A Novel User Authentication and Key Agreement Protocol for Accessing Multi-Medical Server Usable in TMIS

ARTICLE *in* JOURNAL OF MEDICAL SYSTEMS · MARCH 2015

Impact Factor: 2.21 · DOI: 10.1007/s10916-015-0217-3 · Source: PubMed

2 AUTHORS, INCLUDING:

Ruhul Amin
Indian School of Mines
24 PUBLICATIONS **121** CITATIONS

SEE PROFILE

J Med Syst (2015) 39:33
DOI 10.1007/s10916-015-0217-3

PATIENT FACING SYSTEMS

A Novel User Authentication and Key Agreement Protocol for Accessing Multi-Medical Server Usable in TMIS

Ruhul Amin · G. P. Biswas

Received: 12 November 2014 / Accepted: 26 January 2015

Abstract Telecare Medical Information System *(TMIS)* makes an efficient and convenient connection between patient(s)/user(s) at home and doctor(s) at a clinical center. To ensure secure connection between the two entities (patient(s)/user(s), doctor(s)), user authentication is enormously important for the medical server. In this regard, many authentication protocols have been proposed in the literature only for accessing single medical server. In order to fix the drawbacks of the single medical server, we have primarily developed a novel architecture for accessing several medical services of the multi-medical server, where a user can directly communicate with the doctor of the medical server securely. Thereafter, we have developed a smart card based user authentication and key agreement security protocol usable for *TMIS* system using cryptographic one-way hash function. We have analyzed the security of our proposed authentication scheme through both formal and informal security analysis. Furthermore, we have simulated the proposed scheme for the formal security verification using the widely-accepted *AVISPA* (Automated Validation of Internet Security Protocols and Applications) tool and showed that the scheme is secure against the replay and man-in-the-middle attacks. The informal security analysis is also presented which confirms that the protocol has well security protection on the relevant security attacks. The security and performance comparison analysis confirm that the proposed protocol not only provides security protection on the above mentioned attacks, but it also achieves better complexities along with efficient login and password change phase.

Keywords Authentication · *AVISPA* simulator · Multi-Medical server · *TMIS* · Security attacks

This article is part of Topical Collection on *Patient Facing Systems*

R. Amin (✉) · G. P. Biswas
Department of Computer Science and Engineering, Indian School of Mines, Dhanbad 826004, India
e-mail: amin_ruhul@live.com

G. P. Biswas
e-mail: gpbiswas@gmail.com

Introduction

In *(TMIS)*, medical server generally maintains the electronic medical records of the registered user and provides various resources to the user like, health educators, physicians, hospitals, care-givers, public health organizations and home-care service. User-friendly, omnipresence and the low cost of internet technology, facilitates online medical services, in which a registered user can access the remote service at any instant from anywhere. When a registered user wants to get medical services, s/he uses smart card to the smart devices and transmits data to the medical server through public channel. Since, the system employs public channel, so maintaining user authentication, data privacy, data integrity and confidentiality of the data are very much essential, as the attacker/adversary may have full control over the public channel. Therefore, the attacker/adversary can eavesdrop, intercept, record, modify, delete, and replay the message broadcasting via public channel. In order to design an authentication protocol, many researchers employ several techniques like cryptographic one-way hash function, Chaotic maps, *ECC-RSA* cryptosystem and some other operations like X-OR, concatenate etc. The cryptographic hash-function and the chaotic maps both are important for designing user authentication protocol and each provide same level of security, but the computation

Published online: 15 February 2015

cost of hash function is very less than the chaotic maps operation. Moreover, the hash function based protocol is easier for implementation than the chaotic map based [15, 16, 18, 31, 32, 35, 43]. Therefore, we have used cryptographic one-way hash function for designing our proposed authentication protocol. As most of the user's use low entropy password, it is easier for an attacker to break the password based security system. It also has been observed that most of the password based user authentication protocols [10, 19, 37, 48, 51] suffer from off-line password guessing attack. Therefore, the biometric template such as fingerprint, iris, retina etc. should be incorporated in the user authentication protocol for providing higher security system. The biometric template based authentication protocol is more suitable than the password based protocol, because it possesses some important properties like 1) Biometric key cannot be lost or forgotten and very difficult to copy or share, 2)Biometric key is extremely hard to forge or distribute 3) Guessing biometric key is dreadfully difficult. In order to design an efficient user authentication and key agreement protocol for accessing either single medical server or multi-medical server, the following security aspects should be achieved:

1. An efficient login phase is necessary so that the protocol can detect wrong input information(s) in the early stage.
2. An authentication phase should be efficient in terms of computation and communication complexities.
3. Resistance of off-line password guessing attack.
4. Resistance of off-line identity guessing attack.
5. Resistance of user-impersonation attack.
6. Resistance of server masquerading attack.
7. Mutual authentication property should be provided.
8. The protocol should resist session key discloser attack.
9. The protocol should provide perfect forword/backword secrecy.
10. Resistance of insider attack.
11. Resistance of replay attack.
12. Resistance of denial-of-service(DoS) attack.
13. Avoidance of clock synchronization problem.
14. Password change phase should be provided and to be efficient.
15. The computation, communication and storage cost should be as minimum as possible.
16. No verification table should be involved in the server end.
17. Session key agreement and verification is essential.

Literature review

To ensure security and privacy during information transmission via public channel, the smart card based anonymous remote user authentication schemes are generally adopted. Last few years many password or biometric template based remote user authentication and key agreement protocols [1–3, 5, 10, 11, 14, 17, 21, 22, 24–30, 34, 36] have been proposed in the literature for different application systems. But it has been observed that most of the user authentication protocols still are not completely free from security attacks. In 2010, Wu et al. [49] proposed an efficient user authentication scheme for telecare medical information system and adding a pre-computing phase for low computational cost. But, Debiao He [12] demonstrated that Wu et al. [49] protocol fails to resist impersonation attack and insider attack and presented an enhance scheme of Wu et al. protocol and claimed that the enhance scheme is completely free from security attacks and takes low computational cost.

In 2012, Wei et al. [48] identified that both Wu et al. [49] and Debiao He [12] protocols are inefficient to meet two-factors authentication and also proposed a scheme, which is efficient and achieves two-factors authentication. Thereafter, Zhu [54] described that Wei et al. [48] protocol is vulnerable to off-line password guessing attack and also proposed an improved scheme for TMIS system. Then, Lee and Liu [33] demonstrated that Zhu's scheme cannot resist parallel session attack and presented a improved scheme and declared that their protocol is efficient in terms of security and applicable for TMIS systems. In 2012, dynamic-ID based authentication and key agreement protocol is presented by Chen et al. [8]. But, Lin [38] demonstrated that Chen et al.'s protocol suffers from user anonymity problem and password can be derived from the stolen smart card. Later, Cao and Zhai [6] demonstrated that Chen et al. protocol is vulnerable to off-line identity guessing attack, off-line password guessing attack and un-detectable online password guessing attack when the user's smart card is lost. They also presented an improved scheme for TMIS system. Thereafter, Xie et al. [50] described that Chen et al.'s [8] protocol suffers from several security weaknesses and proposed an improved scheme.

In 2013, Tan et al. [45] proposed a biometric based remote user authentication scheme for telecare medical information system and declared that their protocol achieves mutual authentication property and session key agreement between the user and the server. But, Yan et al.'s [52] reviews the proposed protocol presented by Tan et al.'s and declares that the scheme is vulnerable to denial-of-service attack. To eliminate the drawbacks of Tan et al.'s [45] protocol, Yan et al.'s [52] proposed an improved scheme for better security protection and performance.

In 2014, Mishra et al. [42] demonstrated that Yan et al. [52] protocol suffers from user anonymity problem, password guessing attack, inefficient login phase,

inefficient password and biometric update phase and three factors authentication problem. They also proposed and improved scheme for better security and performance. In the same year, Mishra et al. [43] have demonstrated that the chaotic maps based Jiang's et al. [18] protocol is insecure against denial of service attack and also has security flaws in the password change phase. Moreover, they proposed chaotic maps based user authentication and key agreement protocol for *TMIS* system to fix the above security weaknesses. Recently, Li et al. [35] described that the Lee et al.'s [31] chaotic maps based user authentication protocol has two security weaknesses such as 1) service misuse attacks for non-registered users and 2) Lack of user identity in the authentication phase and then proposed a better solution for accessing *TMIS* system.

Note that, the literature review regarding user authentication and key agreement protocol for accessing single medical server confirms that most of the protocols are not still completely free from security weaknesses. Therefore, it is most important for developing a secure and efficient user authentication and key agreement protocol for accessing *TMIS* system. In this paper, we have primarily designed a novel architecture for *TMIS* and then designed a secure user authentication and key agreement protocol for accessing multi-medical server, where the user can directly communicate with the physician server like Anesthesiologist, Cardiologist, Gastroenterologist, Hematologist, Nephrologist, Neurologist, Perinatologist etc. on demand. We have then analyzed the security of our authentication scheme through both formal and informal security analysis.

Attacker model

As the authentication protocol is executed over the insecure communication, the attacker has several advantages or capabilities. In the following, We present some valid assumptions.

1. An attacker ($\widehat{A}$) is able to extract the smart card information by monitoring the power consumption [23, 41]. For example if an attacker gets the smart card of the valid user, s/he then may get all the stored information of the smart card.
2. An attacker may eavesdrop all the communication between the entities involved of the protocol over the public channel. It is also assume that an attacker cannot intercept the message over the secure channel.
3. An attacker can guess low entropy password and identity individually easily but guessing two secret parameters (e.g. password, identity) is computationally infeasible in polynomial time.
4. An attacker can modify, delete and resend, reroute the eavesdrops message.
5. An attacker may be a legitimate user or vice versa.
6. The attacker knows the protocol description that means the protocol is public.
7. If we assume that the length of the user's identity and password is n character, then the probability of guessing approximately composed of n character is $\frac{1}{2^{6n}}$ as pointed out by [7].

