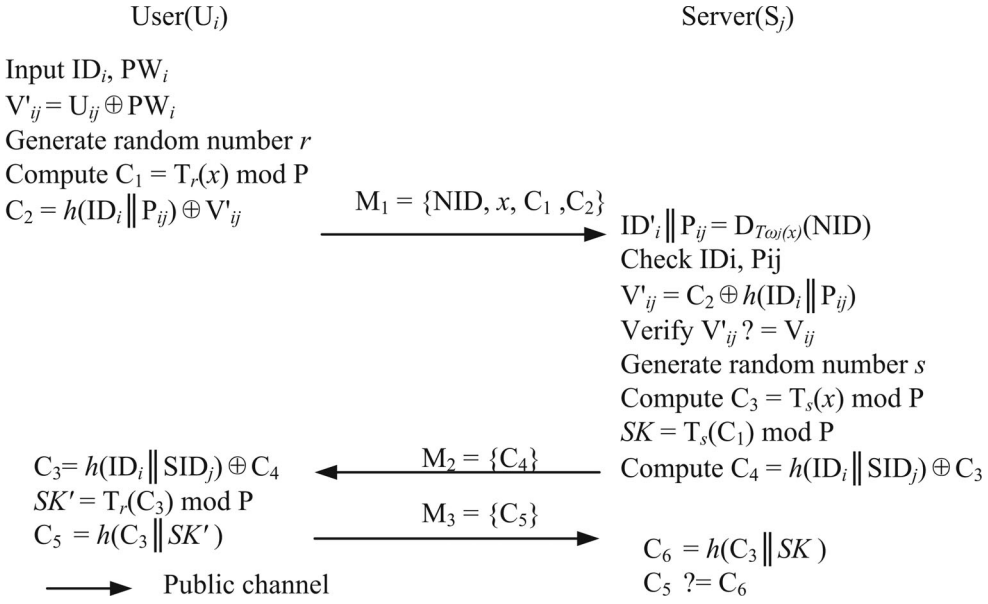

Fig. 3 Login and Authentication phase

a legal server. Finally, calculates $SK' = T_r(C_3) \bmod P$, $C_5 = h(C_3 \parallel SK')$ and sends message $M_3 = \{C_5\}$ to the server S_j via the public and insecure channel.

Step A3: After receiving the message M_3, the server S_j calculates $C_6 = h(C_3 \parallel SK)$ and verifies whether C_5 and C_6 are equal. If not equal, the server S_j terminates the session. Otherwise, the server S_j believes that the patient U_i is a legal user in TMIS system, and sets the SK as their shared session key.

According to the semi-group feature, it is easy to prove that $SK = SK'$. In other words, in this session, the patient U_i and the server S_j generate a shared session key between them. This session key can protect subsequent communications and provide mutual authentication.

Security analysis

This section makes a portrait of the security analysis of the AKAS scheme, including the security properties and formal security analysis under the BAN logic model.

Formal security analysis (BAN logic)

This subsection involves the formal security analysis of the AKAS scheme under the Burrows-Abadi-Needham logic (BAN logic) model. Some of the notations used in the BAN logic model are described as under in Table 2.

Some logical postulates or rules employed in the BAN logic model are shown as under:

Rule 1. Message meaning rule: $\frac{P|\equiv Q \overset{sk}{\longleftrightarrow} P, P \lhd \{X\}_{sk}}{P|\equiv Q|\sim X}$

Rule 2. Nonce verification rule: $\frac{P|\equiv \#X, P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$

Rule 3. Seeing rule: $\frac{P|\equiv P \left| \overset{sk}{\longleftrightarrow} Q, P \lhd \{X\}_{sk} \right.}{P \lhd X}$

Rule 4. Freshness rule: $\frac{P|\equiv \#X}{P|\equiv \#XY}$

It is necessary and essential for the AKAS scheme to satisfy the following goals to ensure its security under the BAN logic model, making full use of the above-mentioned postulates and assumption.

