ITM Web of Conferences **13**, 01020 (2017)
CMES2017
DOI: 10.1051/itmconf/20171301020

A Three Factor Remote User Authentication Scheme Using Collision Resist Fuzzy Extractor in Single Server Environment

Debasis Giri[1] and *Tanmoy* Maitra[2,⋆]

[1] *Department of Computer Science and Engineering, Haldia Institute of Technology, Haldia-721657, India*
[2] *Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032, India*

Abstract. Due to rapid growth of online applications, it is needed to provide such a facility by which communicators can get the services by applying the applications in a secure way. As communications are done through an insecure channel like Internet, any adversary can trap and modify the communication messages. Only authentication procedure can overcome the aforementioned problem. Many researchers have proposed so many authentication schemes in this literature. But, this paper has shown that many of them are not usable in real world application scenarios because, the existing schemes cannot resist all the possible attacks. Therefore, this paper has proposed a three factor authentication scheme using hash function and fuzzy extractor. This paper has further analyzed the security of the proposed scheme using random oracle model. The analysis shows that the proposed scheme can resist all the possible attacks. Furthermore, comparison between proposed scheme and related existing schemes shows that the proposed scheme has better trade-off among storage, computational and communication costs.

1 Introduction

Nowadays, online applications like bill payment, banking system, telecare medical system, social networking, e-voting and so on are rapidly used for their easy and efficient access. All the applications are going through a client/server environment and communications are done through public channel like Internet because of availability of public bandwidth. Therefore, all communication messages of the applications are public. As a result, any one can trap and modify the communication messages. For this purpose, in such communication system, authentication scheme is rapidly used by which after verifying the communicators and their messages, a secure communication can be done through public channel. In this regard, smart card and password based user authentication scheme is very much popular for online communication system. However, a suitable smart card based authentication scheme should satisfy the following property:

Low cost: Computational cost, communication cost and smart card storage cost are three basic network parameters to measure performances of an authentication scheme. Therefore, it has to considered that these three parameters are reduced as much possible when an authentication scheme is going to be designed.

Prevention of security attacks: During communication through public channel, it is needed to secure the message from outsider adversary. An authentication scheme needs to be designed such a way by which from the communication messages, any adversary will unable to extract useful information and the scheme can resist security attacks.

Session key agreement: After authentication in both ways, a common secret session key is needed to carry on the communications within the same session after encrypting the paintext messages by the session key.

Mutual authentication: Mutual authentication is a essential property for an authentication scheme by which all the communicators can authenticate or verify each other.

Efficient login procedure: For an authentication scheme, it is needed that smart card checks the wrong inputs before going to send a login message to server. By checking wrong inputs in login phase, extra communication overhead can be avoided. Therefore, it is crucial property of a good authentication scheme.

⋆e-mail: tanmoy.maitra@live.com

DOI: 10.1051/itmconf/20171301020

Efficient password change procedure: A valid user can change the password freely and securely without taking any help from server. For this purpose, the smart card should verify the old password in the password change phase, so that an unauthorized user cannot change the authorized user's password even if it gets the valid users' smart card. The afore mentioned property should present in an authentication scheme.

Traceability: In an authentication scheme, it is also needed to trace the sender of a message for corresponding receiver [1]. Otherwise, any one can mount denial of service (DoS) attack.

Furthermore, to enhanced the security of password based authentication scheme, biometric feature (i.e, finger print, ires, retina etc.) [2] is added with the password. Therefore, this research focus to design a biometric plus password based efficient authentication scheme by considering all the afore mentioned properties.

1.1 Literature Survey

A brief survey of existing authentication schemes is described in this section. First Lamport [3] proposed a password-based authentication scheme based on one way hash function. However, Shimizu et al. [4] showed that the Lamport's scheme [3] suffers from different attacks. After that so many remote user authentication schemes [5–16] have been proposed in this regard which are based on only password. But, the researchers have considered biometric feature [2] with the password to enhance the security label. Therefore, many researchers have proposed biometric and password based authentication schemes in [17–28]. Li and Hwang [17] proposed a biometrics-based remote user authentication scheme in 2010. However, in 2011, Das [18] showed that Li and Hwang's scheme [17] had flaws in the login phase, authentication phase and password change phase and therefore, Das also proposed an authentication scheme. An [19] showed that Das's scheme [18] cannot resist the server masquerading attack, user impersonation attack, password guessing attack and insider attack, and so proposed an improved scheme. Li et al. [20] found that An's scheme [19] suffered from the denial-of-service (DoS) attack, the forgery attack and also did not provide forward secrecy. Furthermore, in 2013, Lee and Hsu [21] pointed out that Das's scheme [18] is also suffering from privileged insider attack and the off-line password guessing attack. Therefore, Lee and Hsu [21] proposed a biometric based authentication scheme to overcome the weaknesses of Das's scheme [18]. In 2013, Tan [22] proposed a three-factor authentication scheme. But, Yan et al. [23] pointed out that Tan's scheme [22] is vulnerable to the Denial-of-Service (DoS) attack. However, recently, Mishra et al. [24] showed that Yan et al.'s scheme [23] suffers from off-line password guessing attack and has inefficient login and password change phases. Huang et al. [29] proposed an authentication scheme based on RSA. But Amin et al. [30] proved that Huang et al.'s scheme [29] unable to protect forgery attack and also introduced an authentication protocol in [30].

1.2 Contribution

This paper proposes a three factor authentication scheme using hash function and fuzzy extractor, where three factor means (1) users' password, (2) users' biometric and (3) smart card. This paper further analyzes the security of the proposed scheme using random oracle model. The analysis shows that an adversary cannot mount any attacks on the proposed scheme due to hardness of inversion of one-way hash function as well as it has to solve hardness of fuzzy factor. Furthermore, comparison between proposed scheme and related existing schemes shows that the proposed scheme has better trade-off among storage, computational and communication costs. It is a great contribution that the proposed scheme resists all the possible attacks with better trade-off among different costs.

1.3 Road Map

This section describes a road map which has been followed throughout this paper. Section 2 briefly introduces some preliminary mathematical concepts for introducing the proposed scheme. Section 3 describes a network model and an adversary model to analyze the proposed scheme. A proposed scheme is described in section 4. Section 5 describes cryptanalysis of the proposed scheme and Section 6 compares the performances of the proposed scheme with previously published schemes. Conclusion of this paper appears in section 7.

2 Preliminaries

In this section, a briefly review the basic concepts of cryptographic one-way hash function and collision resist fuzzy extractor are introduced.
Definition 1. A collision resistant cryptographic one-way hash function [25, 27] maps a string of arbitrary length to a string of fixed length called the hashed value. It can be symbolized as: $H: A \rightarrow B$, where A is a binary string of

DOI: 10.1051/itmconf/20171301020

arbitrary length and B is a binary string of fixed length n. If $Adv_{\mathcal{A}}^{H}(t_1)$ is the advantage to an adversary $\mathcal{A}$ to choose a random pair $(a, b) \in A \times A$ such that $H(a) = H(b)$, where $a \neq b$ for the time duration t_1, it can be considered that $Adv_{\mathcal{A}}^{H}(t_1)$ is the probability in the advantage which is computed over the random choices made by the adversary $\mathcal{A}$ for the time duration t_1. Then the cryptographic one-way hash function $H(\cdot)$ is called collision-resistant, if $Adv_{\mathcal{A}}^{H}(t_1) \leq \xi_1$, for any small $\xi_1 > 0$. $Adv_{\mathcal{A}}^{H}(t_1)$ is represented as:

$$Adv_{\mathcal{A}}^{H}(t_1) = Pr\Big[(a, b) \in_R A \times A \mid (a \neq b) \ \wedge \ H(a) \ = \ H(b)\Big], \tag{1}$$

where $Pr[\mathcal{E}]$ denotes the random event $\mathcal{E}$.