Road map of the paper

After presenting satisfactory introduction in section "Introduction", the section "Preliminaries" discusses the concept and property of cryptographic one-way hash function and the bio-hashing technique as preliminaries of our works. In section "Our proposed architecture", we introduce our proposed architecture for accessing multi-medical server and then proposed user authentication security protocol for *TMIS* system in section "Proposed protocol". The formal security verification using *AVISPA* appears in section "Simulation for formal security verification using *AVISPA* tool" and the informal security analysis is given in section "Informal security analysis of the proposed protocol". The performance comparison are also made and given in section "Performance evaluation". Finally, we conclude the paper in section "Conclusion".

Preliminaries

In this section, we briefly introduce the basic concepts of cryptographic one-way hash function and bio-hashing technique.

Cryptographic One-way hash function

A cryptographic one-way hash function maps a string of arbitrary length to a string of fixed length called the hashed value. It can be symbolized as: $h : X \rightarrow Y$, where $X = \{0,1\}^*$, and $Y = \{0,1\}^n$. X is binary string of arbitrary length and Y is a binary string of fixed length n. It is used in many cryptographic applications such as digital signature, random sequence generators in key agreement, authentication protocols and so on. Cryptographic one-way hash function satisfies the following properties:

1. *Easiness:* Given $m \in X$, it can be easily compute y such that $y = h(m)$.
2. *Preimage Resistant:* It is hard to find m from given y, where $h(m) = y$.
3. *Second-Preimage Resistant:* It is hard to find input $m' \in X$ such that $h(m) = h(m')$ for given input $m \in X$ and $m' \neq m$.

4. *Collision Resistant:* It is hard to find a pair $(m, m') \in X \times X$ such that $h(m) = h(m')$, where $m \neq m'$.
5. *Mixing-Transformation:* On any input $m \in X$, the hashed value $y = h(m)$ is computationally indistinguishable from a uniform binary string in the interval $\{0, 2^n\}$, where n is the output length of hash $h(\cdot)$.

Definition 1 The advantages (Adv) of an attacker $\widehat{A}$ for finding collision resistance property of the one-way hash function is given as follows: $Adv_{\widehat{A}}^{hash}(t) = Prb[(m, m') \Longleftarrow_R \widehat{A}$ and $h(m) = h(m')]$, where $Prb[E]$ represents the probability of an event (E) in a random experiment, $\Longleftarrow_R \widehat{A}$ represents messages (m, m') is selected by the attacker randomly and $Adv_{\widehat{A}}^{hash}(t)$ represents the advantages of the probability over random choice by the attacker $\widehat{A}$ for the time duration t. The cryptographic one-way hash function is said to collision-resistant, if $Adv_{\widehat{A}}^{hash}(t) \leq \epsilon$, for any small values $\epsilon > 0$.

Bio-hashing

The biometric technology has the great importance for providing genuine user authentication in any authentication system. Generally, imprint biometric characteristics (face, fingerprint, palmprint etc.) may not be exactly same at each time. Therefore, high false rejection of registered users resulting low false acceptation, is often occurs in the evaluation of biometric systems. In order to resolve the high false rejection rate, Jina et al. [20] proposed a two-factor authenticator on iterated inner products between tokenised pseudo-random number and the user specific fingerprint features, which produces a set of user specific compact code that coined as Bio-Hashing. Later, Lumini and Nanni [39] proposed the improvement of Bio-Hashing. As pointed out by [7], Bio-Hashing is used to map a user/patients biometric feature onto user specific random vectors in order to generate a code, called bio-code and then discritizes the projection coefficients into zero and one. Bio-Hashing is always one-way function and secure as cryptographic one-way hash function.

Our proposed architecture

In this section, we have presented our proposed architecture and access control mechanism. The proposed architecture is shown in Fig. 1. There are basically four types of entities involved in the proposed architecture such as 1) many users/patients U_i, 2) single medical registration server (MRS), 3) many medical servers (MS_j) and 4) several physician servers (PS_k). The single medical registration server (MRS) is responsible for providing registration to the new patients (U_i) and medical server (MS_j). The physician servers (PS_k) provide several resources on demand to the registered users/patients through medical server, whereas the user(s)/patient(s) only access the physician servers through MS_j for solving several personal problems. Whenever users/patients want to access desired physician server of the medical server, initially he/she inserts the smart card and provides biometric template, identity along with password to the smart card reader device (SCR). The smart card reader (SCR) then verifies the authenticity of the user and transmits the login message to the medical server including the identity of the medical server (ID_{msj}) and the physician server (ID_k) . Based on the login message, the medical server first authenticates the user and then transmits another message to the physician server. The (PS_k) similarly authenticates the (MS_j, U_i) and forwards a message to the user through open channel. The user initially verifies the authenticity of the physician server and then computes a session key for transferring data securely with the physician server. After establishing session key, they both can exchange information(s) securely.

Proposed protocol

In this section, we proposed our user authentication and key agreement protocol based on the proposed new architecture shown in Fig. 1. As mentioned earlier, the protocol employs four types of entities (U_i, MRS, MS_j, PS_k), where the PS_k may different servers like Anesthesiologist, Cardiologist, Gastroenterologist, Hematologist, Nephrologist, Neurologist, Perinatologist etc. In our proposed authentication protocol, there are mainly five phases namely user registration phase, medical server registration phase, login phase, authentication and key agreement phase and password update phase. All these phases are presented below and all the notations are listed in Table 1:

Medical server registration phase

Whenever, the medical servers MS_j $(1 < j \leq m)$ want to join for providing several medical resources to the remote patients, MS_j must have to register with the MRS. For doing that, the MS_j chooses a desired identity ID_{ms_j} and forwards it to the MRS. On receiving it, MRS computes $X_j = h(ID_{ms_j} \parallel X_c)$ and transmits it to the MS_j through secure channel and completes the registration procedure. It may be noted that the identity of each MS_j must be primary key.

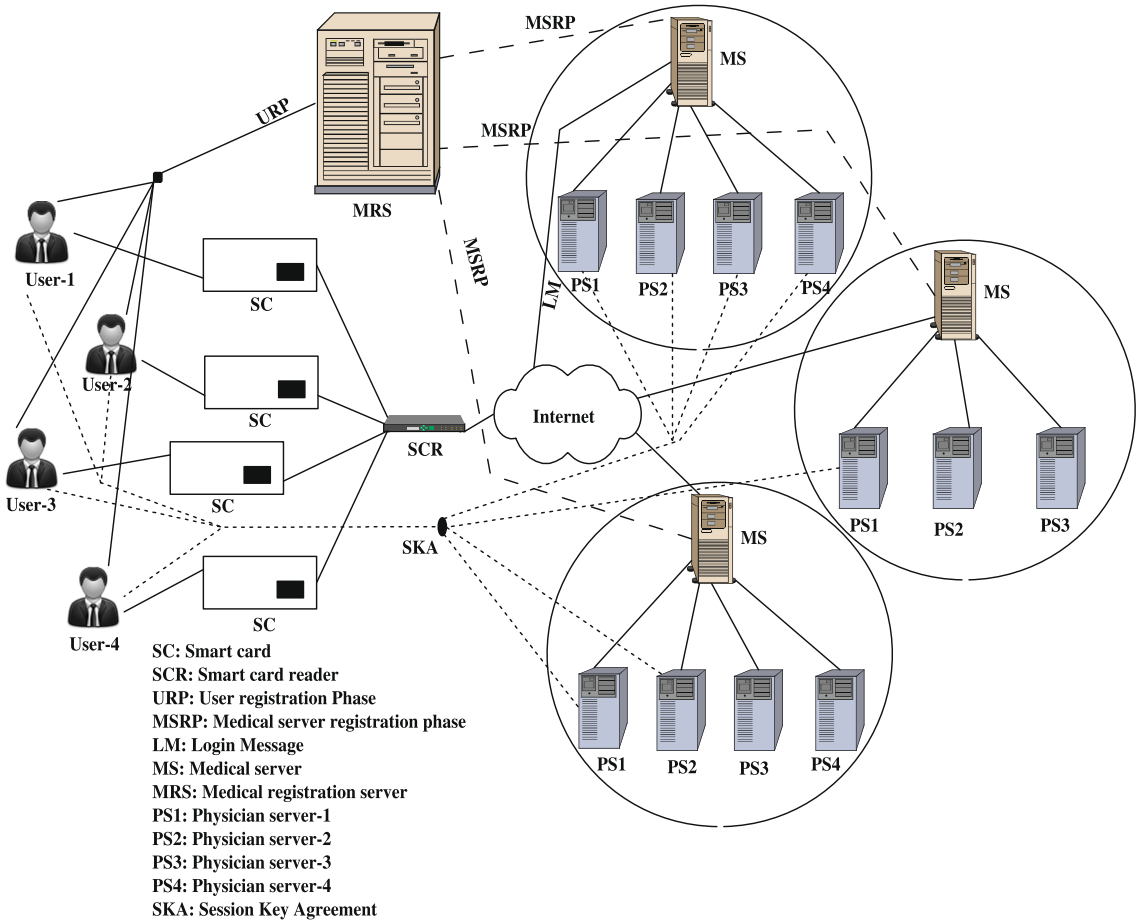

Fig. 1 Proposed Architecture for Accessing Multi-Medical Server System

User registration phase

Step R1: It is the initial phase for the U_i for accessing the medical services and any user U_i $(1 < i \leq n)$ can register with the (MRS). The user primarily chooses his/her desired identity ID_i, password PW_i, biometric template like fingerprint B_i and then sends $\langle ID_i, PWD_i, B_i \rangle$ to the (MRS) through secure channel or in person after computing $PWD_i = h(ID_i \parallel PW_i)$ at the time of registration.

Step R2: After receiving the registration request, MRS computes $F_i = H(B_i)$ by using the bio-hashing technique, $REG_i = h(ID_i \parallel PWD_i)$, $A_j = h(ID_i \parallel X_j) \oplus REG_i$, $P_j = h(ID_{ms_j} \parallel X_j \parallel F_i) \oplus h(REG_i \parallel F_i)$ for $(1 < j \leq m)$. Then, MRS stores a table containing the tuples $\langle ID_{ms_j}, A_j, P_j \rangle$ for $(1 < j \leq m)$ and further stores $\langle REG_i, h(), H() \rangle$ into the memory of smart card and issues it through secure channel or in person and completes the registration process, where $ID_{ms_j}, X_j = h(ID_{ms_j} \parallel X_c)$, X_c are the identity, secret key of the medical server and secret key of the MRS respectively. It may be noted that m represents number of medical servers in the system and according to memory availability of the smart card, the system may control minimum 100 medical servers which is enough. It is our assumption that a user always chooses very low entropy $\langle ID_i, PW_i \rangle$ which are guessable individually in polynomial time.