Goal 1: $U_i \mid\equiv U_i \overset{SK}{\longleftrightarrow} S_j$

Goal 2: $S_j \mid\equiv U_i \overset{SK}{\longleftrightarrow} S_j$

Goal 3: $U_i \mid\equiv S_j \mid\sim (T_s(x))$

Initially, the messages exchanged in the AKAS scheme can be transformed into idealized form in the following manner.

M1: $U_i \rightarrow S_j : \{NID, x, C_1, C_2\}$

M2: $S_j \rightarrow U_i : \{C_4\}$

M3: $U_i \rightarrow S_j : \{C_5\}$

Secondly, the following assumptions have been constituted to demonstrate the security of the AKAS scheme.

A1: $U_i \mid\equiv \#r, S_j \mid\equiv \#s$

A2: $S_j \mid\equiv U_i \mid\Rightarrow \#r, U_i \mid\equiv S_j \mid\Rightarrow \#s$

A3: $S_j \mid\equiv \#T_s(x)$

A4: $U_i \mid\equiv \#T_u(x)$

A5: $U_i \overset{SK}{\longleftrightarrow} S_j$

Based on the BAN logic rules and the assumptions, we make a specific analysis of the idealized form of the AKAS scheme. The main proofs are declared as under:

Proof Because there is a seeing rule $U_i \lhd \left(U_i \overset{SK}{\longleftrightarrow} S_j\right)$, by applying freshness rule and the assumption A4, we get the expression $U_i \mid\equiv \# \left(U_i \overset{SK}{\longleftrightarrow} S_j\right)$. According to the assumption A5 $U_i \overset{SK}{\longleftrightarrow} S_j$ and the message meaning rule, we could gain the expression $U_i \mid\equiv S_j \mid\sim (U_i \overset{SK}{\longleftrightarrow} S_j)$. According to the nonce verification rule we know the expression $U_i \mid\equiv S_j \mid\equiv (U_i \overset{SK}{\longleftrightarrow} S_j)$. Finally, we prove that the patient U_i believes that SK is the shared session key between the patient U_i and the server S_j. □

Proof Because there is a seeing rule $S_j \lhd (U_i \overset{SK}{\longleftrightarrow} S_j)$, by applying freshness rule and the assumption A3

Table 2 Notation description

Symbol	Description
$P \mid\equiv X$	The principal P believes X, or alternatively, P believes the statement X.
$P \lhd X$	P sees X, P receives some message X and may read or repeat it in any message.
$P \mid\sim X$	P once said X, P had sent some message X and P believed that message when sent.
$\#(X)$	The message X may be treated as fresh.
$P \mid\Rightarrow X$	P has got jurisdiction over X, or P has authority over X and could be trusted.
$P \overset{sk}{\longleftrightarrow} Q$	P and Q can communicate with the shared session key sk.
$\langle X \rangle_Y$	The formulate X is combined with the formulate Y.
$\{X\}_{sk}$	X is encrypted with the key sk.

$S_j \mid\equiv \#T_s(x)$, we get the expression $S_j \mid\equiv \#(U_i \overset{SK}{\longleftrightarrow} S_j)$. According to the message meaning rule and the assumption A5 $U_i \overset{SK}{\longleftrightarrow} S_j$, we could obtain the expression $S_j \mid\equiv U_i \mid\sim \left(U_i \overset{SK}{\longleftrightarrow} S_j\right)$. According to the nonce verification rule, we know the expression $S_j \mid\equiv U_i \mid\equiv (U_i \overset{SK}{\longleftrightarrow} S_j)$. Finally, we prove that the server S_j believes that SK is the shared session key between the server S_j and the patient U_i. □