Definition 2. A collision resistant fuzzy extractor [2, 27] can be model as a procedure, known as Gen, which takes a binary string say, B of some metric space M as an input, where $M \in \{0, 1\}^k$, for some k bits and produces a random string say, $\phi \in_R \{0, 1\}^n$, for some n bits and an auxiliary string say, $\theta \in_R \{0, 1\}^r$, for some r bits, where $r = k$ or n bits. It can be mathematically represented by $Gen : M \rightarrow \phi \times \theta$. Another procedure, known as Rep, takes a binary string say, B' of the metric space $M \in \{0, 1\}^k$, where $B \neq B'$ and a uniform distribution binary string say, $\theta' \in_R \{0, 1\}^r$ to produce the random string $\phi' \in_R \{0, 1\}^n$, symbolized as $Rep : M \times \theta' \rightarrow \phi'$. If $Adv_{\mathcal{A}}^{FE}(t_2)$ is the advantage to an adversary $\mathcal{A}$ to choose a pair $(B, B') \in_R M \times M$ randomly such that $des(B, B') \leq \delta d$, $Gen(B)$ = $Gen(B')$ and $Rep(B, \theta) = Rep(B', \theta')$, where δd is the difference tolerance level and $B \neq B'$ for the time duration t_2, it can be considered that $Adv_{\mathcal{A}}^{FE}(t_2)$ is the probability that the advantage is computed over the random choices made by $\mathcal{A}$ for the time duration t_2. Then the fuzzy extractor FE is called collision-resistant, if $Adv_{\mathcal{A}}^{FE}(t_2) \leq \xi_2$, for any small $\xi_2 > 0$. $Adv_{\mathcal{A}}^{FE}(t_2)$ is represented as:

$$\begin{aligned} Adv_{\mathcal{A}}^{EF}(t_2) = Pr\Big[&(B, B') \in_R M \times M \mid (B \neq B') \ \wedge \ des(B, B') \leq \delta d \\ &\wedge Gen(B) = Gen(B') \ \wedge \ Rep(B, \theta) = Rep(B', \theta')\Big], \end{aligned} \tag{2}$$

for all probabilistic polynomial-time algorithms Gen and Rep.

3 Model

This section will introduce two following models:

Network Model: Architecture of the proposed scheme is shown in Figure 1 where, users have to register to a remote server to get their smart card which is known as registration procedure (see Figure 1(a)). Whenever the registered users want to get service from the remote server by accessing their smart card through public channel like Internet, the smart card sends a login request message to the remote server. After verifying the login request message, the remote server sends corresponding reply message to the sender. After receiving the reply message, the corresponding smart card checks the validity of the reply message. Upon receiving correct reply message, both user and the remote server agree for a shared secret session key (See Figure 1(b)).

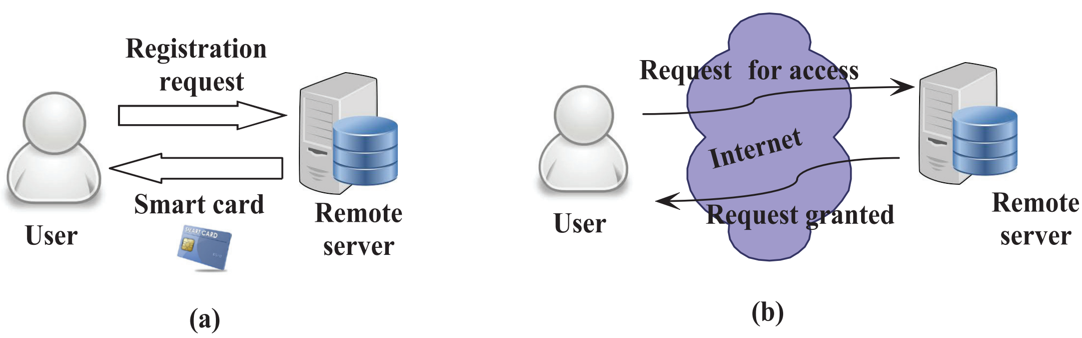

Figure 1. Network architecture of proposed scheme (a) registration procedure and (b) login and authentication procedure

Adversary Model: To analyze the security of the proposed scheme, Dolev-Yao threat model [31] has been considered in which the communicating parties communicate through an insecure channel. Therefore, an adversary $\mathcal{A}$ can trap the transmitted messages over the public or insecure channel, and furthermore he/she can modify, delete or change the contents of the transmitted messages. The adversary $\mathcal{A}$ also obtains the information which are stored in the user's smart card by monitoring the power consumption [32, 33]. Generally, identity and password of the user are low entropy in cryptography, that means the adversary can guess the identity and password individually using dictionary attack in polynomial time. But, the adversary cannot guess identity and password simultaneously in on-line/off-line within a polynomial time as pointed out in [34]. According to our adversary model, we consider two following cases:

- *Case 1.* A third party from outside of the system tries to mount various attacks on authentication system as an adversary $\mathcal{A}$.
- *Case 2.* A registered user from inside of the system tries to extract secret information of the server by which he/she can mount various attacks on authentication system as an other user or adversary $\widehat{\mathcal{A}}$.

4 Proposed Scheme

This section proposes an authentication scheme. A nomenclature is given in Table 1 to introduce the proposed scheme. The proposed scheme consists of five phases namely, 1) initialization phase, 2) registration phase, 3) login phase, 4) authentication and key agreement phase and 5) password update phase. The phases are as follows:

Table 1. Nomenclature

Term	***Usage***
U_i	*i-th* Patient
RS	Remote server
pw_i	Password of user U_i
ID_i	Identity of user U_i
B_i	Biometric parameter of U_i
r_i	Random number chosen by smart card
y_i	Random number chosen by RS
T	Current timestamp
$des(\cdot)$	Distance measurement function
δd	Estimated difference
X'	Parameter X computed or extracted by smart card
X^*	Parameter X computed or extracted by RS
δT	Estimated timestamp
SK_i	Shared session key between U_i and RS
$H(\cdot)$	Cryptographic one-way hash function
s	Secret key of remote server
$\|\|$	Concatenation operation
$\oplus$	Bit wise XOR operation

4.1 Initialization Phase

A remote server RS runs algorithm $\mathcal{G}$ to compute a large prime number q. Then it selects a random number s such that $s \in_R Z_q^*$. It further chooses a collision resist cryptographic one-way hash function $H : \{0, 1\}^* \rightarrow \{0, 1\}^n$, where n is a fixed length integer number. Finally, RS publishes $H(\cdot)$ as public and keeps s as secret.

4.2 Registration Phase

Whenever a new user U_i wants to register to the remote server RS, this phase is invoked. This phase is as follows:

1. The user inputs their biometric feature (i.e., finger print) to a sensor. The sensor generates a corresponding biometric information B_i and provides it to U_i.

