J Supercomput (2018) 74:3504–3520
https://doi.org/10.1007/s11227-015-1601-y

An improved and robust biometrics-based three factor authentication scheme for multiserver environments

Shehzad Ashraf Chaudhry[1] · **Husnain Naqvi**[1] ·
Mohammad Sabzinejad Farash[2] · **Taeshik Shon**[3] ·
Muhammad Sher[1]

Published online: 30 December 2015

Abstract The rapid advancement in communication technologies enables remote users to acquire a number of online services. All such online services are provided remotely facilitating the users to freely move any where with out disruption of the services. In order to ensure seamless and secure services to the remote user such services espouse authentication protocols. A number of authentication protocols are readily available to achieve security and privacy in remote client server architecture. Most of these schemes are tailored for single server architecture. In such scenario, if a user wants to attain the services provided by more than one servers he has to register with each server. In recent times, multiserver authentication has got much attention, where a user can register once and then can acquire services provided by multiple servers. Very recently, Lu et al. proposed a biometric, smart card and password-based three

✉ Shehzad Ashraf Chaudhry
shahzad@iiu.edu.pk

Husnain Naqvi
husnain.naqvi@iiu.edu.pk

Mohammad Sabzinejad Farash
sabzinejad@khu.ac.ir

Taeshik Shon
taeshik.shon@gmail.com

Muhammad Sher
m.sher@iiu.edu.pk

[1] Department of Computer Science and Software Engineering, International Islamic University, Islamabad, Pakistan

[2] Kharazmi University, Tehran, Iran

[3] Division of Information and Computer Engineering, College of Information Technology, Ajou University, San 5, Woncheon-Dong, Yeongtong-Gu, Suwon 443-749, Korea

factor authentication scheme usable for multiserver environments. Furthermore, Lu et al. identified their scheme to resist known attacks. However, the analysis in this paper ascertains that Lu et al.'s scheme is vulnerable to impersonation attack. An adversary registered to the system just after knowing the public identity of a user can impersonate himself as the latter. Then we propose an improvement over Lu et al.'s scheme. Our improvement is more robust than the existing schemes. The security of proposed scheme is substantiated formally along with informal security discussion, while same is also validated using a popular automated tool ProVerif. The analysis confirms that proposed scheme achieves mutual authentication and is robust against known attacks. In addition, the proposed scheme does not incur any extra computation as compared with Lu et al.'s scheme.

Keywords Biometrics · Authentication · Multiserver · Impersonation attack · Smart card stolen attack · Anonymity · ProVerif

1 Introduction

During the recent times, wireless and mobile technologies have endured growth. Now a huge number of people are using mobile/wireless devices (e.g. smart phones, notebooks and PDAs) to access varying online services from anywhere and at anytime. These services include: remote medical treatment, video conferencing, VoIP, net-browsing and government services. However, the real constraint to such online services is the underlying public Internet infrastructure, which allows the attacker to intercept, eavesdrop and temper the messages transmitted between two honest entities. Therefore, it is most important to ensure the security of transmitted messages as well as the privacy of the participants. Password-based authentication scheme if employed properly can resolve such security issues. The first authentication scheme was proposed by Lamport [1]. However, their scheme was vulnerable to different attacks but it provided a basis for future research. The failure of Lamport's scheme was the usage of only a single factor (i.e. password) for authentication. Afterwards a number of two factor authentication using password as well as smart card were proposed [2–21]. Similarly to enhance the security, a number of three factor authentication schemes using password, smart card, and biometrics were also proposed [22–29]. All the mentioned biometric-based authentication schemes are usable in single server environments. In such cases, the user has to register to various servers, which in turn limits the scalability, because he has to remember a number of identities and password also he needs a separate smart card for each server. In 2010, Yoon and Yoo [30] proposed a biometric-based authentication scheme for multiserver environments. However in 2014, He and Wang [31] found a number of weaknesses including vulnerability to impersonation and smart card theft attack in Yoon et al.'s scheme. Then He et al. proposed an improved scheme. In 2014, Chaung and Chen [32] presented an authentication scheme based on biometrics for multiserver environments and claimed that their scheme is resistant to all known attacks, but soon Mishra et al. [33] realized that scheme proposed by Chaung and Chen is vulnerable to: (1) smart card theft attack; (2) server spoofing attack; and (3) denial of services attack. Mishra et al. then presented an authentication scheme to enhance the security. Very recently Lu et al. [34] identified that Mishra et al.'s scheme cannot resist user impersonation and server spoofing attacks. They also demonstrated that Mishra et al.'s scheme does not provide perfect forward secrecy. Lu et al. then proposed a new biometric-based three factor authentication scheme for multiserver environments. Lu et al. further claimed that their scheme is robust against numerous attacks. However, the analysis in this paper proves that Lu et al.'s scheme is defenseless against user impersonation attack. We show that a dishonest user of the system can impersonate as another user of the system by just knowing the public identity of the latter.