Login phase

After completing registration procedure successfully, the U_i can access any medical server at anytime from anywhere through a card reader or terminal device which is connected to the medical servers. All the steps of this phase are presented below:

Step-L1: The U_i primarily inserts his/her smart card into the card reader device and inputs biometric template B_i to the specific sensor device. The card reader then computes $F_i^* = H(B_i)$ and matches it with the stored F_i. If it matches, biometric verification passes successfully

Table 1 List of notations used

Symbol	Description
U_i	i-th User/patients ($1 < i \leq n$)
MRS	Medical registration server
MS_j	Medical server ($1 < j \leq m$)
PS_k	Physician server ($1 < k \leq p$)
PW_i	Password of the user U_i
ID_i	Identity of the user U_i
ID_{ms_j}	Identity of the medical server MS
ID_k	Identity of the physician server PS
B_i	Biometric of the user U_i
X_c	Secret key of the MRS
X_j	Secret key of the MS
X_k	Shared secret key between PS and MS
R_c	Random number generated by the U_i
R_{ms}	Random number generated by the MS
R_k	Random number generated by the PS
$h(\cdot)$	A secure One-way hash function
$H(\cdot)$	Bio-hashing function:
$\oplus$	Bit-wise Xor operation
$\|$	Concatenation operation

and asks to input $\langle ID_i, PW_i \rangle$ to the U_i; otherwise, stops the connection.

Step-L2: The card reader computes $REG_i^* = h(ID_i \| PW_i)$ and matches it with thestored REG_i. The matching result ensures whether the U_i has provided valid $\langle ID_i, PW_i \rangle$ or not. If it matches, the U_i chooses desired identity of medical and physician's server; otherwise, stops the session.

Step-L3: Based on the medical server's identity, the smart card reader (SCR) first retrieves $\langle A_j, P_j \rangle$ from the stored table of the smart card and then generates a random nonce R_c. The smart card computes $C_i = A_j \oplus REG_i = h(ID_i \| X_j)$, $D_i = h(C_i \| R_c)$, $E_i = P_j \oplus h(REG_i \| F_i) = h(ID_{ms_j} \| X_j \| F_i)$, $G_i = ID_i \oplus E_i$, $L_i = E_i \oplus R_c$ and transmits $\langle ID_{ms_j}, ID_k, F_i, D_i, G_i, L_i \rangle$ to the medical server MS_j as a login message through public/open channel.

Authentication and key agreement phase

The main aim of this phase is to achieve mutual authentication and session key agreement between the U_i and the physician server (PS_k). All the steps of this phase are presented below:

Step-A1: Based on the received login message, the MS_j computes $E_i^* = h(ID_{ms_j} \| X_j \| F_i)$ and extracts $ID_i^* = G_i \oplus E_i^*$, $R_c^* = L_i \oplus E_i^*$. Then, the MS_j further computes $C_i^* = h(ID_i^* \| X_j)$, $D_i^* = h(C_i^* \| R_c^*)$ and matches D_i^* with the received D_i. If it matches, the medical server believes the authenticity of the U_i; otherwise, stops the session.

Step-A2: The medical server generates a random nonce R_{ms} and computes $N_j = h(ID_k \| X_k \| F_i)$, $O_j = ID_i \oplus N_j$, $S_j = h(ID_i \| X_k) \oplus R_{ms}$, $RAN_j = R_c^* \oplus R_{ms}$, $Q_j = h(ID_i \| X_k \| N_j \| R_{ms})$ and transmits $\langle ID_k, O_j, S_j, Q_j, RAN_j, F_i \rangle$ to the physician server (PS_k) through public channel.

Step-A3: After receiving the message, the PS_k computes $N_j' = h(ID_k \| X_k \| F_i)$, $ID_i' = O_j \oplus N_i'$, $R_{ms}' = h(ID_i' \| X_k) \oplus S_j$, $R_c' = RAN_j \oplus R_{ms}'$, $Q_j' = h(ID_i' \| X_k \| N_j' \| R_{ms}')$ and matches Q_i' with the received Q_i. If it matches, the PS_k believes the authenticity of the (MS_j) and (U_i); otherwise, stops the session.

Step-A4: The (PS_k) then generates a random number R_k and computes $SK = h(ID_i' \| ID_k \| R_c' \| R_k)$, $T_k = h(h(ID_i' \| X_k) \| SK)$, $RAN_k = R_c' \oplus R_k$, $V_k = h(ID_i' \| X_k) \oplus R_k$, where SK is the session key between the U_i and the PS_k. Finally, the PS_k transmits $\langle T_k, RAN_k, V_k \rangle$ to the U_i through public channel.

Step-A5: After receiving the message from the PS_k, the U_i computes $R_k^* = RAN_k \oplus R_c$, $W_k = V_k \oplus R_k^* = h(ID_i \| X_k)$, $SK^* = h(ID_i \| ID_k \| R_c \| R_k^*)$, $T_k^* = h(W \| SK^*)$ and matches T_k^* with the received T_k. If it matches, the U_i believes that the PS_k is authentic and session key SK between the U_i and PS_k is verified.

Password change phase

In any password based user authentication scheme, it is a good property for designing password change phase to provide to change the password facility efficiently without help of the medical registration server. For doing that, Initially, the U_i inserts the smart card to the card reader and executes steps-*L1* and *L2* of the login phase for the authenticity of the U_i. After successful authentication, the card reader executes the following step for changing the password efficiently.

Step-P1: After verifying the user, the card reader asks to input a new password PW_i^{new} to the U_i. After getting it, the card reader computes $PWD_i^{new} = h(ID_i \| PW_i^{new})$, $REG_i^{new} = h(ID_i \| PWD_i^{new})$, $A_j^{new} = A_j \oplus REG_i \oplus REG_i^{new}$, $P_j^{new} = P_j \oplus h(REG_i \| F_i) \oplus h(REG_i^{new} \| F_i)$ and then replaces $\langle REG_i, A_j, P_j \rangle$ with the new values $\langle REG_i^{new}, A_j^{new}, B_j^{new} \rangle$ respectively and completes the password change phase successfully.

Simulation for formal security verification using *AVISPA* tool

In this section, the formal security analysis is presented to proof that the proposed authentication protocol is secure or *SAFE* against attacker. Based on the definition 1 (see section "Preliminaries"), we have primarily presented two theorems for formal security against $(\widehat{A})$ and then using the widely-accepted *AVISPA* [46](Automated Validation of Internet Security Protocols and Applications) tool for proving the proposed protocol is secure against passive and active attacks including the replay and man-in-the-middle attacks. The reveal oracle can be defined as: It is the oracle which will unconditionally output the input string (m) from the corresponding hash value $y = h(m)$.

Algorithm 1 $ALGO1^{HASH}_{\widehat{A},UAKPMS}$

Input: $\langle REG_i, A_j, P_j, h(), H(), ID_{ms_j}, ID_k, F_i, D_i, G_i, L_i\rangle$
Output: 0 or 1.
Call Reveal oracle on input REG_i for retrieving ID_i, PWD_i, B_i as $(ID_i^{'} \parallel PWD_i^{'} \parallel B_i^{'}) \leftarrow$ Reveal(REG_i).
Computes $REG_i^* = h(ID_i^{'} \parallel PWD_i^{'}$
if $(REG_i^{'} == REG_i)$ **then**
  Call Reveal oracle on input $PWD_i^{'}, F_i^{'}$ for retrieving $\langle PW_i, B_i\rangle$ information as $PW_i^{'} \leftarrow$ Reveal$(PWD_i^{'})$ and $B_i^{'} \leftarrow$ Reveal$(F_i^{'})$
  Computes $PWD_i^* = h(ID_i^{'} \parallel PW_i^{'})$ and $F_i^* = H(B_i^{'})$
  if $(PWD_i^* == PWD_i^{'})$ AND $(F_i^* == F_i^{'})$ **then**
    Accepts $PW_i^{'}$ and biometric template $B_i^{'}$ as the correct password and biometric template of the valid patients respectively.
    Call Reveal oracle on input D_i for retrieving confidential information $\langle ID_i, E_i, R_c\rangle$ as $(ID_i^{'} \parallel X_j^{'} \parallel E_i^{'} \parallel R_c^{'}) \leftarrow$ Reveal(D_i)
    Computes $G_i^{'} = ID_i^{'} \oplus E_i^{'}$ AND $L_i^{'} = E_i^{'} \oplus R_c^{'}$
    if $(ID_i^{'} == ID_i)$ AND $(G_i^{'} == G_i)$ AND $(L_i^{'} == L_i)$ **then**
      Accepts $ID_i^{'}$ as the correct identity of the valid patients and retrieves $E_i^{'}, R_c^{'}$ as a valid confidential parameters.
      Return(1) Success.
    else
      Return(0) Failure
    else
      Return(0) Failure
    end if
  end if
  Return(0)
end if

Theorem 1 *It is our assumption that the cryptographic one-way hash function closely behaves like an oracle, the proposed scheme is provably secure against an attacker for deriving the* $\langle ID_i, PW_i, B_i\rangle$ *of a legal user* U_i *even if the attacker knows all the smart card information(s).*

Proof Initially, we develop an attacker $\widehat{A}$ who has the ability to derive the user's identity ID_i, password PW_i and the biometric template B_i from the proposed protocol called as $UAKPMS$ (user authentication and key agreement protocol for multi-medical server). It is our assumption that an attacker has got the smart card of a valid patient by some means and extracted all the confidential parameters $\langle REG_i, A_j, P_j, h(), H()\rangle$ by monitoring power consumption [23, 41]. It is also our assumption that the attacker $\widehat{A}$ traps the login message $\langle ID_{ms_j}, ID_k, F_i, D_i, G_i, L_i\rangle$ between the user U_i and the medical server MS_j. The $(\widehat{A})$ then executes the algorithm $ALGO1^{HASH}_{\widehat{A},UAKPMS}$ for deriving $\langle ID_i, PW_i, B_i\rangle$ of a valid patient as given in the Algorithm 1.