2. The user U_i chooses an identity ID_i, password pw_i and generates an unique pair (θ_i, ϕ_i) from B_i by computing $(\phi_i, \theta_i) \leftarrow Gen(B_i)$. U_i then computes $pwr_i = H(pw_i||\phi_i)$ and sends $\langle ID_i, pwr_i\rangle$ to RS through a secure channel.

U_i	RS
Input biometrics to a sensor to get biometric information B_i	
Choose ID_i and pw_i	
Compute $(\phi_i, \theta_i) \leftarrow Gen(B_i)$	
Calculate $pwr_i = H(pw_i \| \phi_i)$	
$\xrightarrow{\langle ID_i, pwr_i \rangle}$	Compute $A_i = H(s \| ID_i)$, $D_i = H(A_i)$
	Check $D_i = ?$ database
	Calculate $C_i = A_i \oplus pwr_i$, store D_i into its database
Calculate $\widehat{B_i} = B_i \oplus H(ID_i \| pw_i)$	$\xleftarrow{\langle C_i, D_i, des(\cdot) \rangle}$
$\widehat{\theta_i} = \theta_i \oplus H(ID_i \| pw_i)$	
Reserve $\langle C_i, D_i, \widehat{B_i}, \widehat{\theta_i}, des(\cdot) \rangle$ into smart card	$\rightarrow$ Secure channel

Figure 2. Registration phase of the proposed scheme

3. After getting a registration request $\langle ID_i, pwr_i \rangle$ from U_i, RS computes $A_i = H(s \| ID_i)$ and $D_i = H(A_i)$. RS then checks D_i is present in its database or not. If it is present in its database, RS sends a decline message to U_i because, ID_i is already used by another user. Therefore, the user U_i has to choose another identity until fresh identity is not obtained. If D_i is not present in its database, RS computes $C_i = A_i \oplus pwr_i$ and stores parameters $\langle C_i, D_i, des(\cdot) \rangle$ into the memory of smart card, where $des(\cdot)$ is a distance measurement function. Then RS issues the smart card for U_i and sends it through a secure channel or by person. RS then updates its database by adding D_i into the list.

4. After getting the smart card, U_i inserts it into a terminal or card reader and submits their identity ID_i and password pw_i.

5. The smart card computes $\widehat{B_i} = B_i \oplus H(ID_i \| pw_i)$, $\widehat{\theta_i} = \theta_i \oplus H(ID_i \| pw_i)$. Finally, the smart card stores $\langle \widehat{B_i}, \widehat{\theta_i} \rangle$ into its memory. Note that, smart card stored parameters are $\langle C_i, D_i, \widehat{B_i}, \widehat{\theta_i}, des(\cdot) \rangle$.

Figure 2 shows the registration phase of the proposed scheme.

4.3 Login Phase

Whenever a registered user U_i wants to access the remote server, this phase is invoked. U_i inserts their smart card into a card reader or terminal and provides their biometric information B_i^* through sensor, identity ID_i and password pw_i to the smart card. The smart card then executes following steps:

1. The smart card computes $B_i' = \widehat{B_i} \oplus H(ID_i \| pw_i)$ and checks $des(B_i^*, B_i') \leq \delta d$. If it does not hold, the smart card rejects U_i; otherwise, it follows next step.

2. The smart card computes $\theta_i' = \widehat{\theta_i} \oplus H(ID_i \| pw_i)$, $\phi_i' \leftarrow Rep(B_i^*, \theta_i')$, $pwr_i' = H(pw_i \| \phi_i')$, $A_i' = C_i \oplus pwr_i'$, $D_i' = H(A_i')$ and checks computed D_i' and stored D_i are equal or not. If equality does not hold, the smart card rejects U_i; otherwise, it follows next step.

3. The smart card chooses a random number $r_i \in_R Z_q^*$ and further computes $E_i = r_i \oplus A_i'$, $G_i = T_i^1 \oplus A_i' \oplus H(r_i)$ and $F_i = H(r_i \| A_i' \| T_i^1)$, where T_i^1 is the current login timestamp of U_i. The user U_i then sends a login request message $\langle ID_i, G_i, F_i, E_i \rangle$ to the registration server RS through a public channel.

Figure 3 shows the login phase of the proposed scheme.

4.4 Authentication and Key Agreement Phase

After receiving the login request message $\langle ID_i, G_i, F_i, E_i \rangle$ from the user U_i at timestamp T_s, the remote server RS computes following steps:

1. RS computes $A_i^* = H(s \| ID_i)$, $r_i^* = A_i^* \oplus E_i$, $T_i^{1^*} = A_i^* \oplus G_i \oplus H(r_i^*)$ and checks $(T_s - T_i^{1^*}) \leq \delta T$. It it does not hold, RS rejects U_i; otherwise, executes the next step.

U_i	**Terminal**	RS
Input ID_i, pw_i and biometric B_i^*		
	Compute $B_i' = \widehat{B_i} \oplus H(ID_i \| pw_i)$	
	Check $des(B_i^*, B_I') \leq \delta d$	
	Compute $\theta_i' = \widehat{\theta_i} \oplus H(ID_i \| pw_i)$, $\phi_i' \leftarrow Rep(B_i^*, \theta_i')$	
	$pwr_i' = H(pw_i \| \phi_i')$, $A_i' = C_i \oplus pwr_i'$	
	$D_i' = H(A_i')$	
	Check computed D_i' =? stored D_i	
	Choose random number $r_i \in_R Z_q^*$	
	Compute $E_i = r_i \oplus A_i'$, $G_i = T_i^1 \oplus A_i' \oplus H(r_i)$	
	$F_i = H(r_i \| A_i' \| T_i^1)$	
	$\xrightarrow{\langle ID_i, G_i, F_i, E_i \rangle}$	$\rightarrow$ Insecure channel

Figure 3. Login phase of the proposed scheme

Terminal	RS
$\xrightarrow{\langle ID_i, G_i, F_i, E_i \rangle}$	Compute $A_i^* = H(s \| ID_i)$, $T_i^{1^*} = A_i^* \oplus G_i$
	Check $(T_s - T_i^{1^*}) \leq \delta T$
	Compute $r_i^* = A_i^* \oplus E_i$, $F_i^* = H(r_i^* \| A_i^* \| T_i^{1^*})$
	Check computed F_i^* =? received F_i
	Chooses a random number $y_i \in_R Z_q^*$
	Compute $Q_i = A_i^* \oplus y_i$, $K_i = T_s \oplus A_i^* \oplus H(y_i)$
Compute $y_i' = A_i' \oplus Q_i$, $T_s' = A_i' \oplus K_i \oplus y_i'$	$SK_i = H(y_i \| r_i^*)$, $L_i = H(T_s \| SK_i \| A_i^*)$
Check $(T_i^2 - T_s') \leq \delta T$	$\xleftarrow{\langle Q_i, L_i, K_i \rangle}$
Compute $SK_i' = H(y_i' \| r_i)$	
$L_i' = H(T_s' \| SK_i' \| A_i')$	
Check computed L_i' =? received L_i	
U_i agrees upon the shared secret key SK_i	$\rightarrow$ Public channel

Figure 4. Authentication and key agreement phase of the proposed scheme

2. RS computes $F_i^* = H(r_i^* \| A_i^* \| T_i^{1^*})$ and checks computed F_i^* and received F_i are equal or not. If it does not hold, RS rejects login request message of U_i; otherwise, follows the next step.