Rest of the paper is prescribed as follows: Sect. 2 accommodates notations used throughout the paper and basic concepts relating to one way hash functions, bio-hashing and the common adversarial model. Section 3 elaborates review of Lu et al.'s biometric-based authentication scheme for multiserver environments, followed by its cryptanalysis performed in Sect. 4. The proposed enhanced scheme is presented in Sect. 5. The formal and informal security analysis is performed in Sect. 6. The automated security validation of proposed scheme using ProVerif is performed in Sect. 7. The performance evaluation is done in Sect. 8. Finally, the conclusion is made in Sect. 9.

2 Preliminaries

This section elaborates some basics related to hash functions, bio-hashing, and adversarial model along with the notations used throughout the paper outlined in Table 1.

2.1 One way hash functions

A one way hash function $H : \{0, 1\}^* \rightarrow Z_q^*$ takes arbitrary length string S as input and outputs a fixed length code $C = H(S)$, the fixed length out put C is termed as hash value/hash code. A slight change in S results a significant change in C. Following are the properties to qualify a secure hash function:

- It is computationally easy to find $C = H(S)$, if S is given.
- It is computationally infeasible to compute S, if $C = H(S)$ is given.
- It is difficult to find two inputs S and T such that $H(S) = H(T)$. This property is known as collision-resistance property.

Table 1 Notation guide

Notations	Description	Notations	Description
$RC, \mathcal{S}_j$	Registration center, server	$\mathcal{U}_i, \mathcal{A}$	User, attacker
SID_j, ID_{ui}	Identities of $\mathcal{S}_j, \mathcal{U}_i$	PW_{ui}, BIO_{ui}	$\mathcal{U}_i$'s password and biometrics
x_{ui}	$\mathcal{U}_i$'s private key	Pub_{sj}, Pri_{sj}	Public and private key pair of $\mathcal{S}_j$
PSK_{rs}	Secret key between $\mathcal{S}_j$ and RC	SC_{ui}	$\mathcal{U}_i$'s smart card
$h(.), H(.)$	Hash and bio hash functions	$\|, \oplus$	Concatenation, Xor operators

Definition 1 (*Collision-resistant hash functions*) Let $H(.)$ be a collision resistant hash function. The probability for an attacker $\mathcal{A}$ to find a twain $(S \neq T)$ such that $H(S) = H(T)$ is defined as $Adv_{\mathcal{A}}^{HASH}(t_{e1}) = Prb[(S, T) \Leftarrow_r \mathcal{A} : (S \neq T) \ and \ H(S) = H(T)]$. Where $\mathcal{A}$ can randomly select a twain (S, T). The carried advantage of $\mathcal{A}$ over the randomly made selections within polynomial time t_{e1} is illustrated as $Adv_{\mathcal{A}}^{HASH}(t_{e1})$. The collision-resistant property for secure has functions implies that $Adv_{\mathcal{A}}^{HASH}(t_{e1}) \leq \epsilon$ for any sufficiently small $\epsilon > 0$.