Proof We must confirm that the server S_j has sent the fresh message to the patient U_i, and verify that the server S_j is the legal server, that is $U_i \mid\equiv S_j \mid\sim (T_s(x))$. According to the assumption A5, we know that the server S_j and the patient U_i share the session key SK, and the patient U_i has received the message C_4 sent by the server S_j, that is, $\frac{U_i \mid\equiv U_i \overset{SK}{\longleftrightarrow} S_j, U_i \triangleleft C_4}{U_i \mid\equiv S_j \mid\sim C_4}$. According to the message meaning rule, we know that the message transmitted between the server S_j and the patient U_i contains $T_s(x)$, then, we get the expression $\frac{U_i \mid\equiv U_i \overset{SK}{\longleftrightarrow} S_j, U_i \triangleleft h(ID_i \| SID_j) \oplus C_3}{U_i \mid\equiv S_j \mid\sim h(ID_i \| SID_j) \oplus C_3}$. So, the patient U_i believes that the server S_j has sent the message contained $T_s(x)$, that is, $\frac{U_i \mid\equiv S_j \mid\sim C_3, U_i \triangleleft T_s(x)}{U_i \mid\equiv S_j \mid\sim T_s(x)}$. According to the assumption A3 $S_j \mid\equiv \#T_s(x)$, we get the expression $U_i \mid\equiv S_j \mid\sim (T_s(x))$. Finally, we prove that the patient U_i believes the server S_j has sent the fresh message contained $T_s(x)$, and verify that the server S_j is a legal server. □

By performing the logic deduction process above, the expected goal has been achieved.

$U_i \mid\equiv U_i \overset{SK}{\longleftrightarrow} S_j$, that is, the patient U_i believes that SK is the shared session key between the patient U_i and the server S_j.

$S_j \mid\equiv U_i \overset{SK}{\longleftrightarrow} S_j$, that is, the server S_j believes that SK is the shared session key between the server S_j and the patient U_i.

$U_i \mid\equiv S_j \mid\sim (T_s(x))$, that is, the patient U_i believes the server S_j has sent the fresh message contained $T_s(x)$.

The above BAN logic analysis formally attests that the AKAS scheme satisfies mutual authentication and the session key SK is mutually constructed between the patient U_i and the server S_j.

Security properties

The proposed scheme can withstand a series of security attacks such as replay attacks and privileged user attacks as well as a series of security features such as strong anonymity and mutual authentication. The comparison in terms of security properties between our AKAS scheme and other schemes [20–23] has been given in Table 3. We visibly realize that our improved AKAS scheme has a higher level of security than the other existing schemes. The security properties of the proposed AKAS scheme are expounded as below:

Replay attacks

The proposed AKAS scheme cannot launch the replay attacks. Suppose that the attacker *Eve* intercepts the message M_1 and replays it to the server S_j planing on impersonating the patient U_i. However, *Eve* can not be competent to construct a valid $C_5 = h(C_3 \parallel SK')$ to pass the validation process of the server S_j unless it can correctly guess the shared session key SK. However, when an attacker *Eve* keep trying to compute the session key SK utilizing intercepted messages $T_s(x)$, $T_r(x)$, and the random number x stored in the smart card SC, the attacker *Eve* will face the CMCDHP problem. In addition, the identity ID_i of the patient U_i is protected by a secure one-way hash function so that the attacker *Eve* can not extract the identity ID_i from the intercepted message. Moreover, the attacker can not compromise the new dynamic identity NID without the knowledge of the patient's identity ID_i. Consequently, the AKAS scheme can withstand the replay attack.

Privileged insider attacks

The proposed AKAS scheme can withstand the privileged user attacks. In the registration phase, the secret password PW_i of the patient U_i is protected by the random integer N_i and O_i. As a result, a malicious privileged insider on the server side cannot gain the patient's secret password during the registration phase.

Table 3 Security comparisons with other relevant schemes

Attack	Tsaur et al. [20]	Li et al. [21]	Lee et al. [22]	Shu [23]	Ours
Replay attack	Yes	Yes	Yes	Yes	Yes
Password guessing attack	Yes	Yes	No	No	Yes
Mutual authentication	Yes	Yes	Yes	Yes	Yes
Privileged user attack	No	No	Yes	Yes	Yes
user anonymity	No	No	No	No	Yes

Strong anonymity

The proposed AKAS scheme can provide strong anonymity. In a multi-server architecture, the scheme should satisfy that other servers can not acquire the identity information of both parties in this session when they communicate with each other, in addition to providing the required anonymity in the single-server environment. The concrete analysis is as follows:

First, the attacker *Eve* can easily extract the information $\{NID, U_{ij}, P_{ij}, x, h(.), P\}$ stored in the smart card. However, since the patient's identity ID_i is protected by NID, the attacker must decrypt NID to acquire the identity information of the patient, that is, the attacker must calculate the decryption key $T_{\omega_j}(x)$. On account of the private key ω_j which is confidential, the attacker *Eve* can not decrypt NID to gain the patient's identity.