In the following, we define the success probability for $ALGO1^{HASH}_{\widehat{A},UAKPMS}$:

$$SUCC1^{HASH}_{A,UAKPMS} = Prb[ALGO1^{HASH}_{\widehat{A},UAKPMS} = 1] - 1$$

where $Prb[E]$ is the probability of an event (E). Then, the advantages function of the $ALGO1^{HASH}_{\widehat{A},UAKPMS}$ is given below:

$$Adv1^{HASH}_{\widehat{A},UAKPMS}(t1, qr1) = Max_{\widehat{A}}[Adv1^{HASH}_{\widehat{A},UAKPMS}]$$

where the maximum is taken over all $\widehat{A}$ with the execution time $t1$ and the $qr1$ indicates that the number of queries made to the reveal oracle. The proposed scheme is said to be provably secure against the $\widehat{A}$ for deriving the $\langle ID_i, PW_i, B_i\rangle$, if $Adv1^{HASH}_{\widehat{A},UAKPMS}(t1, qr1) \leq \epsilon$ for any small value $\epsilon > 0$. Based on the $ALGO1^{HASH}_{\widehat{A},UAKPMS}$, if an attacker has the ability to invert the cryptographic one-way hash function, then only he/ she can easily derive $\langle ID_i, PW_i, B_i\rangle$ and win the game. However, it is computationally infeasible in polynomial time that is $Adv_{\widehat{A}^{HASH}(t)} \leq \epsilon$ for any small $\epsilon > 0$ (see section "Preliminaries"). Therefore, the condition $Adv1^{HASH}_{\widehat{A},UAKPMS}(t1, qr1) \leq \epsilon$, as $Adv1^{HASH}_{\widehat{A},UAKPMS}(t1, qr1)$ depends on the advantage $Adv_{\widehat{A}^{HASH}(t)}$. This proves that the proposed scheme is secure for deriving the user's information $\langle ID_i, PW_i, B_i\rangle$ against an attacker. □

Theorem 2 *It is our assumption that the cryptographic one-way hash function closely behaves like an oracle, the proposed scheme is provably secure against an attacker for deriving the secret key* X_c, X_j *and* X_k *of the MRS,* MS_j *and* PS_k *respectively and the session key* SK *between the* U_i *and* PS_k *even if the* $\widehat{A}$ *knows all the information including smart card and all the transmitted messages.*

Proof We develop an attacker $\widehat{A}$ (similar to theorem 1) who has the ability to derive the long-term confidential parameters like secret key of the MRS, MS_j and PS_k entity of our proposed protocol ($UAKPMS$). It is our assumption that an attacker not only knows all the smart card parameters $\langle REG_i, A_j, P_j, h(), H()\rangle$ by monitoring power consumption [23, 41], s/he also knows all the transmitted messages $\langle ID_{ms_j}, ID_k, F_i, D_i, G_i, L_i\rangle$, $\langle ID_k, O_j, S_j, Q_j, RAN_j, F_i\rangle$, $\langle T_k, RAN_k, V_k\rangle$ of our proposed protocol. The $(\widehat{A})$ then executes the algorithm $ALGO2^{HASH}_{\widehat{A},UAKPMS}$ for deriving $\langle X_c, X_j, X_k, SK\rangle$ of the MRS, MS_j, PS_k and the session key of the proposed protocol as given in the Algorithm 2. □

Algorithm 2 $ALGO2^{HASH}_{\widehat{A},UAKPMS}$

Input: $\langle REG_i, A_j, P_j, h(), H(), ID_{ms_j}, ID_k, F_i, D_i, G_i, L_i$
$ID_k, O_j, S_j, Q_j, RAN_j, F_i, T_k, RAN_k, V_k\rangle$
Output: 0 or 1.
(Traps the login request message during the login phase $\langle ID_{ms_j}, ID_k, F_i, D_i, G_i, L_i\rangle$, where $D_i = h(h(ID_i \parallel X_j) \parallel R_c)$, $G_i = ID_i \oplus E_i$, $L_i = E_i \oplus R_c$).
Calls Reveal oracle on input D_i for retrieving $\langle ID_i, X_j, R_c, E_i\rangle$ as $(ID_i^{'} \parallel X_j^{'} \parallel R_c^{'}, E_i^{'}, X_c^{'}) \leftarrow$ Reveal(D_i).
Computes $D_i^{*} = h((ID_i^{'} \parallel X_j^{'}) \parallel R_c^{'}$, $(X_j^{''} = h(ID_{msj} \parallel X_c^{'}))$, $G_i^{'} = ID_i^{'} \oplus E_i^{'}$, $L_i^{'} = E_i^{'} \oplus R_c^{'}$
if $(D_i^{'} == D_i)$ AND $(X_j^{''} == X_j^{'})$ AND $(G_i^{'} == G_i)$ AND $(L_i^{'} == L_i)$ **then**
  Calls Reveal oracle on input $Q_j^{'}$ for retrieving X_k, R_{ms} information as $(ID_i^{'} \parallel X_k^{'} \parallel R_{ms}^{'}) \leftarrow$ Reveal$(Q_j^{'})$
  Computes $N_j^{'} = h(ID_k \parallel X_k^{'} \parallel F_i)$ and $Q_j^{'} = h(ID_i^{'} \parallel X_k^{'} \parallel N_j^{'} \parallel R_{ms}^{'})$
  if $(Q_j^{'} == Q_j)$ **then**
    Computes $SK^{'} = h(ID_i^{'} \parallel)$
    Calls Reveal oracle on input (T_k) for retrieving session key SK as $ID_i^{'} \parallel X_k^{'} \parallel SK^{'} \parallel R_k^{'} \leftarrow$ Reveal$(T_k^{'})$.
    Computes $T_k^{''} = h(h(ID_i^{'} \parallel X_k^{'}) \parallel SK^{'})$
    if $(T_k^{''} == T_k^{'})$ **then**
      Accepts $\langle X_c^{'}, X_j^{'}, X_k^{'}\rangle$ as the correct secret key of the MRS, MS and PS respectively and the $SK^{'}$ is the correct session key between the patients and the physician servers.
      Return(1) Success.
    else
      Return(0) Failure
    else
      Return(0) Failure
    end if
  end if
  Return(0)
end if

We define the success probability for the algorithm 2 which is given below:

$$SUCC2^{HASH}_{A,UAKPMS} = Prb[ALGO2^{HASH}_{\widehat{A},UAKPMS} = 1] - 1,$$

where $Prb[E]$ is the probability of an event (E). The advantages function of the $ALGO2^{HASH}_{\widehat{A},UAKPMS}$ is

$$Adv2^{HASH}_{\widehat{A},UAKPMS}(t2, qr2) = Max_{\widehat{A}}[Adv2^{HASH}_{\widehat{A},UAKPMS}],$$

where the maximum is taken over all $\widehat{A}$ with the execution time $t2$ and the $qr2$ indicates that the number of queries made to the Reveal oracle. The proposed scheme is said to be provably secure against the $\widehat{A}$ for deriving $\langle X_c, X_j, X_k, SK\rangle$, if $Adv2^{HASH}_{\widehat{A},UAKPMS}(t2, qr2) \leq \epsilon$ for any small value $\epsilon > 0$. If the attacker has the ability based on the $ALGO2^{HASH}_{\widehat{A},UAKPMS}$, to invert the cryptographic one-way hash function, then only s/he can easily derive $\langle X_c, X_j, X_k, SK\rangle$ and win the game. However, it is computationally infeasible in polynomial time that is $Adv_{\widehat{A}^{HASH}(t)} \leq \epsilon$ for any small $\epsilon > 0$ (see section "Preliminaries"). Therefore, we have $Adv2^{HASH}_{\widehat{A},UAKPMS}(t2, qr2) \leq \epsilon$, as $Adv2^{HASH}_{\widehat{A},UAKPMS}(t2, qr2)$ depends on the advantage $Adv_{\widehat{A}^{HASH}(t)}$. This proves that the proposed scheme is secure for deriving $\langle X_c, X_j, X_k, SK\rangle$ against an attacker.

Brief description of *AVISPA* tool

AVISPA is considered as a widely-accepted for the formal security verification which measures whether the security protocol is *SAFE* or *UNSAFE* and Supports High Level Protocol Specification Language called as *HLPSL*. The structure of the *AVISPA* tool is shown in Fig. 2. Currently, *AVISPA* [4] implements four different back-ends and abstraction based methods which are integrated through the *HLPSL*. The First back-ends called the On-the-fly Model-Checker *(OFMC)* responsible for several symbolic techniques to explore the state space in a demand-driven way. The second back-end, called the *CL-AtSe* (Constraint-Logic-based Attack Searcher), provides a translation from

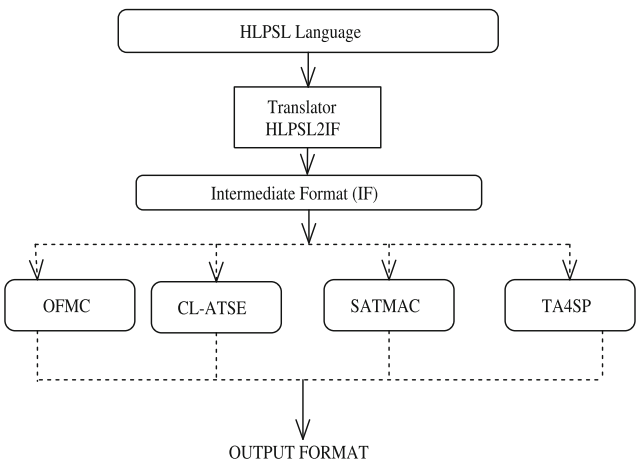

Fig. 2 Architecture of the *AVISPA* Tool

any security protocol specification written as transition relation in intermediate format *(IF)* into a set of constraints which are effectively used to find whether there are attacks on protocols. The third-one is called *SAT* based Model checker which generates a propositional formulae and then fed to a state-of-the-art *SAT* solver and any model found is translated back into an attack. The Tree Automata based on Automatic Approximations for the Analysis of Security Protocols *(TA4SP)* is the last back-ends of the *AVISPA* tool responsible for approximates the intruder knowledge by using regular tree languages. As mentioned earlier, the *HLPSL* specification is translated into the intermediate form *(IF)* using the hlpsl2if translator. Intermediate form *(IF)* is a lower level language than *HLPSL* is read directly by the back-ends to the *AVISPA* Tool. It may be noted that this intermediate translation step is transparent to user.