2.2 Bio-hashing

The biometric refers to the measurable and distinct features used to mark and describe human. Biometric is often used for enabling the authentication to work provided the physical appearance of person. The biometric features (e.g. finger prints, facial expressions and retina etc.) may slightly vary at each imprint, which may cause a number of false rejection of legal users. Consequently impacting the usability of the system. To cope with false rejection, Jin et al. [35] presented a two factor authenticator using iterated inner product of human biometric features and tokenized random number. To accommodate this, user specific codes are generated. The user specific codes are termed as Bio-hash codes. During recent times, many bio-hashing schemes are proposed [36,37]. Bio-hashing is proved to be a convenient technique usable in small devices, such as smart card, smart phone.

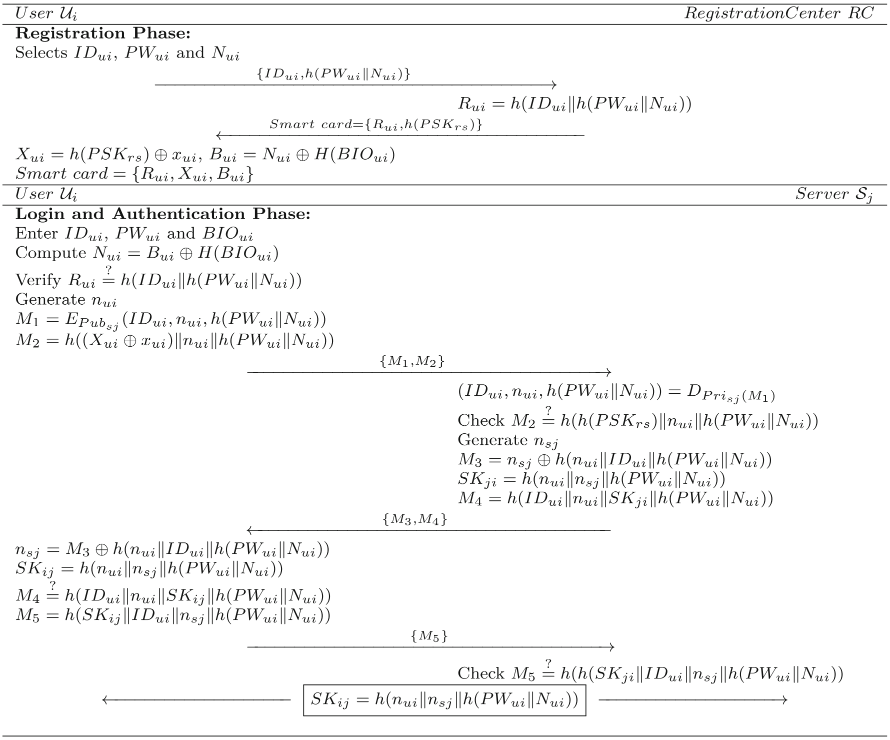
2.3 Adversarial model

In this paper, we consider the common adversarial model as mentioned in [38–40]. Where according to capabilities of the adversary $\mathcal{A}$, following assumptions are made:

1. $\mathcal{A}$ completely controls the public communication link. $\mathcal{A}$ is able to intercept, replay, modify, remove or can send a new fabricated message.
2. $\mathcal{A}$ can extract information contained in smart card by examining power analysis or leaked information [41,42].
3. $\mathcal{A}$ can be an outsider or can be a dishonest user of the system.
4. Identities of the registered users and servers are public and known to insiders.
5. The servers are assumed to be secure and $\mathcal{A}$ can not compromise any server of the system. (i.e. PSK_{rs} cannot be accessible to any adversary).