3. RS chooses a random number $y_i \in_R Z_q^*$ and further computes $Q_i = A_i^* \oplus y_i$, $K_i = T_s \oplus A_i \oplus H(y_i)$, $SK_i = H(y_i \| r_i^*)$, $L_i = H(T_s \| SK_i \| A_i^*)$ and sends a reply message $\langle Q_i, L_i, K_i \rangle$ to U_i through a public channel. RS accepts SK_i as a shared secret session key.

After receiving the reply message $\langle Q_i, L_i, K_i \rangle$ from RS at timestamp T_i^2, the smart card of the user U_i further executes the following steps to verify the reply message of RS:

1. The smart card computes $y_i' = A_i' \oplus Q_i$, $T_s' = A_i' \oplus K_i \oplus H(y_i')$ and checks $(T_i^2 - T_s') \leq \delta T$. It it does not hold, the smart card rejects the reply message; otherwise, executes the next step.

2. The smart card computes $SK_i' = H(y_i' \| r_i)$, $L_i' = H(T_s' \| SK_i' \| A_i')$ and checks computed L_i' and received L_i are equal or not. If they are equal, the user U_i agrees upon the shared secret key SK_i; otherwise, rejects the reply message.

Figure 4 shows the authentication and key agreement phase of the proposed scheme.

4.5 Password Update Phase

Whenever a user U_i wants to change their password, this phase is invoked. U_i inserts their smart card into a card reader or terminal and provides their biometric information B_i^* through sensor, identity ID_i and password pw_i to the smart card. The smart card then executes following steps:

1. The smart card computes $B_i' = \widehat{B_i} \oplus H(ID_i \| pw_i)$ and checks $des(B_i^*, B_I') \leq \delta d$. If it does not hold, the smart card rejects U_i; otherwise, it follows next step.

U_i	**Terminal**
Input ID_i, pw_i and biometric B_i	
	Compute $B'_i = \widehat{B_i} \oplus H(ID_i \Vert pw_i)$ Check $des(B^*_i, B'_I) \leq \delta d$ $\theta'_i = \widehat{\theta_i} \oplus H(ID_i \Vert pw_i)$, $\phi'_i \leftarrow Rep(B^*_i, \theta'_i)$ $pwr'_i = H(pw_i \Vert \phi'_i)$, $A'_i = C_i \oplus pwr'_i$ $D'_i = H(A'_i)$ Check computed D'_i =? stored D_i
Input a new $pw_i^{[new]}$	
	Compute $pwr_i^{[new]} = H(pw_i^{[new]} \Vert \phi'_i)$ $C_i^{[new]} = A'_i \oplus pwr_i^{[new]}$, $\widehat{B_i^{[new]}} = B'_i \oplus H(ID_i \Vert pw_i^{[new]})$ $\widehat{\theta_i^{[new]}} = \theta'_i \oplus H(ID_i \Vert pw_i^{[new]})$ store $C_i^{[new]}$, $\widehat{B_i^{[new]}}$, $\widehat{\theta_i^{[new]}}$ by replacing C_i, $\widehat{B_i}$, $\widehat{\theta_i}$

Figure 5. Password update phase of the proposed scheme

2. The smart card computes $\theta'_i = \widehat{\theta_i} \oplus H(ID_i\|pw_i)$, $\phi'_i \leftarrow Rep(B^*_i, \theta'_i)$, $pwr'_i = H(pw_i\|\phi'_i)$, $A'_i = C_i \oplus pwr'_i$, $D'_i = H(A'_i)$ and checks computed D'_i and stored D_i are equal or not. If equality does not hold, the smart card rejects U_i; otherwise, gives permission to enter their new password.

3. The user U_i selects their new password $pw_i^{[new]}$ and proves it to the smart card. The smart card then further proceeds to next step.

4. The smart card computes $pwr_i^{[new]} = H(pw_i^{[new]}\|\phi'_i)$, $C_i^{[new]} = A'_i \oplus pwr_i^{[new]}$, $\widehat{B_i^{[new]}} = B'_i \oplus H(ID_i\|pw_i^{[new]})$ and $\widehat{\theta_i^{[new]}} = \theta'_i \oplus H(ID_i\|pw_i^{[new]})$. The smart card then stores $C_i^{[new]}$, $\widehat{B_i^{[new]}}$ and $\widehat{\theta_i^{[new]}}$ in the place of C_i, $\widehat{B_i}$ and $\widehat{\theta_i}$ respectively into the memory of smart card.

Figure 5 shows the password update phase of the proposed scheme.

5 Security Analysis of Proposed Scheme

The formal security analysis of the proposed scheme under the random oracle model is presented in this section. This security analysis uses the formal security analysis under the generic group model of cryptography. In the following, this work defines random oracles for the formal security analysis of the proposed scheme:

- $Oracle\mathcal{H}$ is a random oracle which maintains a tuple $\langle x, y\rangle$ such that $y = H(x)$. It returns x from y upon receiving a query (qH, y) if $\langle x, y\rangle$ is present in the tuple; otherwise returns a random number r_1. Then it stores a new entry $\langle r_1, y\rangle$ into its tuple.
- $Oracle\mathcal{FE}$ is a random oracle which contains two parts:

  1. $Oracle\mathcal{FE}_{Gen}$ unconditionally outputs the pair (ϕ, θ) from the corresponding tuple $\langle B, \phi, \theta\rangle$ upon receiving a query $(qGen, B)$ such that $(\phi, \theta) \leftarrow Gen(B)$ if $\langle B, \phi, \theta\rangle$ is present in its tuple; otherwise returns two random numbers r_2 and r_3. Then it stores new entry $\langle B, r_2, r_3\rangle$ into its tuple.
  2. $Oracle\mathcal{FE}_{Rep}$ unconditionally outputs ϕ from the corresponding tuple $\langle B', \phi, \theta\rangle$ upon receiving a query $(qRep, B', \theta)$ such that $\phi \leftarrow Rep(B', \theta)$ if $\langle B', \phi, \theta\rangle$ is present in its tuple; otherwise returns random number r_4. Then it stores new entry $\langle B', r_4, \theta\rangle$ into its tuple.

Theorem 1. *Under the assumption that a cryptographic one-way hash function $H(\cdot)$ and fuzzy extractor FE act as random oracles, the proposed scheme is provably secure against an adversary $\mathcal{A}$ for deriving the password pw_i and biometric parameter B_i of a user U_i even if the adversary $\mathcal{A}$ gets parameters that are stored into the memory of U_i's smart card and traps the communication messages between U_i and the remote server RS.*

Proof 1. This research construct an adversary $\mathcal{A}$ who has the ability to derive the password pw_i and biometric parameter B_i of a user U_i. For this purpose, this research assumes that the smart card of a user U_i is lost or stolen. Thus, the adversary $\mathcal{A}$ can extract the stored parameters $\langle C_i, D_i, \widehat{B_i}, \widehat{\theta_i}\rangle$ from the memory of the smart card of the user U_i by power monitoring [32][33]. The adversary $\mathcal{A}$ also traps login request message $\langle ID_i, G_i, F_i, E_i\rangle$ and a reply message $\langle Q_i, L_i, K_i\rangle$. The adversary $\mathcal{A}$ runs the experiment, $EXP1^{oracle}_{\mathcal{A},\,TFUAS}$ for our three factor user authentication scheme (TFUAS) to derive the password pw_i and biometric parameter B_i of the user U_i as given in the Algorithm 1.