Second, the attacker controls the communication channel between the patient U_i and the server S_j, so that it can intercept and eavesdrop on all the messages transmitted on the public channel, and keep trying to find out the identities of the patients or the servers. In our proposed AKAS scheme, three messages $M_1 = \{NID, x, C_1, C_2\}$, $M_2 = \{C_4\}$ and $M_3 = \{C_5\}$ are sent, respectively. In the message M_1 , the identity ID_i of the patient is hidden in NID. In the message M_2, the patient's identity ID_i is encrypted and protected by a one-way hash function, and the attacker *Eve* cannot acquire it. In the message M_3, there is no information about the patient's identity ID_i, so it is out of the question for the attacker to gain the patient's identity.

Third, in the message M_1, M_3, there is no information about the identity of the server, so it is unhelpful for the attacker *Eve* to obtain the identity of the server. The servers' identity contained in the message M_2 is also concealed by the hash function, and the attacker cannot get it. All this being said, the AKAS scheme achieve strong anonymity.

Mutual authentication

The proposed AKAS scheme can achieve mutual authentication to participants. The server S_j and patient U_i in the scheme are able to provide mutual authentication by verifying the message M_4 and M_5 separately. Therefore, the scheme finishes mutual authentication.

Performance analysis

In this section, we evaluate the performance of proposed AKAS scheme and other relevant schemes in computation costs, communication costs and memory requirements.

Computation costs analysis

This subsection deals with computation costs of AKAS scheme with other relevant schemes as shown in Table 4.

A few notations used in the comparison are as under:

T_c:the time for executing the Chebyshev operation.

T_h:the time taken for the hash operation.

T_s:the time for the symmetric key cryptography.

According to [28] published in 2017 , the computational complexity of the above operation is: $T_c \approx 21.04\ ms$, $T_s \approx 8.6\ ms$, and $T_h \approx 0.58\ ms$. Because the running time of the exclusive-or (XOR) operation is able to negligible, it is remarked that the exclusive-or (XOR) operation is ignored.

Table 4 demonstrates the comparison for Tsaur et al. [20], Li et al. [21], Lee et al. [22] and Shu [23] against proposed AKAS scheme in terms of computation costs. First, Table 4 indicates that the computational complexity of the proposed AKAS scheme is much lower than that of the scheme [22, 23]. Besides, Lee et al.'s scheme [22] and Shu's scheme [23] fails to meet the user anonymity. Although Tsaur et al.'s scheme [20] and Li et al.'s scheme [21] have advantages of high efficiency, they can not resist privileged user attacks and password guessing attacks. Simultaneously, they do not achieve user anonymity and perfect forward secrecy. Second, from the aspect of communication, Table 4 discloses that the schemes [20–23] and the proposed AKAS scheme need three times of message exchange to complete mutual authentication and shared session key agreement. Finally, from the total computational complexity, Table 4 reveals that the proposed scheme reduces symmetric encryption/decryption operations twice and Chebyshev chaotic mapping operations twice, and improves the efficiency by 21% compared with Lee et al.'s