3 Review of Lu et al.'s scheme

In this section, we briefly review Lu et al.'s biometric-based authentication scheme. Lu et al. employed public key technique to achieve user anonymity and forward secrecy. Their scheme involves three participants: a user $\mathcal{U}_i$, a server $\mathcal{S}_j$ and the registration center RC. The scheme is illustrated in Fig. 1. We also elaborate Lu et al.'s scheme by the following three phases.



$User\ \mathcal{U}_i$
$RegistrationCenter\ RC$
Registration Phase:
Selects ID_{ui}, PW_{ui} and N_{ui}
$\{ID_{ui}, h(PW_{ui} \| N_{ui})\}$
$R_{ui} = h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$
$Smart\ card = \{R_{ui}, h(PSK_{rs})\}$
$X_{ui} = h(PSK_{rs}) \oplus x_{ui}$, $B_{ui} = N_{ui} \oplus H(BIO_{ui})$
$Smart\ card = \{R_{ui}, X_{ui}, B_{ui}\}$
$User\ \mathcal{U}_i$
$Server\ \mathcal{S}_j$
Login and Authentication Phase:
Enter ID_{ui}, PW_{ui} and BIO_{ui}
Compute $N_{ui} = B_{ui} \oplus H(BIO_{ui})$
Verify $R_{ui} \overset{?}{=} h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$
Generate n_{ui}
$M_1 = E_{Pub_{sj}}(ID_{ui}, n_{ui}, h(PW_{ui} \| N_{ui}))$
$M_2 = h((X_{ui} \oplus x_{ui}) \| n_{ui} \| h(PW_{ui} \| N_{ui}))$
$\{M_1, M_2\}$
$(ID_{ui}, n_{ui}, h(PW_{ui} \| N_{ui})) = D_{Pri_{sj}(M_1)}$
Check $M_2 \overset{?}{=} h(h(PSK_{rs}) \| n_{ui} \| h(PW_{ui} \| N_{ui}))$
Generate n_{sj}
$M_3 = n_{sj} \oplus h(n_{ui} \| ID_{ui} \| h(PW_{ui} \| N_{ui}))$
$SK_{ji} = h(n_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$
$M_4 = h(ID_{ui} \| n_{ui} \| SK_{ji} \| h(PW_{ui} \| N_{ui}))$
$\{M_3, M_4\}$
$n_{sj} = M_3 \oplus h(n_{ui} \| ID_{ui} \| h(PW_{ui} \| N_{ui}))$
$SK_{ij} = h(n_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$
$M_4 \overset{?}{=} h(ID_{ui} \| n_{ui} \| SK_{ij} \| h(PW_{ui} \| N_{ui}))$
$M_5 = h(SK_{ij} \| ID_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$
$\{M_5\}$
Check $M_5 \overset{?}{=} h(h(SK_{ji} \| ID_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$
$SK_{ij} = h(n_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$

Fig. 1 Lu et al.'s scheme

3.1 Registration phase

Registration involves following three steps:

Step Reg 1: $\mathcal{U}_i$ selects his identity ID_{ui}, password PW_{ui}, a random number N_{ui} along with his master private key x_{ui}. Then $\mathcal{U}_i$ scans his biometrics BIO_{ui}. Further, $\mathcal{U}_i$ sends $\{ID_{ui}, h(PW_{ui}, N_{ui})\}$ to RC on a private channel.

Step Reg 2: RC computes $R_{ui} = h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$ and personalizes the smart card SC_{ui} by $\{R_{ui}, h(PSK_{rs})\}$, where PSK_{rs} is the shared secret key between RC and $\mathcal{S}_j$. RC using private channel sends SC_{ui} to $\mathcal{U}_i$.