AVISPA is a role-oriented language in which each participants play a role during the protocol execution. Each roles are independent of the other, getting some initial information by parameters, communicating with the other roles by channels. The intruder is modeled using the Dolev-Yao [13] model with the possibility for the intruder to assume a legitimate role in a protocol run. The role system also describes the number of sessions, the number of principals and the roles. Based on the four back-ends, the *OUTPUT FORMAT (OF)* is generated and after successful execution, the *(OF)* describes the result whether the protocol is safe or unsafe or under what condition the output is obtained.

Brief specification of the proposed protocol

This section briefly discusses of our proposed authentication scheme for the roles of the U_i, the MRS, the MS, the PS, the session, the goal and the environment. In Fig. 3, we have implemented the role for the U_i in *HLPSL* language. In the user registration phase, the U_i initially sends registration message $\langle ID_i, PWD_i = h(ID_i \parallel PW_i), B_i \parallel$ to the medical registration server MRS through secure channel with the help of Snd() operation. The type declaration channel(dy) means that the channel is for the Dolev-Yao threat model. The declaration $secret(PWi, subs2, Ui)$ and $secret(IDi, subs3, Ui, MRS, MS, PS)$ indicate that the PWi is only known to the Ui and the IDi is kept secret permanently to the $\langle U_i, MRS, MS, PS\rangle$ respectively. The U_i receives the smart card with the information $\langle REG_i, A_j, P_j, h(), H()\rangle$ securely using the symmetric key SKu_is_j with the help of the Rcv() operation. During the login phase, the smart card generates a random nonce with the help of new() operation and sends $\langle ID_{ms_j}, ID_k, F_i, D_i, G_i, L_i\rangle$ to the MS_j via public channel. The declaration $witness(Ui, MS, alice\_mserver, Rc')$ indicates that the Ui has freshly

```
role alice (Ui, MRS, MS, PS : agent,
SKuisj : symmetric_key,
SK1: symmetric_key,
 % H is hash function
H: hash_func, Snd, Rcv: channel(dy))
played_by Ui
def=
local State : nat,
IDi, IDmsj, IDk, PWi, Bi, Xc, Xj, Xk, Fi,
PWDi, Aj, Pj, REGi, Rc, Rms, Rk: text,
Ci, Di, Ei, Gi, Li, Nj, Oj,Sj,RANj, Qj,
Tk, RANk, Vk, SK: message,
Inc : hash_func
const alice_server, server_mserver, mserver_pserver,
pserver_alice, alice_mserver, subs1, subs2, subs3,
subs4, subs5, subs6, subs7, subs8 : protocol_id
init State :=0
transition
1. State = 0 ∧ Rcv(start) =|>
State' := 1 ∧ PWDi' := H(IDi.PWi)
∧ Snd({IDi.PWDi'.Bi}_SKuisj)
 %send registration request message to MRS
∧ secret({PWi}, subs2, Ui)
∧ secret({IDi}, subs3, {Ui,MRS,MS,PS})
2. State = 1 ∧ Rcv({Fi'. REGi'. Aj'. Pj'}_SKuisj) =|>
 % Receive smart card securely from the MRS
 % Start login phase
State' := 2 ∧ Rc' := new()
∧ IDmsj' := new()
∧ IDk' := new()
∧ Ci' := xor(Aj', REGi')
∧ Di' := H(Ci'.Rc')
∧ Ei' := xor(Pj', H(REGi'.Fi'))
∧ Gi' := xor(IDi, Ei')
∧ Li' := xor(Ei', Rc')
∧ Snd(Fi'. Di'. Gi'. Li'. IDmsj'.IDk')
 %sends login message to the MRS through public channel
∧ witness(Ui, MS, alice_mserver, Rc')
/\secret({Rc'},subs6,{Ui,MS,PS})
 % Receive reply message from the PS
3. State = 2 ∧ Rcv(Tk'.RANk'.Vk') =|>
State' := 3 ∧  Rk' := new()
∧ request(PS, Ui, pserver_alice, Rk')
end role
```

Fig. 3 Role specification for the alice (Ui) of the proposed scheme in *HLPSL*

generated the value Rc' for the MS. During the authentication phase, the U_i receives $\langle T_k', RAN_k', V_k'\rangle$ with the help of the Rcv() operation. The declaration $request(PS, Ui, pserver\_alice, Rk')$ means that the Ui authenticates the PS.

In Fig. 4, we have presented the role for the MRS who only responsible for providing registration to the new user and medical server in *HLPSL* language. Initially, the MRS receives identity of the medical server with the help of the Rcv() operation and sends X_j to the MS_j through secure channel with the help of the Snd() operation. The secure channel indicates that the parameters is transmitted

through encrypted form X_j with the help of symmetric key $SK1$. The declaration $secret(X_j, subs1, MRS, MS)$ and $secret(X_c, subs4, MRS)$ indicate that the key X_j and X_c are kept secret permanently to the *(MRS, MS) and (MRS)* respectively. After that, the *(MRS)* receives the registration request message for the U_i and transmits a smart card with the information $\langle F_i, REG_i, A_j, P_j, h(), H()\rangle$ securely to the U_i.

In Fig. 5, we have presented the role for the *mserver* in *HLPSL* language. Initially, *mserver* generates an identity with the help of the new() operation and receives a message including secret key of the server and login message from the user. Then, the *mserver* generates a random number (Rms) with the help of the new() operation and sends $Snd(Oj'.Sj'.Qj'.IDk'.Fi'.RANj')$ to the physician server (PS_k) through public channel and transmits secret key (Xk) of the *pserver* via secure channel. The declaration $secret(Xk, subs5, MS, PS)$ and $secret(Rms', subs7, Ui, MS, PS)$ state that the parameters Xk and Rms are kept secret permanently to the MS, PS and Ui, MS, PS respectively. The declaration $witness(MS, PS, mserver\_pserver\_rms, Rms')$ tells that the (MS) generates a random number freshly for the *pserver* and $request(MS, PS, mserver\_pserver, Rms')$ indicates that the PS authenticates the MS.

In Fig. 6, we have presented the role for the *pserver* in *HLPSL* language. Initially, the *pserver* receives an authentication message $Rcv(Xk'\_SK1, Oj'.Sj'.Qj'.IDk.Fi'.RANj')$ including the secret key of the *pserver* from the MS. The *pserver* then generates a random number with the help of the new() operation and transmits $Snd(Tk'.RANk'.Vk')$ to the (U_i) through public channel. The declaration $witness(PS, Ui, pserver\_alice, Rk')$

```
role server (MRS, Ui, MS, PS  : agent,
SKuisj : symmetric_key,
SK1: symmetric_key,
% H is hash function
H : hash_func,
Snd, Rcv: channel(dy) )
played_by MRS
def=
local State : nat,
IDi, IDmsj, IDk, PWi, Bi, Xc, Xj, Xk, Fi,
PWDi, Aj, Pj, REGi, Rc, Rms, Rk: text,
Ci, Di, Ei, Gi, Li, Nj, Oj,Sj,RANj, Qj,
Tk, RANk, Vk, SK: message,
Inc : hash_func
const alice_server, server_mserver, mserver_pserver,
pserver_alice, alice_mserver, subs1, subs2, subs3,
subs4, subs5, subs6, subs7, subs8 : protocol_id
init State :=0
transition
1. State = 0 ∧ Rcv(IDmsj') =|>
State' := 1 ∧ secret({Xj}, subs1, {MRS, MS})
∧ secret({Xc}, subs4, {MRS})
∧ Xj' := H(IDmsj' . Xc)
∧ Snd({Xj'}_SK1)
% send secret key to the MRS securely
2. State = 1 ∧ Rcv({IDi.H(IDi.PWi).Bi}_SKuisj) =|>
State' := 2 ∧ secret({Xj}, subs1, {MRS, MS})
∧ secret({PWi}, subs2, Ui)
∧ secret({IDi}, subs3, {Ui, MRS, MS, PS })
∧ secret({Xc}, subs4, {MRS})
∧ Fi' := H(Bi)
∧ REGi' := H(IDi.H(IDi.PWi))
∧ Aj':= xor(H(IDi.Xj), REGi')
∧ Pj' := xor(H(IDmsj.Xj.Fi'), H(REGi'. Fi'))
∧ Snd({Fi'. REGi'. Aj'. Pj'}_SKuisj)
% Send registration reply message to the Ui
end role
```

Fig. 4 Role specification for the server (MRS) of the proposed scheme in *HLPSL*

```
role mserver (MS, Ui, MRS, PS : agent,
SKuisj : symmetric_key,
SK1: symmetric_key,
% H is hash function
H : hash_func,
Snd, Rcv: channel(dy) )
played_by MS
def=
local State : nat,
IDi, IDmsj, IDk, PWi, Bi, Xc, Xj, Xk, Fi,
PWDi, Aj, Pj, REGi, Rc, Rms, Rk: text,
Ci, Di, Ei, Gi, Li, Nj, Oj,Sj,RANj, Qj,
Tk, RANk, Vk, SK: message,
Inc : hash_func
const alice_server, server_mserver,
mserver_pserver, pserver_alice, alice_mserver,
subs1, subs2, subs3, subs4, subs5, subs6,subs7,subs8 : protocol_id
init State :=0
transition
1. State = 0 ∧ Rcv(start) =|>
State' := 1 ∧ IDmsj' := new()
∧ Snd(IDmsj')
2. State = 1 ∧ Rcv({Xj'}_SK1, Fi'.Di'.Gi'.Li'.IDmsj'.IDk') =|>
% Receive secret key including login message parameter
State' := 2 ∧ Rms' := new()
∧ Ei' := H(IDmsj.Xj'.Fi')
∧ IDi' := xor(Gi', Ei')
∧ Rc' := xor(Li', Ei')
∧ Xk' := H(IDk' . Xj')
∧ Nj' := H(IDk'.Xk'.Fi')
∧ Oj' := xor(IDi', Nj')
∧ Sj' := xor(H(IDk'.Xk'), Rms')
∧Qj'  := H(IDi' . Xk' . Nj' . Rms')
∧RANj' := xor(Rc',Rms')
∧ Snd (Oj'. Sj' . Qj'. IDk'. Fi'. RANj')
∧ Snd({Xk'}_SK1)
% send secret key Xk including authentication message to the PS
∧ secret({Xk}, subs5, {MRS,PS})
∧ secret({Rms'}, subs7, {Ui,MS,PS})
∧ witness(MS, PS, mserver_pserver_rms, Rms')
∧ request(MS, PS, mserver_pserver, Rms')
end role
```

Fig. 5 Role specification for the mserver (MS) of the proposed scheme in *HLPSL*

indicates that the $pserver$ generates freshly a random number for the Ui and the declaration $secret$ $(Rk', subs8, PS, Ui)$ indicates that the parameters Rk' is kept secret to the (PS, Ui).