Table 4 Performance comparisons with other relevant schemes

Phases		Tsaur et al. [20]	Li et al. [21]	Lee et al. [22]	Shu [23]	Ours
Registration phase	User server	$3T_h + T_s$	$3T_h + T_s$	$2T_h$	T_h	$T_h + T_s$
		T_h	T_h	T_h	T_h	T_h
Login phase	User server	$4T_h + 3T_s$	$4T_h + 3T_s$	$5T_h + 3T_c$	$T_h + 2T_s + 3T_c$	$3T_h + 2T_c$
		$2T_h + 4T_s$	$2T_h + 4T_s$	$6T_h + 3T_c$	$2T_h + 2T_s + 3T_c$	$3T_h + T_s + 2T_c$
Total		74.6 ms	74.6 ms	134.36 ms	134.94 ms	106 ms

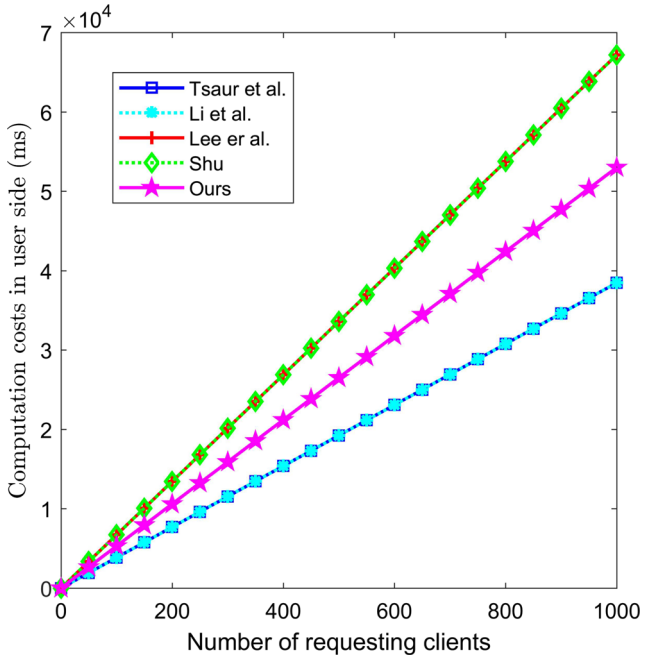

Fig. 4 Computation costs in user side

[22] and Shu's scheme [23]. Therefore, compared with the existing scheme, Lee et al.'s [22] and Shu's scheme [23]: whether the user side or server side, the proposed scheme save much computation costs. (as shown in Figs. 4 and 5). The proposed AKAS scheme is not only could withstand the multifarious security attacks, but also more robust and suitable for the TMIS system.

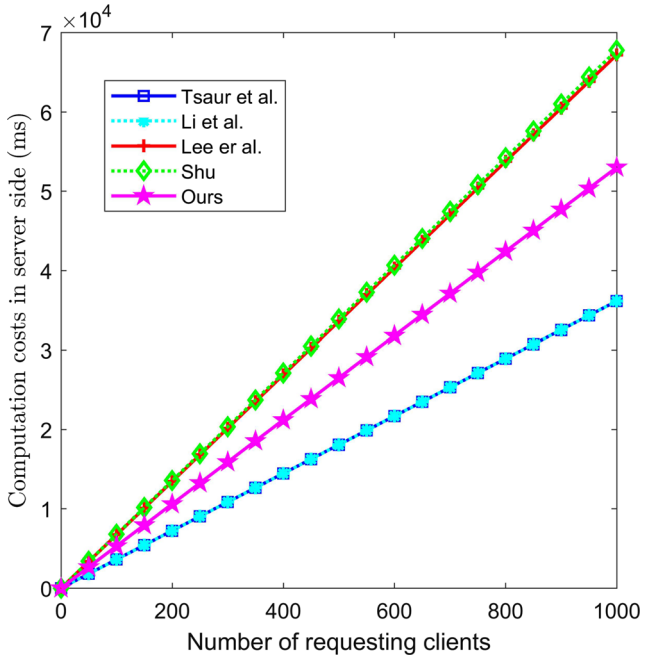

Fig. 5 Computation costs in server side

Table 5 Comparison of communication costs and memory requirements

	Login phase	Authentication phase	Memory requirements
Tsaur et al. [20]	832	512	736
Li et al. [21]	672	512	576
Lee et al. [22]	640	640	816
Shu [23]	576	416	976
Ours	736	320	912

Communication costs and memory requirements analysis

In this subsection, we calculate the communication costs transmitted between patient U_i and sever S_j of our AKAS scheme and other existing and related schemes, and the memory requirements of the smart card to store the security parameters. In this paper, we suppose that the identity digest of the user identity, hash function, timestamps, Chebyshev chaotic map, and nonce have the same length of 160 bits. Besides, the key for symmetric encryption/decryption has the length of 256 bits, and the prime p which is used in the modular arithmetic has the length of 16 bits.