Step Reg 3: Upon receiving smart card, $\mathcal{U}_i$ computes $X_{ui} = h(PSK_{rs}) \oplus x_{ui}$, $B_{ui} = N_{ui} \oplus H(BIO_{ui})$. Then $\mathcal{U}_i$ deletes $h(PSK_{rs})$ from smart card (SC_{ui}) and stores X_{ui} and B_{ui} in the smart card (SC_{ui}). Finally, the smart card (SC_{ui}) contains $\{R_{ui}, X_{ui}, B_{ui}, h()\}$.

3.2 Login and authentication phase

During login and authentication phase, $\mathcal{U}_i$ inserts his SC_{ui} into card reader, imprints his biometrics (BIO_{ui}) and submits ID_{ui} and PW_{ui}. The steps performed by SC_{ui} and $\mathcal{S}_j$ are as follows:

Step LA1: SC_{ui} computes $N_{ui} = B_{ui} \oplus H(BIO_{ui})$ and $R'_{ui} = h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$.

Step LA2: SC_{ui} verifies $R_{ui} \overset{?}{=} h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$, if not true, SC_{ui} aborts the session.

Step LA3: SC_{ui} generates a random number n_{ui} and computes $M_1 = E_{Pub_{sj}}(ID_{ui}, n_{ui}, h(PW_{ui} \| N_{ui}))$ and $M_2 = h((X_{ui} \| x_{ui}) \| n_{ui} \| h(PW_{ui} \| N_{ui}))$.

Step LA4: Further, SC_{ui} sends login message $\{M_1, M_2\}$ to $\mathcal{S}_j$.

Step LA5: For the received login message, $\mathcal{S}_j$ using his private key decrypts M_1 to get $(ID_{ui}, n_{ui}, h(PW_{ui} \| N_{ui}))$.

Step LA6: $\mathcal{S}_j$ checks whether $M_2 \overset{?}{=} h(h(PSK_{rs}) \| n_{ui} \| h(PW_{ui} \| N_{ui}))$, if not true $\mathcal{S}_j$ aborts the session. Otherwise, $\mathcal{S}_j$ selects a random number n_{sj} and computes $M_3 = n_{sj} \oplus h(n_{ui} \| ID_{ui} \| h(PW_{ui} \| N_{ui}))$, the session key $SK_{ji} = h(n_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$ and $M_4 = h(ID_{ui} \| n_{ui} \| SK_{ji} \| h(PW_{ui} \| N_{ui}))$. Further, $\mathcal{S}_j$ sends $\{M_3, M_4\}$ to $\mathcal{U}_i$.

Step LA7: For the received login message, $\mathcal{U}_i$ computes $n_{sj} = M_3 \oplus h(n_{ui} \| ID_{ui} \| h(PW_{ui} \| N_{ui}))$ and session key $SK_{ij} = h(n_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$. $\mathcal{U}_i$ then checks $M_4 \overset{?}{=} h(ID_{ui} \| n_{ui} \| SK_{ij} \| h(PW_{ui} \| N_{ui}))$. If it holds, $\mathcal{U}_i$ ponders $\mathcal{S}_j$ as authenticated.

Step LA8: Finally, $\mathcal{U}_i$ computes and sends $M_5 = h(SK_{ij} \| ID_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$ to $\mathcal{S}_j$.

Step LA9: $\mathcal{S}_j$ checks $M_5 \overset{?}{=} h(h(SK_{ji} \| ID_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$ if it holds, $\mathcal{S}_j$ ponders $\mathcal{U}_i$ as authenticated.

The computed shared key between $\mathcal{U}_i$ and $\mathcal{S}_j$ is:

$$SK_{ij} = h(n_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui})) \quad (1)$$

3.3 Password change phase

$\mathcal{U}_i$ inserts his smart card (SC_{ui}) in specialized reader. $\mathcal{U}_i$ then inputs ID_{ui}, PW_{ui} and BIO_{ui}. SC_{ui} computes $N_{ui} = B_{ui} \oplus H(BIO_{ui})$ and checks $R_{ui} = h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$, if it holds SC_{ui} asks for new password. $\mathcal{U}_i$ inputs new password PW_{ui}^{new}. SC_{ui} computes $R_{ui}^{new} = h(ID_{ui} \| h(PW_{ui}^{new} \| N_{ui}))$. Finally SC_{ui} replaces R_{ui} by R_{ui}^{new}.