In Fig. 7, we have presented the roles for the session, goal and the environment in *HLPSL* language. In the session segment, all the basic roles including the roles for the (Ui, MRS, MS) and the (PS) are instanced with concrete arguments. The environment section contains the global constant and composition of one or more session and the intruder knowledge is also given. The current version (2006/02/2013) of *HLPSL* supports the standard authentication and secrecy goals. In our implementation, the following eight secrecy goals and three authentications are verified.

1. The $secrecy\_of subs1$ represents that the key (Xj) is kept secret to only (MRS, MS).
2. The $secrecy\_of subs2$ represents that the password (PWi) is only known to (Ui).
3. The $secrecy\_of subs3$ indicates that the user's identity (IDi) is only known to all the entities of the proposed protocol except the third party.

```
role pserver (PS, Ui, MRS, MS  : agent,
SKuisj : symmetric_key,
SK1: symmetric_key,
 % H is hash function
H: hash_func,
Snd, Rcv: channel(dy) )
played_by PS
def=
local State : nat,
IDi, IDmsj, IDk, PWi, Bi, Xc, Xj, Xk, Fi,
PWDi, Aj, Pj, REGi, Rc, Rms, Rk: text,
Ci, Di, Ei, Gi, Li, Nj, Oj,Sj,RANj, Qj,
Tk, RANk, Vk, SK: message,
Inc : hash_func
const alice_server, server_mserver, mserver_pserver,
pserver_alice, alice_mserver,
subs1, subs2, subs3, subs4, subs5, subs6, subs7, subs8 : protocol_id
init State :=0
transition
 % Receive authentication message including secret key from the MS
1. State = 0 ∧ Rcv({Xk'}_SK1,Oj'. Sj' . Qj'. IDk. Fi'. RANj') =|>
State' := 1 ∧ Rk' := new()
∧ Nj' := H(IDk.Xk'.Fi')
∧ IDi' := xor(Oj', Nj')
∧ Rms' := xor(H(IDi'. Xk'), Sj')
∧ Rc' := xor(RANj', Rms')
∧ SK' := H(IDi'. IDk. Rc'. Rk')
∧ Tk':= H(H(IDi'. Xk'). SK')
∧ RANk' := xor(Rc', Rk')
∧ Vk' := xor(H(IDi'.Rk'))
∧ Snd(Tk'.RANk'.Vk')
 % send authentication message to the Ui
∧ witness(PS,Ui,pserver_alice, Rk')
∧ request(Ui, PS, pserver_alice, Rc')
∧ secret({Rk'}, subs8, {PS,Ui})
end role
```

Fig. 6 Role specification for the pserver (PS) of the proposed scheme in *HLPSL*

```
role session(Ui, MRS, MS, PS: agent,
SKuisj : symmetric_key,
SK1: symmetric_key,
H: hash_func)
def=
 local SI, SJ, RI, RJ, TI, TJ, PI, PJ: channel (dy)
 composition
alice(Ui, MRS, MS, PS, SKuisj, SK1, H, SI, RI)
∧ server(Ui, MRS, MS, PS, SKuisj, SK1, H, SJ, RJ)
∧ mserver(Ui, MRS, MS, PS, SKuisj, SK1, H, TI, TJ)
∧ pserver(Ui, MRS, MS, PS, SKuisj, SK1, H, PI, PJ)
end role
role environment()
def=
const ui, mrs,ms,ps: agent,
skuisj : symmetric_key,
sk1: symmetric_key,
h: hash_func,
idi, idmsj, idk, pwi, bi, xc, xj, xk, fi,
pwdi, aj, pj, regi, rc, rms, rk : text,
alice_server, server_mserver, mserver_pserver,
pserver_alice, alice_mserver, subs1, subs2, subs3,
subs4, subs5, subs6, subs7, subs8 : protocol_id
intruder_knowledge = {ui, mrs, ms, ps, h, fi,aj,pj,regi}
composition
session( ms, mrs, ui, ps, skuisj, sk1, h)
∧ session(ui, mrs, ms, ps, skuisj, sk1, h)
∧ session(ui, ms, ps, mrs, skuisj, sk1, h)
∧ session(ms, ps, ui, mrs, skuisj, sk1, h)
end role
goal
secrecy_of subs1
secrecy_of subs2
secrecy_of subs3
secrecy_of subs4
secrecy_of subs5
secrecy_of subs6
secrecy_of subs7
secrecy_of subs8
authentication_on alice_mserver_rc
authentication_on mserver_pserver_rms
authentication_on pserver_alice_rk
end goal
environment()
```

Fig. 7 Role specification for the session, goal and environment (S) of the proposed scheme in *HLPSL*

4. The $secrecy\_of subs4$ indicates that the (Xc) is only known to the (MRS).
5. The $secrecy\_of subs5$ indicates that the (Xk) is only known to the (PS, MS).
6. The $secrecy\_of subs6$ indicates that the random number (Rc') is only known to the (Ui, PS, MS).
7. The $secrecy\_of subs7$ indicates that the random number (Rms') is only known to (Ui, MS, PS).

8. The $secrecy\_of subs8$ indicates that the random number (Rk') is only known to the (Ui, PS).
9. The $authentication\_onalice\_mserver\_rc$ represents that the (Ui) generates a random number (rc), where (rc) is only known to (Ui) and if the (MS) receives it through message securely, (MS) then authenticates the (Ui).
10. The $authentication\_onmserver\_pserver\_rms$ represents that the (RMS) generates a random number (rms), where (rms) is only known to (RMS) and if the (PS) receives it through message securely, (PS) then authenticates the (RMS).
11. The $authentication\_onpserver\_alice\_rk$ represents that the (PS) generates a random number (rk), where (rk) is only known to (PS) and if the (Ui) receives it through message securely, (Ui) then authenticates the (PS).

Simulation result

In this section, we specify simulation result of our proposed scheme based on the widely-accepted two back-ends such as *OFMC* and *CL-AtSe* using the *AVISPA* web tool. The Figs. 8 and 9 confirm that the proposed protocol is *SAFE* under two back-ends *OFMC* and *CL-AtSe* respectively. Moreover, the simulation results using *AVISPA* clearly ensure that the proposed scheme is secure against active and passive attacks including replay and man-in-the-middle attacks.

Informal security analysis of the proposed protocol

In this section, we have analyzed the security of our proposed user authentication scheme informally for proving that the protocol provides strong security protection on the relevant security attacks. In the following, we justify several security attacks protection approach.

```
% OFMC
% Version of 2006/02/13
SUMMARY
SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
/home/avispa/web-interface-computation/./tempdir/workfileT6hy8b.if
GOAL
as_specified
BACKEND
OFMC
COMMENTS
STATISTICS
parseTime: 0.00s
searchTime: 0.22s
visitedNodes: 23 nodes
depth: 4 plies
```

Fig. 8 Simulation result for the *OFMC* back-end

```
SUMMARY
SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
TYPED_MODEL
PROTOCOL
/home/avispa/web-interface-computation/./tempdir/workfileGqYqkK.if

GOAL
As Specified
BACKEND
CL-AtSe

STATISTICS
Analysed : 2469 states
Reachable : 129 states
Translation: 0.61 seconds
Computation: 0.01 seconds
```

Fig. 9 Simulation result for the *CL-AtSe* back-end

Off-line identity-password guessing attack

A passive attacker may try to guess the user's password or identity in off-line after extracting all the smart card information $\langle ID_{ms_j}, A_j, P_j, REG_i, h(), H() \rangle$ and the communicating messages between the entities involved in the protocol. However, the attacker cannot successfully verify the guessed password PW^g or identity ID_i^g which are presented below:

- For verifying the guessed password PW^g using $REG_i = h(ID_i \parallel PW_i)$, an attacker requires the user's original identity ID_i. However, it is not possible by an attacker from the proposed protocol. If an attacker tries to guess both $\langle ID_i, PW_i \rangle$ parameters at a time, the probability of guessing is approximately $\frac{1}{2^{12n}}$, which implies that it is not feasible in polynomial time.
- The attacker may also try to guess using $\langle A_j, P_j \rangle$ parameters. However, these parameters contains additionally secret key X_j of the medical server. Therefore, the probability of guessing is so negligible that the attacker fails to guess the parameters.

Privileged insider attack

Most of the security system breaks due to insider attack. So, it is an important task of the protocol designer that always keeps user's confidential information secret from the server (though the server is trusted). If an insider of the system (system manager or administrator) gets the user's correct password by some means, then s/he may use that password in others account of the others server, as most of the users use same password for a set of accounts. In our protocol, the user does not submit the password

PW_i in its original form to the medical registration server during the user registration phase. The U_i only submits $PWD_i = h(ID_i \parallel PW_i)$. Therefore, an insider cannot extract the user's password. Moreover, an adversary cannot guess the user's password, as the probability of guessing two unknown parameters at a time is $\frac{1}{2^{|2n}}$, which is very negligible and infeasible in polynomial time.

Stolen smart card attack

An attacker can try to use the stolen smart card of the valid user after extracting the stored parameters of the smart card by monitoring the power consumption [23, 41]. To login successfully to a server, an attacker has to make valid login message $\langle ID_{ms_j}, ID_k, F_i, D_i, G_i, L_i \rangle$. However, the attacker cannot compute the Valid D_i, which is justified below:

- The parameter D_i is protected by the non-invertible cryptographic one-way hash function and is dependent on the $\langle ID_i, X_j, R_c \rangle$ parameters.
- The smart card of the U_i does not store ID_i. Moreover, the attacker cannot extract X_j from the transmitted login message and the known smart card parameters. Therefore, the proposed authentication protocol provides strong security on smart card stolen attack.