Algorithm 1 $EXP1^{oracle}_{\mathcal{A},\,TFUAS}$

Input: $C_i, D_i, \widehat{B_i}, \widehat{\theta_i}, ID_i, G_i, F_i, E_i, Q_i, L_i, K_i$
Output: 0 or 1

Calls $Oracle\mathcal{H}$ on the input D_i to retrieve the information $A_i = H(s\|ID_i)$ as $(A_i^*) \leftarrow Oracle\mathcal{H}(D_i)$
Calls $Oracle\mathcal{H}$ on the input F_i to retrieve the information A_i, r_i and T_i^1 as $(r_i^* \parallel A_i^{**}\|T_i^{1^*}) \leftarrow Oracle\mathcal{H}(F_i)$
Calls $Oracle\mathcal{H}$ on the input L_i to retrieve the information SK_i, T_s and A_i as $(T_s^*\|SK_i^*\|A_i^{***}) \leftarrow Oracle\mathcal{H}(L_i)$
if $(A_i^{***} == A_i^{**} == A_i^*)$ **then**
  Computes $r_i^{**} = A_i^* \oplus E_i$
  if $(r_i^{**} == r_i^*)$ **then**
    Computes $y_i^* = A_i^* \oplus Q_i$, $T_i^{1^{**}} = A_i^* \oplus G_i \oplus H(r_i^*)$ and $T_s^{**} = A_i^* \oplus K_i \oplus H(Y_i^*)$
    if $(T_i^{1^{**}} == T_i^{1^*})$ && $(T_s^{**} == T_s^*)$ **then**
      Computes $pwr_i^* = C_i \oplus A_i^*$
    else
      Return 0 (**Failure**)
    end if
  else
    Return 0 (**Failure**)
  end if
else
  Return 0 (**Failure**)
end if
repeat
  Chooses a password $pw_i^{[guess]}$
  Computes $B_i^{[guess]} = \widehat{B_i} \oplus H(ID_i\|pw_i^{[guess]})$ and $\theta_i^{[guess]} = \widehat{\theta_i} \oplus H(ID_i\|pw_i^{[guess]})$
  Calls $Oracle\mathcal{FE}_{Rep}$ on the input $B_i^{[guess]}$ and $\theta_i^{[guess]}$ to retrieve the information ϕ_i, as $(\phi_i^*) \leftarrow Oracle\mathcal{FE}_{Rep}(B_i^{[guess]}, \theta_i^{[guess]})$
  Computes $pwr_i^{[guess]} = H(pw_i^{[guess]}\|\phi_i^*)$
until $(pwr_i^{[guess]} == pwr_i^*)$
if $(pwr_i^{[guess]} == pwr_i^*)$ **then**
  Return 1 (**Success**)
else
  Return 0 (**Failure**)
end if

We define the success probability for $EXP1^{oracle}_{\mathcal{A},\,TFUAS}$ as $Succ1^{oracle}_{\mathcal{A},\,TFUAS} = Pr[EXP1^{oracle}_{\mathcal{A},\,TFUAS} = 1]$. Then the advantage of $EXP1^{oracle}_{\mathcal{A},\,TFUAS}$ is given by $Adv1^{oracle}_{\mathcal{A},\,TFUAS}(t, qH, qFE) = max_{\mathcal{A}}\{Succ1^{oracle}_{\mathcal{A},\,TFUAS}\}$, where the maximum is taken over all $\mathcal{A}$ with the execution time t, the number of queries qH made to the $Oracle\mathcal{H}$ oracle and the number of queries qFE made to the $Oracle\mathcal{FE}$. Our proposed scheme is said to be provably secure against the adversary $\mathcal{A}$ for deriving the password pw_i and biometric parameter B_i of a user U_i, if $Adv1^{oracle}_{\mathcal{A},\,TFUAS}(t, qH, qFE) \leq \xi$, for any small $\xi > 0$. According to algorithm $EXP1^{oracle}_{\mathcal{A},\,TFUAS}$ (see Algorithm 1), if the adversary $\mathcal{A}$ gets success to compute inversion of the cryptographic one-way hash function $H(\cdot)$ and also gets success to solve hardness of fuzzy extractor, he/she can successfully derive the password pw_i and biometric parameter B_i of the user U_i by using of the $Oracle\mathcal{H}$ random oracle and $Oracle\mathcal{FE}$ random oracle, and wins the game. But, according to Definition 1 and Definition 2, we know that $Adv_{\mathcal{A}}^{Oracle\mathcal{H}}(t) \leq \xi_1$, for any small $\xi_1 > 0$ and $Adv_{\mathcal{A}}^{Oracle\mathcal{FE}}(t) \leq \xi_2$, for any small $\xi_2 > 0$. Since, we get $Adv1^{oracle}_{\mathcal{A},\,TFUAS}(t, qH, qFE) \leq \xi$, for any small $\xi > 0$ because, the proposed scheme depends on both $Adv_{\mathcal{A}}^{Oracle\mathcal{H}}(t)$ and $Adv_{\mathcal{A}}^{Oracle\mathcal{FE}}(t)$. Thus, our proposed scheme is secure against the adversary $\mathcal{A}$ for deriving the password pw_i and biometric parameter B_i of the user U_i.

Theorem 2. *Under the assumption that a cryptographic one-way hash function $H(\cdot)$ acts as a random oracle, the proposed scheme is provably secure against an adversary $\mathcal{A}$ for deriving the secret key s of the remote server RS even if the adversary $\mathcal{A}$ gets parameters that are stored into the memory of U_i's smart card and traps the communication messages between a user U_i and the remote server RS.*

Proof 2. This research construct an adversary $\mathcal{A}$ who has the ability to derive the secret key s of the remote server RS. For this purpose, this research considers same assumptions as discussed in Theorem 1. The adversary $\mathcal{A}$ runs the experiment, $EXP2^{oracle}_{\mathcal{A},\,TFUAS}$ for our three factor user authentication scheme (TFUAS) to derive the secret key s of the remote server RS as given in the Algorithm 2.

Algorithm 2 $EXP2^{oracle}_{\mathcal{A},\,TFUAS}$

Input: $D_i, ID_i, G_i, F_i, E_i, Q_i, L_i, K_i$
Output: 0 or 1

Calls $Oracle\mathcal{H}$ on the input D_i to retrieve the information $A_i = H(s||ID_i)$ as $(A_i^*) \leftarrow Oracle\mathcal{H}(D_i)$
Calls $Oracle\mathcal{H}$ on the input F_i to retrieve the information A_i, r_i and T_i^1 as $(r_i^* \; || \; A_i^{**}||T_i^{1^*}) \leftarrow Oracle\mathcal{H}(F_i)$
Calls $Oracle\mathcal{H}$ on the input L_i to retrieve the information SK_i, T_s and A_i as $(T_s^*||SK_i^*||A_i^{***}) \leftarrow Oracle\mathcal{H}(L_i)$
if $(A_i^{***} == A_i^{**} == A_i^*)$ **then**
  Computes $r_i^{**} = A_i^* \oplus E_i$
  if $(r_i^{**} == r_i^*)$ **then**
    Computes $y_i^* = Q_i \oplus A_i^*$, $T_i^{1^{**}} = A_i^* \oplus G_i \oplus H(r_i^*)$ and $T_s^{**} = A_i^* \oplus K_i \oplus H(y_i^*)$
    if $(T_i^{1^{**}} == T_i^{1^*})$ && $(T_s^{**} == T_s^*)$ **then**
      Calls $Oracle\mathcal{H}$ on the input A_i^* to retrieve the information s and ID_i as $(s^*||ID_i^*) \leftarrow Oracle\mathcal{H}(A_i^*)$
      if $(ID_i == ID_i^*)$ **then**
        Accepts s^* as secret key of RS
        Return 1 (**Success**)
      else
        Return 0 (**Failure**)
      end if
    else
      Return 0 (**Failure**)
    end if
  else
    Return 0 (**Failure**)
  end if
else
  Return 0 (**Failure**)
end if