The communication costs of our AKAS scheme and other relevant schemes have been shown in Table 5. In the login phase of our AKAS scheme, patient U_i sends $\{NID, x, C_1, C_2\}$ to the server S_j. Based on the aforesaid assumptions, the cost of this message is 736 bits. In the authentication phase, server S_j sends the message $\{C_4\}$ to the patient U_i and the patient U_i subsequently sends the message $\{C_5\}$ to the server S_j, which costs 160 * 2 = 320 bits. As such, Tsaur et al.'s scheme [20] costs 832 bits and 512 bits respectively, Li et al.'s scheme [21] costs 672 bits

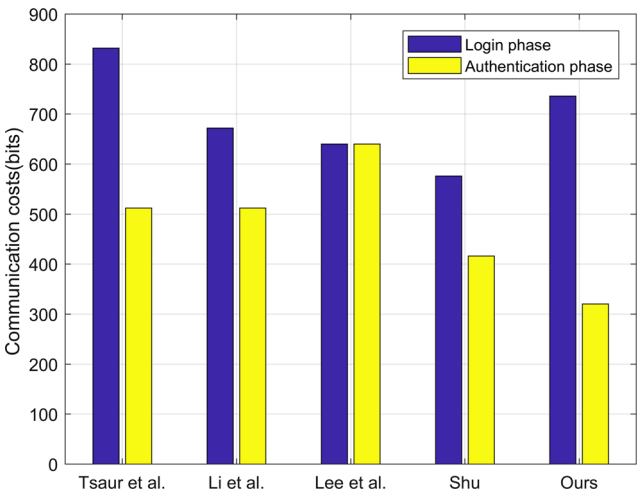

Fig. 6 Communication costs

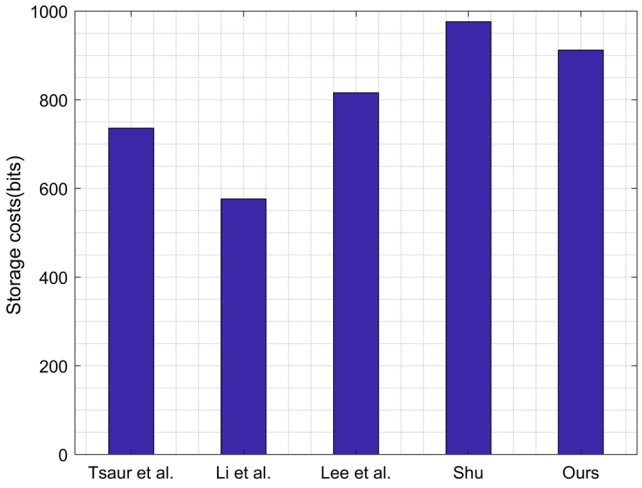

Fig. 7 Storage costs

and 512 bits, Lee et al.'s scheme [22] costs 640 bits and 640 bits, and Shu's scheme [23] costs 576 bits and 416 bits.

From Fig. 6, we extrapolate that the proposed AKAS scheme is more efficient than Tsaur et al.'s scheme [20], Li et al.'s scheme [21], Lee et al.'s scheme [22] and Shu's scheme [23] in authentication phase. Even though the total communication costs of our AKAS scheme is slightly higher than that of Shu's scheme [23], our AKAS scheme can resist multifarious attacks and provide more security properties. Hence, the performance of the proposed scheme surpasses that of the other related schemes.