User-server impersonation attack

In this attack, upon receiving the transmitting messages of the protocol, an attacker may try to impersonate as a legitimate user or server after generating valid messages. However, the proposed scheme has strong security protection on the transmitted messages which are justified below:

- At first, an attacker tries to compute valid login message $\langle ID_{ms_j}, ID_k, F_i, D_i, G_i, L_i \rangle$ which will be authenticated to the medical server. However, the attacker cannot compute valid login parameter $\langle D_i \rangle$, as it requires the knowledge of the $\langle ID_i, X_j \rangle$. Therefore, the proposed protocol provides strong security on the login message.
- We supposed that the attacker traps transmitting message $\langle ID_k, O_j, S_j, Q_j, RAN_j, F_i \rangle$ between the MS and the PS and tries to impersonate as a valid MS to the PS. However, the attacker fails to compute valid above mentioned message, as s/he cannot compute valid $\langle Q_j \rangle$ parameter because of unknown parameters $\langle ID_i, X_k \rangle$, where X_k is shared between the MS and PS only.
- We again supposed that the attacker traps valid $\langle T_k, RAN_k, V_k \rangle$ message between the PS and the U_i and tries to impersonate as a legitimate PS to the U_i. However, the attacker cannot compute valid $T_k = h((ID_i \parallel X_k) \parallel SK)$ which requires the knowledge of $\langle ID_i, X_k \rangle$. Therefore, the attacker fails to impersonate as legitimacy entity of the proposed protocol.

Known key secrecy

It is our assumption that the session key $SK = h(ID_i \parallel ID_k \parallel R_c \parallel R_k)$ is compromised by an attacker who tries to establish previous session key of the proposed protocol. As the each SK is hashed with non-invertible cryptographic one-way hash function, therefore, no information(s) can be retrieved from the session key due to collision property of the hash function. Hence, the proposed scheme achieves known key secrecy property.

Session key agreement and verification

It is confirmed that the U_i and the PS both computes same session key $SK = h(ID_i \parallel ID_k \parallel R_c \parallel R_k)$ of the proposed protocol during the authentication phase. In Step A4, the PS computes $T_k = h(h(ID_i' \parallel X_k) \parallel SK)$ and transmits $\langle T_k \rangle$ to the U_i through public channel. Then, the U_i verifies the authenticity of the $\langle T_k \rangle$ parameter which ensures that the session key is verified. Therefore, the proposed protocol provides session key agreement and verification.

Session key discloser attack

The security of the session key $SK = h(ID_i \parallel ID_k \parallel R_c \parallel R_k)$ of the proposed protocol depends upon the difficulty of cryptographic one-way hash function. To compute the session key, attacker needs $\langle ID_i, R_c, R_k \rangle$ parameters from the proposed protocol. However, extraction of these parameters $\langle ID_i, R_c, R_k \rangle$ are not possible from the known parameters by the attacker. Therefore, the proposed protocol resists session key discloser attack.

Message freshness

Timestamp method is the another way for resisting replay attack of the proposed protocol. But, this method may sometimes suffer from clock synchronization problem. To overcome it, the proposed scheme requires global clock time, that is, the user and the medical server should maintain same time which requires extra cost of the protocol. For avoiding this problem, our proposed protocol uses random nonces instead of timestamp to verify the freshness of the message.

Table 2 Computation cost and functionality comparison of proposed scheme with existing related schemes

Schemes ⇒	[53]	[44]	[47]	[9]	[51]	[37]	[40]	Proposed
Login Phase	$4T_h$+$1T_e$	$7T_h$	$4T_h$+$2T_{spm}$	$4T_h$	$3T_h$	$2T_h$	$4T_h + 1T_e + 1T_{spm}$	$5T_h$
Authentication Phase	$4T_e$+$4T_h$	$24T_h$	$7T_h$+$4T_{spm}$	$12T_h$	$24T_h$	$25T_h$	$6T_h + 1T_{spm}$	$14T_h$
A1	√	√	×	√	×	×	√	√
A2	×	×	×	√	×	×	√	√
A3	√	√	×	×	×	×	√	√
A4	×	√	×	×	×	×	√	√
A5	√	×	×	√	√	×	√	√
Skey	×	×	×	√	√	√	√	√
MA	×	×	×	×	×	×	√	√
WPD	×	√	×	√	√	√	√	√
SKV	×	×	×	×	×	√	×	√
E/D	√	√	×	√	√	√	×	√

$A1$: Resist off-line password guessing attack, $A2$: Resist Insider attack, $A3$: User Impersonation Attack, $A4$: Session key discloser attack, $A5$: Resist replay attack, $Skey$: Session key agreement, MA: Satisfy mutual authentication, WPD: Early wrong password detection, SKV: Whether session key verification property achieved or not, E/D: Whether the protocol is independent of encryption/decryption algorithm or not, √: Yes, ×: No, T_h: Execution time for One-way hash function, T_e: Execution time for exponentiation operation, T_{spm}: Execution time for encryption/decryption operation.

No encryption/decryption

It is our great achievement that the proposed protocol does not use any cryptographic symmetric key encryption/decryption algorithms like *AES, RC4* etc.

Fast error detection

In the login or password change procedures, the smart card detects the error immediately if the attacker keys in the wrong biometric template, identity and password to the card reader. As a result, non-registered user cannot generate fake login message, which reduces congestion in the network and avoids extra computation and communication cost as well.

No verification table

The proposed protocol is independent of the password verifier table that means the entities (MRS, MS, PS) of the protocol does not store any verification table in the database of the server. Therefore, the $\widehat{A}$ has no way to get the secret information(s) of the entities.

Table 3 Communication cost and number of message transmission flow comparisons of the proposed scheme with related existing schemes

Schemes ⇓	Communication cost for login	Communication cost for Authentication	Communication mode
Yang et al. [53]	1472	1344	(2) $SC \rightarrow S_j, S_j \rightarrow SC$
Sood et al. [44]	896	1216	(5) $SC \rightarrow S_j, S_j \rightarrow CS, CS \rightarrow S_j, S_j \rightarrow SC, SC \rightarrow S_j$
Wang et al. [47]	320	256	(2) $SC \rightarrow S_j, S_j \rightarrow SC$
Chung et al. [9]	512	512	(2) $SC \rightarrow S_j, S_j \rightarrow SC$
Xue et al. [51]	768	2176	(4) $SC \rightarrow S_j, S_j \rightarrow CS, CS \rightarrow S_j, S_j \rightarrow SC$
Li et al. [37]	512	1664	(4) $SC \rightarrow S_j, S_j \rightarrow CS, CS \rightarrow S_j, S_j \rightarrow SC$
Proposed	768	1152	(3) $SC \rightarrow MS, MS \rightarrow PS, PS \rightarrow SC$

SC: Smart Card, S_j:Service provider server, CS: Control server, MS: Medical server, PS: Physician server

Performance evaluation

The proposed protocol handles several medical and physician servers and also provides medical resources to the many users efficiently after performing one-time registration to the medical registration server. Therefore, we have compared the performance of the proposed authentication scheme with other related existing multi-server based authentication schemes such as Yang et al. [53], Sood et al. [44], Wang et al. [47], Chunag et al. [9], Xue et al. [51], Li et al. [37] and Maitra et al. [40] etc. The computation and communication complexities are the most important factors to measure the performance of any user authentication and key agreement protocol and it would be more efficient if the complexities are less than the existing related schemes. This paper mainly uses cryptographic one-way hash funcion h(), xor ($\oplus$) and concatenate ($\|$) operation for designing our secure authentication scheme. As the cost for the xor and concatenate operations are negligible, we only consider one-way hash function in our comparison. It can reasonably be assumed that the length of the identity (user (ID_i), medical server (ID_{msj}), physician server (ID_k)), user password (PW_i), biometric (B_i), random nonces $\langle(R_c, R_{ms}, R_k)\rangle$ and message digest $h()$ take 128 bits each for measuring the communication cost of the proposed protocol.

In Table 2, we have presented security functionality comparison of the proposed protocol with other existing related protocols and it has been observed that none of the protocols are completely free from security weaknesses. However, the informal security analysis confirms that the proposed protocol provides strong security protection on the relevant attacks including identity-password guessing attacks, user-server impersonation attacks, insider attack, smart card stolen attack and session key discloser attack etc. The result of *AVISPA* simulation tool ensures that the proposed protocol is secure against passive and active attacks including replay and man-in-the-middle attacks.

In Table 3, we have summarized the computation and communication costs comparison of the proposed protocol with some others related existing protocols. After ensuring strong security of the proposed protocol, the Table 3 proves that the proposed authentication protocol is relatively better than existing related protocols in terms of computation and communication cost complexities.

Conclusion

Recently, many user authentication protocols have been proposed in the literature for accessing the single medical server, but still most of the protocols fail to achieve complete security requirements. In order to avoid multi-registrations and multi-smart cards, this paper have contributed a novel architecture and user authentication with key agreement security protocol for accessing multi-medical servers. We have then analyzed the security through formal and informal security analysis of the proposed authentication scheme. It has observed that the protocol satisfies all the desirable security attributes which are demonstrated in the security analysis. Furthermore, the simulation result has also presented for the formal security verification using the widely-accepted *AVISPA* tool and shown that the protocol is secure against passive and active attacks including the replay and man-in-the-middle attacks. The performance of the proposed protocol in terms of computation and communication overheads are also made and confirm that the protocol is relatively better than the related existing schemes. Considering efficiency and security, we conclude that the proposed protocol is appropriate for practical implementation for accessing the multi-medical servers. In the future, we aim to reduce complexities of the authentication scheme without compromising the security.