We define the success probability for $EXP2^{oracle}_{\mathcal{A},\,TFUAS}$ as $Succ2^{oracle}_{\mathcal{A},\,TFUAS} = Pr[EXP2^{oracle}_{\mathcal{A},\,TFUAS} = 1]$. Then the advantage of $EXP2^{oracle}_{\mathcal{A},\,TFUAS}$ is given by $Adv2^{oracle}_{\mathcal{A},\,TFUAS}(t, qH) = max_{\mathcal{A}}\{Succ2^{oracle}_{\mathcal{A},\,TFUAS}\}$, where the maximum is taken over all $\mathcal{A}$ with the execution time t, the number of queries qH made to the $Oracle\mathcal{H}$ oracle. The proposed scheme is said to be provably secure against the adversary $\mathcal{A}$ for deriving the secret key s of the remote server RS, if $Adv2^{oracle}_{\mathcal{A},\,TFUAS}(t, qH) \leq \xi$, for any small $\xi > 0$. According to algorithm $EXP2^{oracle}_{\mathcal{A},\,TFUAS}$ (see Algorithm 2), if the adversary $\mathcal{A}$ gets success to compute inversion of the cryptographic one-way hash function $H(\cdot)$, he/she can successfully derive the secret key s of the remote server RS by using of the $Oracle\mathcal{H}$ random oracle and wins the game. But, according to Definition 1, we know that $Adv_{\mathcal{A}}^{Oracle\mathcal{H}}(t) \leq \xi_1$, for any small $\xi_1 > 0$. Since, we get $Adv2^{oracle}_{\mathcal{A},\,TFUAS}(t, qH) \leq \xi$, for any small $\xi > 0$ because, the proposed scheme depends on $Adv_{\mathcal{A}}^{Oracle\mathcal{H}}(t)$. Thus, our proposed scheme is secure against the adversary $\mathcal{A}$ for deriving the secret key s of the remote server RS.

Theorem 3. *Under the assumption that a cryptographic one-way hash function $H(\cdot)$ acts as a random oracle, the proposed scheme is provably secure against an adversary $\mathcal{A}$ for deriving a shared secret session key SK_i between a user U_i and the remote server RS even if the adversary $\mathcal{A}$ gets parameters that are stored into the memory of U_i's smart card and traps the communication messages between U_i and the remote server RS.*

Proof 3. This research construct an adversary $\mathcal{A}$ who has the ability to derive the session key SK_i between a user U_i and the remote server RS. For this purpose, this research considers same assumptions as discussed in Theorem 1. The adversary $\mathcal{A}$ runs the experiment, $EXP3^{oracle}_{\mathcal{A},\,TFUAS}$ for our three factor user authentication scheme (TFUAS) to derive the session key SK_i between the user U_i and the remote server RS as given in the Algorithm 3.

Algorithm 3 $EXP3^{oracle}_{\mathcal{A},\,TFUAS}$

Input: $D_i, ID_i, G_i, F_i, E_i, Q_i, L_i, K_i$
Output: 0 or 1

Calls $Oracle\mathcal{H}$ on the input D_i to retrieve the information $A_i = H(s||ID_i)$ as $(A_i^*) \leftarrow Oracle\mathcal{H}(D_i)$
Calls $Oracle\mathcal{H}$ on the input F_i to retrieve the information A_i, r_i and T_i^1 as $(r_i^* \parallel A_i^{**}||T_i^{1^*}) \leftarrow Oracle\mathcal{H}(F_i)$
Calls $Oracle\mathcal{H}$ on the input L_i to retrieve the information SK_i, T_s and A_i as $(T_s^*||SK_i^*||A_i^{***}) \leftarrow Oracle\mathcal{H}(L_i)$
if $(A_i^{***} == A_i^{**} == A_i^*)$ **then**
  Computes $r_i^{**} = A_i^* \oplus E_i$
  if $(r_i^{**} == r_i^*)$ **then**
    Computes $T_i^{1^{**}} = A_i^* \oplus G_i \oplus H(r_i^*)$, $y_i^* = Q_i \oplus A_i^*$ and $T_s^{**} = A_i^* \oplus K_i \oplus y_i^*$
    if $(T_i^{1^{**}} == T_i^{1^*})$ && $(T_s^{**} == T_s^*)$ **then**
      Calls $Oracle\mathcal{H}$ on the input SK_i^* to retrieve the information y_i and r_i as $(y_i^{**}||r_i^{***}) \leftarrow Oracle\mathcal{H}(SK_i^*)$
      if $(r_i^{***} == r_i^*)$&& $(y_i^* == y_i^{**})$ **then**
        Accepts SK_i^* as the shared secret session key
        Return 1 (**Success**)
      else
        Return 0 (**Failure**)
      end if
    else
      Return 0 (**Failure**)
    end if
  else
    Return 0 (**Failure**)
  end if
else
  Return 0 (**Failure**)
end if

We define the success probability for $EXP3^{oracle}_{\mathcal{A},\,TFUAS}$ as $Succ3^{oracle}_{\mathcal{A},\,TFUAS} = Pr[EXP3^{oracle}_{\mathcal{A},\,TFUAS} = 1]$. Then the advantage of $EXP3^{oracle}_{\mathcal{A},\,TFUAS}$ is given by $Adv3^{oracle}_{\mathcal{A},\,TFUAS}(t, qH) = max_{\mathcal{A}}\{Succ3^{oracle}_{\mathcal{A},\,TFUAS}\}$, where the maximum is taken over all $\mathcal{A}$ with the execution time t, the number of queries qH made to the $Oracle\mathcal{H}$ oracle. The proposed scheme is said to be provably secure against the adversary $\mathcal{A}$ for deriving the session key SK_i between the user U_i and the remote server RS, if $Adv3^{oracle}_{\mathcal{A},\,TFUAS}(t, qH) \leq \xi$, for any small $\xi > 0$. According to algorithm $EXP3^{oracle}_{\mathcal{A},\,TFUAS}$ (see Algorithm 3), if the adversary $\mathcal{A}$ gets success to compute inversion of the cryptographic one-way hash function $H(\cdot)$, he/she can successfully derive the session key SK_i between the user U_i and the remote server RS by using of the $Oracle\mathcal{H}$ random oracle and wins the game. But, according to Definition 1, we know that $Adv^{Oracle\mathcal{H}}_{\mathcal{A}}(t) \leq \xi_1$, for any small $\xi_1 > 0$. Since, we get $Adv3^{oracle}_{\mathcal{A},\,TFUAS}(t, qH) \leq \xi$, for any small $\xi > 0$ because, the proposed scheme depends on $Adv^{Oracle\mathcal{H}}_{\mathcal{A}}(t)$. Thus, our proposed scheme is secure against the adversary $\mathcal{A}$ for deriving the session key SK_i between the user U_i and the remote server RS.