In [20], a smart card SC contains $\{UID_i, u_i, E - T_{ij}, A_{ij}\}$, which requires 736 bits. In [21], a smart card SC contains $\{U_i, E - T_{ij}, A_{ij}\}$, which requires 576 bits. In [22], a smart card SC contains $\{ID_i, u_i, P_i, x, h(), p\}$, which requires 816 bits. In [23], a smart card SC contains $\{ID_i, u_{ij}, P_{ij}, x, T_{\omega_j}(x), h(), p\}$, which requires 976 bits. In the proposed AKAS scheme, a smart card SC contains $\{NID, u_{ij}, P_{ij}, x, h(), p\}$, which requires 912 bits. Compared with the scheme in [23], the proposed AKAS scheme reduces the storage costs as shown in Fig. 7. The comparison in terms of communication costs and memory overhead between the AKAS scheme and other related work is given in Table 5.

Conclusions

This paper makes a concrete analysis of Lee et al.'s and Shu's scheme for multi-server architecture and exhibits its review entirely. Analyzing the Lee et al.'s and Shu's scheme, we reveal that both Lee et al.'s and Shu's scheme fail to finish the security feature, particularly, strong anonymity and withstand password guessing attack. And then, an enhanced mutual authentication and key agreement scheme whose security is reduced to the chaotic map computational Diffie-Hellman problem for multi-server environment in TMIS has been raised and analyzed using BAN logic. Security analysis is also presented, which testifies that our AKAS scheme can not only resist all kinds of security threats and attacks but also more lightweight and efficient than the existed schemes. Therefore, the proposed AKAS scheme is more suitable for the TMIS system.

Acknowledgments This work is supported in part by the National key Research and Development Program of China under Grant No.2017YFB1400704, the Key Research and Development Program of Shaanxi Province under Grant No.2019ZDLGY12-03, 2019ZDLGY13-06, 2019ZDLGY12-04 and 2019Z-DLGY13-01, the National Natural Science Foundation of China under Grant No.61972310, 61972308 and 61902295.

Compliance with Ethical Standards

Conflict of interests Author Hui Qiao declares that she has no conflict of interest. Author Xuewen Dong declares that he has no conflict of interest. Author Yulong Shen declares that he has no conflict of interest.

Ethical approval This article does not contain any studies with human participants performed by any of the authors.

References

1. Lee, I., and Lee, K., The internet of things (IoT): applications, investments, and challenges for enterprises. *Business Horizons* 58(4):431–440, 2015.
2. Seyedi, M., Kibret, B., Lai, D. T., and Faulkner, M., A survey on intrabody communications for body area network applications. *IEEE Trans. Biomed. Eng.* 60(8):2067–2079, 2013.
3. Ji, Y., Zhang, J., Ma, J., Chao, Y., and Xin, Y., Bmpls: Blockchain-based multi-level privacy-preserving location sharing scheme for telecare medical information systems. *J. Med. Syst.* 42(8):147?, 2018.
4. Liu, X., and Ma, W., Cdaka: a provably-secure heterogeneous cross-domain authenticated key agreement protocol with symptoms-matching in tmis. *J. Med. Syst.* 42(8):135, 2018.
5. Lamport, L., Password authentication with insecure communication. *Commun. ACM* 24(11):770–772, 1981.
6. Irshad, A., Sher, M., Nawaz, O., Chaudhry, S. A., Khan, I., and Kumari, S., A secure and provable multi-server authenticated key agreement for TMIS based on Amin et al. scheme. *Multimed. Tools Appl.* 76(15):16463–16489, 2017.
7. Chen, C.-T., and Lee, C.-C., A two-factor authentication scheme with anonymity for multi-server environments. *Security and Communication Networks* 8(8):1608–1625, 2015.
8. Luo, E., Bhuiyan, M. Z. A., Wang, G., Rahman, M. A., Wu, J., and Atiquzzaman, M., Privacyprotector: Privacy-protected patient data collection in IoT-based healthcare systems. *IEEE Commun. Mag.* 56(2):163–168, 2018.
9. Tao, H., Bhuiyan, M. Z. A., Abdalla, A. N., Hassan, M. M., Zain, J. M., and Hayajneh, T., Secured data collection with hardware-based ciphers for IoT-based healthcare. *IEEE Internet Things J.* PP(99).
10. Bhuiyan, M. Z. A., Wang, G., Wu, J., Cao, J., Liu, X., and Wang, T., Dependable structural health monitoring using wireless sensor