References

1. Amin, R., Cryptanalysis and an efficient secure id-based remote user authentication using smart card. *Int. J. Comput. Appl.* 75(13):43–48, 2013.
2. Amin, R., Maitra, T., Giri, D., Article: An improved efficient remote user authentication scheme in multi-server environment using smart card. *Int. J. Comput. Appl.* 69(22):1–6, 2013.
3. Amin, R., Maitra, T., Rana, S.P., An improvement of wang. et. al.'s remote user authentication scheme against smart card security breach. *Int. J. Comput. Appl.* 75(13):37–42, 2013.
4. Armando, A., Basin, D., Boichut, Y., Chevalier, Y., Compagna, L., Cuellar, J., Drielsma, P., Hem, P., Kouchnarenko, O., Mantovani, J., Mdersheim, S., von Oheimb, D., Rusinowitch, M., Santiago, J., Turuani, M., Vigan, L., Vigneron, L., The avispa tool for the automated validation of internet security protocols and applications. In: Computer Aided Verification, Vol. 3576, pp. 281–285: Lecture Notes in Computer Science, 2005.
5. Bhargav-Spantzel, A., Squicciarini, A.C., Modi, S., Young, M., Bertino, E., Elliott, S.J., Privacy preserving multi-factor authentication with biometric. *J. Comput. Secur.* 15(5):529–560, 2007.
6. Cao, T., and Zhai, J., Improved dynamic id-based authentication scheme for telecare medical information systems. *J. Med. Syst.* 37(2):9912, 2013. doi:10.1007/s10916-012-9912-5.
7. Chang, Y.F., Yu, S.H., Shiao, D.R., A uniqueness-and-anonymity-preserving remote user authentication scheme for connected health care. *J. Med. Syst.* 37(2):9902, 2013. doi:10.1007/s10916-012-9902-7.
8. Chen, H.M., Lo, J.W., Yeh, C.K., An efficient and secure dynamic id-based authentication scheme for telecare medical information systems. *J. Med. Syst.* 36(6):3907–3915, 2012.
9. Chuang, M.C., and Chen, M.C., An anonymous multi-server authenticated key agreement scheme based on trust computing

using smart cards and biometrics. *Expert Syst. Appl.* 41(4, Part 1):1411–1418, 2014.
10. Das, A., and Goswami, A., A secure and efficient uniqueness-and-anonymity-preserving remote user authentication scheme for connected health care. *J. Med. Syst.* 37(3):9948, 2013. doi:10.1007/s10916-013-9948-1.
11. Das, A.K., Analysis and improvement on an effcient biometric based remote user authentication scheme using smart cards. *IET Inf. Secur.* 5(3):145–151, 2011.
12. Debiao, H., Jianhua, C., Rui, Z., A more secure authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(3):1989–1995, 2012.
13. Dolev, D., and Yao, A.C., On the security of public key protocols. Information Theory. *IEEE Trans.* 29(2):198–208, 1983.
14. Fan, C.I., and Lin, Y.H., Provably secure remote truly three-factor authentication scheme with privacy protection on biometrics. Information Forensics and Security. *IEEE Trans.* 4(4):933–945, 2009.
15. Guo, C., and Chang, C.C., Chaotic maps-based password-authenticated key agreement using smart cards. *Commun. Nonlinear Sci. Numer. Simul.* 18(6):1433–1440, 2013.
16. Hao, X., Wang, J., Yang, Q., Yan, X., Li, P., A chaotic map-based authentication scheme for telecare medicine information systems. *J. Med. Syst.* 37(2):9919, 2013. doi:10.1007/s10916-012-9919-y.
17. Islam, S.H., and Biswas, G.P., A more efficient and secure id-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem. *J. Syst. Softw.* 84(11):1892–1898, 2011.
18. Jiang, Q., Ma, J., Lu, X., Tian, Y., Robust chaotic map-based authentication and key agreement scheme with strong anonymity for telecare medicine information systems. *J. Med. Syst.* 38(2):1–8, 2014. doi:10.1007/s10916-014-0012-6.
19. Jiang, Q., Ma, J., Ma, Z., Li, G., A privacy enhanced authentication scheme for telecare medical information systems. *J. Med. Syst.* 37(1):9897, 2013. doi:10.1007/s10916-012-9897-0.
20. Jina, A.T.B., Ling, D.N.C., Goh, A., Biohashing: Two factor authentication featuring fingerprint data and tokenised random number. *Pattern Recogn.* 37(11):2245–2255, 2004.
21. Khan, M.K., Kumari, S., Gupta, M., More efficient key-hash based fingerprint remote authentication scheme using mobile device. *Comput.* 96(9):793–816, 2014. doi:10.1007/s00607-013-0308-2.
22. Khan, M.K., and Zhang, J., Improving the security of a flexible biometric remote user authentication scheme. *Comput. Stand. Interfaces* 29(1):82–85, 2007.
23. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Advances in Cryptology CRYPTO 99, *Lecture Notes in Computer Science*, Vol. 1666, 1999.
24. Kumar, M., Gupta, M.K., Kumari, S., An improved efficient remote password authentication scheme with smart card over insecure networks. *Int. J. Netw. Secur.* 13(3):167–177, 2011.
25. Kumari, S., Gupta, M.K., Khan, M.K., Li, X., An improved timestamp-based password authentication scheme: comments, cryptanalysis, and improvement. *Secur. Commun. Netw.* 7:1921–1932, 2014. doi:10.1002/sec.906.
26. Kumari, S., Khan, M., Kumar, R., Cryptanalysis and improvement of a privacy enhanced scheme for telecare medical information systems. *J. Med. Syst.* 37(4):9952, 2013. doi:10.1007/s10916-013-9952-5.
27. Kumari, S., and Khan, M.K., More secure smart card based remote user password authentication scheme with user anonymity. *Secur. Commun. Netw.* 7:2039–2053, 2013. doi:10.1002/sec.916.
28. Kumari, S., and Khan, M.K., Cryptanalysis and improvement of 'a robust smart-card-based remote user password authentication scheme. *Int. J. Commun. Syst.* 27:3939–3955, 2014. doi:10.1002/dac.2590.
29. Kumari, S., Khan, M.K., Li, X., An improved remote user authentication scheme with key agreement. *Comput. & Electr. Eng.* 40(6):1997–2012, 2014. doi:10.1016/j.compeleceng.2014.05.007.
30. Kumari, S., Khan, M.K., Li, X., Wu, F., Design of a user anonymous password authentication scheme without smart card. *Int. J. Commun. Syst.* 27(10):609–618, 2014. doi:10.1002/dac.2853.
31. Lee, C.C., Hsu, C.W., Lai, Y.M., Vasilakos, A., An enhanced mobile-healthcare emergency system based on extended chaotic maps. *J. Med. Syst.* 37(5):9973, 2013. doi:10.1007/s10916-013-9973-0.
32. Lee, T.F., An efficient chaotic maps-based authentication and key agreement scheme using smartcards for telecare medicine information systems. *J. Med. Syst.* 37(6):1–9, 2013. doi:10.1007/s10916-013-9985-9.
33. Lee, T.F., Chang, I.P., Lin, T.H., Wang, C.C., A secure and efficient password- based user authentication scheme using smart cards for the integrated epr information system. *J. Med. Syst.* 37(3):3833–3838, 2013.
34. Li, C.T., and Hwang, M.S., An efficient biometrics-based remote user authentication scheme using smart cards. *J. Netw. Comput. Appl.* 33(1):1–5, 2010.
35. Li, C.T., Lee, C.C., Weng, C.Y., A secure chaotic maps and smart cards based password authentication and key agreement scheme with user anonymity for telecare medicine information systems. *J. Med. Syst.* 38(9):77, 2014. doi:10.1007/s10916-014-0077-2.
36. Li, X., Niu, J.W., Ma, J., Wang, W.D., Liu, C.L., Cryptanalysis and improvement of a biometrics-based remote user authentication scheme using smart cards. *J. Netw. Comput. Appl.* 34(1):73–79, 2011.
37. Li, X., Xiong, Y., Ma, J., Wang, W., An efficient and security dynamic identity based authentication protocol for multi-server architecture using smart cards. *J. Netw. Comput. Appl.* 35(2):763–769, 2012.
38. Lin, H.Y., On the security of a dynamic id-based authentication scheme for telecare medical information systems. *J. Med. Syst.* 37(2):1–5, 2013.
39. Lumini, A., and Nanni, L., Biohashing: Two factor authentication featuring fingerprint data and tokenised random number. *Pattern Recogn.* 40(3):1057–1065, 2007.
40. Maitra, T., and Giri, D., An efficient biometric and password-based remote user authentication using smart card for telecare medical information systems in multi-server environment. *J. Med. Syst.* 38(12):142, 2014. doi:10.1007/s10916-014-0142-x.
41. 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.
42. Mishra, D., Mukhopadhyay, S., Chaturvedi, A., Kumari, S., Khan, M., Cryptanalysis and improvement of yan et al.s biometric-based authentication scheme for telecare medicine information systems. *J. Med. Syst.* 38(6):24, 2014. doi:10.1007/s10916-014-0024-2.
43. 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):120, 2014. doi:10.1007/s10916-014-0120-3.
44. Sood, S.K., Sarje, A.K., Singh, K., A secure dynamic identity based authentication protocol for multi-server architecture. *J. Netw. Comput. Appl.* 34(2):609–618, 2011.
45. Tan, Z., An efficient biometrics-based authentication scheme for telecare medicine information systems. *Netw.* 2(3):200–204, 2013.
46. Tool, A.W.: http://www.avispa-project.org/web-interface/, 2014.
47. Wang, B., and Ma, M., A smart card based efficient and secured multi-server authentication scheme. *Wirel. Pers. Commun.* 68(2):361–378, 2013.

48. Wei, J., Hu, X., Liu, W., An improved authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(6):3597–3604, 2012.
49. 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.
50. Xie, Q., Zhang, J., Dong, N., Robust anonymous authentication scheme for telecare medical information systems. *J. Med. Syst.* 37(2):9911, 2013. doi:10.1007/s10916-012-9911-6.
51. Xue, K., Hong, P., 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.
52. Yan, X., Li, W., Li, P., Wang, J., Hao, X., Gong, P., A secure biometrics-based authentication scheme for telecare medicine information systems. *J. Med. Syst.* 37(5):1–6, 2013.
53. Yang, D., and Yang, B.: A biometric password-based multi-server authentication scheme with smart card. In: 2010 International Conference on, Computer Design and Applications (ICCDA). Vol. 5, pp. 554–559, 2010.
54. Zhu, Z., An efficient authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(6):3833–3838, 2012.