Theorem 1 demonstrated that the proposed scheme is secure against the *off-line password guessing attack*. Theorem 3 demonstrates that the proposed scheme is secure against the *session key recovery attack* because, without knowing random numbers $\{r_i, y_i\}$ then $\mathcal{A}$ cannot compute the session key SK_i. In the proposed scheme, the communicating messages depend on random numbers and the timestamp. Therefore, the communication messages are guaranteed to be different for every session. Thus, $\mathcal{A}$ cannot mount a *replay attack* on this proposed scheme. In this proposed scheme, $\mathcal{A}$ cannot mount a *forgery attack* without knowing secret password pw_i and biometric parameter B_i of a user U_i and the secret key s of the remote server RS. Theorems 1 and 2 show that the secret information of the remote server and the user are secure from $\mathcal{A}$. Thus, it is infeasible to mount a *forgery attack* on this proposed scheme.

A valid user say, $U_{\widehat{A}}$ as an adversary $\widehat{A}$ cannot login into the proposed authentication scheme as an another user say, U_i because, to login into the system, $\widehat{A}$ has to know the secret key s of the remote server RS. As, $\widehat{A}$ is a valid user, it knows their identity $ID_{\widehat{A}}$, password $pw_{\widehat{A}}$ and biometric information $\phi_{\widehat{A}}$. Therefore, $\widehat{A}$ can compute

DOI: 10.1051/itmconf/20171301020

Table 2. Security Functionality comparison of the proposed scheme with related schemes

Schemes	Ref. [17]	Ref. [18]	Ref. [19]	Ref. [22]	Ref. [23]	Ref. [24]	Ref. [26]	Our scheme
A1	√	√	×	-	√	√	×	×
A2	√	√	×	×	×	√	×	×
A3	√	√	√	-	-	×	×	×
A4	√	√	×	-	√	×	×	×
A5	√	√	√	-	-	√	√	×
A6	-	-	×	√	-	×	×	×
EPC	×	√	×	×	×	√	√	√
ELP	×	√	×	×	×	√	√	√

A1: Password guessing attack, A2: Insider attack, A3: Forgery attack, A4: Smart card stolen attack, A5: Replay attack, A6: Denial of Service (DoS) attack, ELP: Efficient login phase, EPC: Efficient password change phase, ×: no, √: yes

Table 3. Computation, communication and storage cost comparison of the proposed scheme with related schemes

Schemes	***Storage cost*** (in bits)	***Communication Cost*** (in bits) Login + Authentication	***Computation cost*** Login	Authentication
Ref. [17]	448	576	$2T_H$	$5T_H$
Ref. [18]	576	832	$3T_H$	$8T_H$
Ref. [19]	576	704	$3T_H$	$6T_H$
Ref. [22]	384	576	$4T_H+1T_{enc}$	$7T_H+1T_{dec}$
Ref. [23]	640	960	$3T_H$	$8T_H$
Ref. [24]	800	1120	$4T_H$	$10T_H+1T_{enc}+1T_{dec}$
Ref. [26]	384	1216	$4T_H$	$7T_H+1T_{enc}+1T_{dec}$
Proposed scheme	512	832	$4T_H$	$6T_H$

$H(s\|ID_{\widehat{A}})$ by computing $C_{\widehat{A}} \oplus H(pw_{\widehat{A}}\|\phi_{\widehat{A}})$, where $C_{\widehat{A}}$ is the smart card store parameter of $\widehat{A}$. From $H(s\|ID_{\widehat{A}})$, $\widehat{A}$ cannot extract s due to hardness of inversion of one-way hash function. Furthermore, Theorem 2 shows that s cannot be extracted or computed from known parameters of $\widehat{A}$. Therefore, $\widehat{A}$ unable to mount any attacks on the proposed scheme.

6 Comparison

In this section, the performances of the proposed scheme with the existing authentication schemes namely, Li and Hwang's scheme [17], Das's scheme [18], An's scheme [19], Tan's scheme [22], Yan et al.'s scheme [23], Mishra et al.'s schemw [24] and He et al.'s scheme [26] are compared. However, the compared schemes in [17–19] and [22–24] and [26] are not suitable for practical use because, the schemes cannot resist the possible attacks as shown in Table 2. In the introduction part of this paper, it has been described that is insecure against security attacks. Moreover, security analysis of the proposed scheme (see Section 5) shows that the proposed scheme can resist all the possible attacks. Thus, the proposed scheme is more secure than other schemes.

Table 3 shows the computational cost, storage cost and communication overhead comparison of schemes in [17–19], [22–24], and [26] with our proposed scheme. For this purpose, only login and authentications phases have been consider due to maximum use. T_H, T_{enc} and T_{dec} are the time required for hash operation, symmetric key encryption and decryption respectively. The proposed scheme takes time for 10 hashing operations in two phases which is the lower among related and compared schemes in [18, 22–24, 26]. It can be reasonably assumed that the length of ID_i and pw_i are 64 bits each. Large prime number q, cryptographic one-way hash function $H(\cdot)$ like SHA-1[1], symmetric key encryption/decryption like AES with key size 128 bits[2], random numbers, symmetric key encryption/decryption and timestamp returns 128 bits each. The communication cost of proposed scheme for login message is (128+128+128+64) = 448 bits and message generated in authentication phase is (128+128+128)

[1]http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf
[2]http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf

= 384 bits. Therefore, total communication cost is (448+384) = 832 bits which is lower than existing and compared schemes in [23, 24, 26]. The storage cost of our proposed scheme is (128 + 128 + 128 + 128) = 512 bits which is also lower than related schemes in [18, 19, 23, 24].

After resisting all possible attacks as shown in Section 5, the proposed scheme provides better trade-off among storage, computational and communication costs than other related existing schemes. Hence, it can be claimed that the proposed scheme is more efficient and secure than other related existing schemes and also it is applicable for practical applications.

- **Advantages of proposed scheme**

In the following, the advantages of the proposed scheme have been discussed.

Efficient login phase: If a user U_i enters faulty password and faulty identity by some means in login phase of the proposed scheme, the smart easily can detect the wrong inputs before going to generate a login request message. For this purpose, smart card computes $B'_i = \widehat{B_i} \oplus H(ID_i \| pw_i)$ and checks $des(B^*_i, B'_I) \leq \delta d$. If it does not hold, the smart card rejects U_i; otherwise, it computes $\theta'_i = \widehat{\theta_i} \oplus H(ID_i \| pw_i)$, $\phi'_i \leftarrow Rep(B^*_i, \theta'_i)$, $pwr'_i = H(pw_i \| \phi'_i)$, $A'_i = C_i \oplus pwr'_i$, $D'_i = H(A'_i)$ and checks computed D'_i and stored D_i are equal or not. If equality does not hold, the smart card rejects U_i; otherwise, accepts the password pw_i and identity ID_i as correct inputs. Therefore, extra communication overhead due to wrong inputs can be avoided in the proposed scheme.

Efficient password change phase: If a user U_i enters faulty password and faulty identity by some means in password phase of the proposed scheme, the smart easily can detect the wrong inputs before going to give permission to the user to submit their new password. For this purpose, smart card computes executes the same steps as mentioned above to check the submitted inputs. If provided inputs are correct, then only the smart card give permission to enter new password to U_i. Furthermore, to change password of a user, smart card does not need to communicate with the remote server. Therefore, communication overhead is also reduced in proposed scheme with efficient wrong input detection.