networks. *IEEE Trans. Dependable Secure Comput.* 14(4):363–376, 2017.
11. Hsiang, H. C., and Shih, W. K., Improvement of the secure dynamic id based remote user authentication scheme for multi-server environment. *Computer Standards and Interfaces* 31(6):1118–1123, 2009.
12. Li, X., Xiong, Y., Ma, J., and 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.
13. Liao, Y. P., and Wang, S. S., A secure dynamic id based remote user authentication scheme for multi-server environment. *Computer Standards and Interfaces* 31(1):24–29, 2009.
14. Lin, H., Wen, F., and Du, C., An improved anonymous multi-server authenticated key agreement scheme using smart cards and biometrics. *Wirel. Pers. Commun.* 84(4):2351–2362, 2015.
15. Lu, Y., Li, L., Yang, X., and Yang, Y., Robust biometrics based authentication and key agreement scheme for multi-server environments using smart cards. *Plos One* 10(5):e0126323, 2015.
16. Odelu, V., Das, A. K., and Goswami, A., A secure biometrics-based multi-server authentication protocol using smart cards. *IEEE Trans. Inf. Forensics Secur.* 10(9):1953–1966, 2015.
17. Sood, S. K., Sarje, A. K., and Singh, K., A secure dynamic identity based authentication protocol for multi-server architecture. *J. Netw. Comput. Appl.* 34(2):609–618, 2011.
18. Xue, K., Hong, P., and Ma, C., A lightweight dynamic pseudonym identity based authentication and key agreement protocol without verification tables for multi-server architecture. *J. Comput. Syst. Sci.* 80(1):195–206, 2014.
19. Tao, W., Liao, W., and Jianfeng, M. A., Analysis and improvement of an authentication protocol for the multi-server architecture. *Journal of Xidian University* 40(6):174–179, 2013.
20. Tsaur, W. J., Li, J. H., and Lee, W. B., An efficient and secure multi-server authentication scheme with key agreement. *J. Syst. Softw.* 85(4):876–882, 2012.
21. Li, C. T., Lee, C. C., Weng, C. Y., and Fan, C., An extended multi-server-based user authentication and key agreement scheme with user anonymity. *KSII Trans. Internet Inf. Syst.* 7(1):119–131, 2013.
22. Lee, C. C., Lou, D. C., Li, C. T., and Hsu, C. W., An extended chaotic-maps-based protocol with key agreement for multiserver environments. *Nonlinear Dyn* 76(1):853–866, 2014.
23. Shu, J., and Commercial, D. E., Authenticated key agreement protocol based on extended chaotic maps for multi-server environments. *Application Research of Computers* 63(5):50507?-050507, 2016.
24. Kocher, P. C., Jaffe, J., and Jun, B., Differential power analysis. In: *International Cryptology Conference*, pp. 388?-397, 1999.
25. Messerges, T., Dabbish, E., and Sloan, R., Examining smart-card security under the threat of power analysis attacks. *IEEE Trans. Comput.* 51(5):541–552, 2002.
26. Chang, Y. F., Yu, S. H., and Shiao, D. R., A uniqueness-and-anonymity-preserving remote user authentication scheme for connected health care. *J. Med. Syst.* 37(2):9902, 2013.
27. Zhang, L., Cryptanalysis of the public key encryption based on multiple chaotic systems. *Chaos, Solitons and Fractals* 37(3):669–674, 2008.
28. Irshad, A., Sher, M., Chaudhry, S., Xie, Q., Kumari, S., and Wu, F., An improved and secure chaotic map based authenticated key agreement in multi-server architecture. *Multimed. Tools Appl.* 77:01, 2017.