Mutual authentication: In the proposed scheme, the remote server RS computes and accepts a secret session key SK_i after verifying legitimacy of a user U_i through login request message and then, RS sends a reply message to the user U_i. The user U_i agrees upon the same secret session key SK_i with RS after verifying legitimacy of RS through reply message. Therefore, both way authentication has been done in the proposed scheme. Furthermore, the proposed scheme can resist all the possible attacks (see, Section 5). Hence, the proposed scheme achieves mutual authentication.

7 Conclusion

This paper have proposed an authentication scheme. After analyzing the proposed scheme it can be stated that the proposed scheme can overcome the all possible attacks and has better trade-off among computational, storage and communication costs. Therefore, the proposed scheme is suitable for real world online applications.

References

[1] T. Maitra, Cryptanalysis of A Secure Remote User Authentication Scheme Using Smart Cards, CoRR **abs/1502.04820** (2015)

[2] Y. Dodis, L. Reyzin, A. Smith, in *Advances in Cryptology - EUROCRYPT 2004* (Springer Berlin Heidelberg, 2004), Vol. 3027 of *Lecture Notes in Computer Science*, pp. 523–540

[3] L. Lamport, Password Authentication with Insecure Communication, Commun. ACM **24**, 770 (1981)

[4] A. Shimizu, T. Horioka, H. Inagaki, A Password Authentication Method for Contents Communications on the Internet, IEICE Tran. Communications **E81-B**, 1666 (1998)

[5] J. Xu, W.T. Zhu, D.G. Feng, An improved smart card based password authentication scheme with provable security, Computer Standards & Interfaces **31**, 723 (2009)

[6] X. Li, J. Niu, M.K. Khan, J. Liao, An enhanced smart card based remote user password authentication scheme, Journal of Network and Computer Applications **36**, 1365 (2013)

[7] T. Maitra, R. Amin, D. Giri, P.D. Srivastava, An Efficient and Robust User Authentication Scheme for Hierarchical Wireless Sensor Networks without Tamper-Proof Smart Card, I. J. Network Security **18**, 553 (2016)

[8] O. Mir, T. van der Weide, C.C. Lee, A Secure User Anonymity and Authentication Scheme Using AVISPA for Telecare Medical Information Systems, Journal of Medical Systems **39**, 89 (2015)

[9] D. Giri, T. Maitra, R. Amin, P. Srivastava, An Efficient and Robust RSA-Based Remote User Authentication for Telecare Medical Information Systems, Journal of Medical Systems **39**, 145 (2014)

[10] K.H. Yeh, A lightweight authentication scheme with user untraceability, Frontiers of Information Technology & Electronic Engineering **16**, 259 (2015)

[11] C. Guo, C.C. Chang, Chaotic maps-based password-authenticated key agreement using smart cards, Communications in Nonlinear Science and Numerical Simulation **18**, 1433 (2013)

[12] R. Amin, T. Maitra, S.P. Rana, An Improvement of Wang. et. al.'s Remote User Authentication Scheme against Smart Card Security Breach, International Journal of Computer Applications **75**, 37 (2013)

[13] C.C. Chang, C.Y. Sun, A Secure and Efficient Authentication Scheme for E-coupon Systems, Wireless Personal Communications **77**, 2981 (2014)

[14] R. Amin, T. Maitra, D. Giri, An Improved Efficient Remote User Authentication Scheme in Multi-server Environment using Smart Card, International Journal of Computer Applications **69**, 1 (2013)

[15] T. Maitra, M.S. Obaidat, S.H. Islam, D. Giri, R. Amin, Security analysis and design of an efficient ECC-based two-factor password authentication scheme, Security and Communication Networks **9**, 4166 (2016)

[16] T. Maitra, M.S. Obaidat, R. Amin, S.H. Islam, S.A. Chaudhry, D. Giri, A robust ElGamal-based password-authentication protocol using smart card for client-server communication, International Journal of Communication Systems **30** (2017)

[17] C.T. Li, M.S. Hwang, An efficient biometrics-based remote user authentication scheme using smart cards, Journal of Network and Computer Applications **33**, 1 (2010)

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

[19] Y. An, Security Analysis and Enhancements of an Effective Biometric-Based Remote User Authentication Scheme Using Smart Cards, Journal of Biomedicine and Biotechnology **2012, Article ID 519723**, 1 (2012)

[20] X. Li, J. Niu, M.K. Khan, J. Liao, X. Zhao, Robust three-factor remote user authentication scheme with key agreement for multimedia systems, Security and Communication Networks (2014)

[21] C.C. Lee, C.W. Hsu, A secure biometric-based remote user authentication with key agreement scheme using extended chaotic maps, Nonlinear Dynamics **71**, 201 (2013)

[22] Z. Tan, An efficient biometrics-based authentication scheme for telecare medicine information systems, Przeglad Elektrotechniczny pp. 200–204 (2013)

[23] X. Yan, W. Li, P. Li, J. Wang, X. Hao, P. Gong, A Secure Biometrics-based Authentication Scheme for Telecare Medicine Information Systems, Journal of Medical Systems **37**, 9972 (2013)

[24] D. Mishra, S. Mukhopadhyay, A. Chaturvedi, S. Kumari, M. Khan, Cryptanalysis and Improvement of Yan et al.'s Biometric-Based Authentication Scheme for Telecare Medicine Information Systems, Journal of Medical Systems **38**, 24 (2014)

[25] T. Maitra, D. Giri, An Efficient Biometric and Password-Based Remote User Authentication using Smart Card for Telecare Medical Information Systems in Multi-Server Environment, Journal of Medical Systems **38**, 142 (2014)

[26] D. He, N. Kumar, J.H. Lee, R. Sherratt, Enhanced three-factor security protocol for consumer USB mass storage devices, Consumer Electronics, IEEE Transactions on **60**, 30 (2014)

[27] D. Giri, R.S. Sherratt, T. Maitra, R. Amin, Efficient biometric and password based mutual authentication for consumer USB mass storage devices, IEEE Transactions on Consumer Electronics **61**, 491 (2015)

[28] D. Giri, R.S. Sherratt, T. Maitra, A novel and efficient session spanning biometric and password based three-factor authentication protocol for consumer USB Mass Storage Devices, IEEE Transactions on Consumer Electronics **62**, 283 (2016)

[29] H.F. Huang, H.W. Chang, P.K. Yu, Enhancement of Timestamp-based User Authentication Scheme with Smart Card, International Journal of Network Security **16**, 463 (2014)

[30] R. Amin, T. Maitra, D. Giri, P.D. Srivastava, Cryptanalysis and Improvement of an RSA Based Remote User Authentication Scheme Using Smart Card, Wireless Personal Communications (2017)

[31] D. Dolev, A.C. Yao, On the security of public key protocols, Information Theory, IEEE Transactions on **29**, 198 (1983)

[32] P. Kocher, J. Jaffe, B. Jun, *Differential Power Analysis*, in *Advances in Cryptology "CRYPTO'99* (1999), Vol. 1666 of *Lecture Notes in Computer Science*, pp. 388–397

[33] T.S. Messerges, E.A. Dabbish, R.H. Sloan, Examining Smart-Card Security Under the Threat of Power Analysis Attacks, IEEE Trans. Comput. **51**, 541 (2002)

[34] S.K. Sood, A.K. Sarje, K. Singh, A secure dynamic identity based authentication protocol for multi-server architecture, Journal of Network and Computer Applications **34**, 609 (2011)