4 Cryptanalysis of Lu et al.'s scheme

This section elaborates the weakness of Lu et al.'s scheme against user impersonation attack. We show that a dishonest legal user $\mathcal{A}$ can easily masquerade himself as an other honest user $\mathcal{U}_i$ considering the common adversarial model as mentioned in Sect. 2.3. Let $\mathcal{A}$ be a legal user having smart card SC_a and wants to impersonate himself as another user $\mathcal{U}_i$. The attack is illustrated in Fig. 2. The description of the same is also detailed in following steps performed during interaction of $\mathcal{A}$ and $\mathcal{S}_j$:

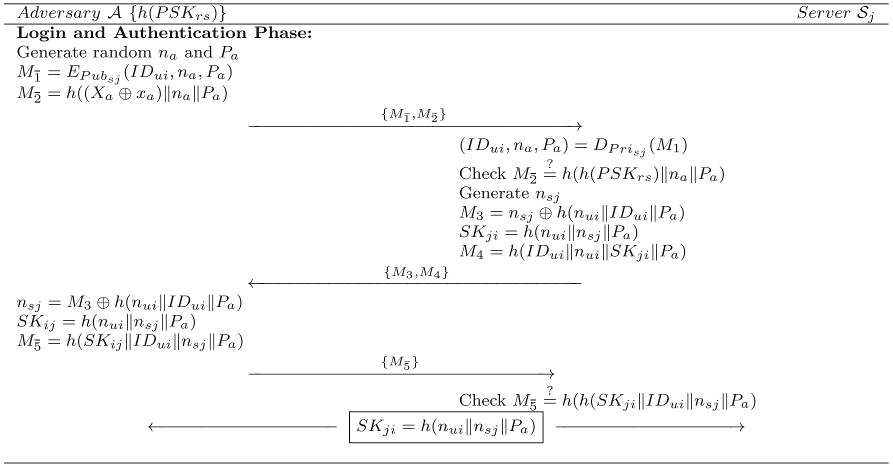

Adversary $\mathcal{A}$ $\{h(PSK_{rs})\}$
Server $\mathcal{S}_j$
Login and Authentication Phase:
Generate random n_a and P_a
$M_{\bar{1}} = E_{Pub_{sj}}(ID_{ui}, n_a, P_a)$
$M_{\bar{2}} = h((X_a \oplus x_a)\|n_a\|P_a)$
$\{M_{\bar{1}}, M_{\bar{2}}\}$
$(ID_{ui}, n_a, P_a) = D_{Pri_{sj}}(M_1)$
Check $M_{\bar{2}} \overset{?}{=} h(h(PSK_{rs})\|n_a\|P_a)$
Generate n_{sj}
$M_3 = n_{sj} \oplus h(n_{ui}\|ID_{ui}\|P_a)$
$SK_{ji} = h(n_{ui}\|n_{sj}\|P_a)$
$M_4 = h(ID_{ui}\|n_{ui}\|SK_{ji}\|P_a)$
$\{M_3, M_4\}$
$n_{sj} = M_3 \oplus h(n_{ui}\|ID_{ui}\|P_a)$
$SK_{ij} = h(n_{ui}\|n_{sj}\|P_a)$
$M_{\bar{5}} = h(SK_{ij}\|ID_{ui}\|n_{sj}\|P_a)$
$\{M_{\bar{5}}\}$
Check $M_{\bar{5}} \overset{?}{=} h(h(SK_{ji}\|ID_{ui}\|n_{sj}\|P_a)$
$SK_{ji} = h(n_{ui}\|n_{sj}\|P_a)$

Fig. 2 Impersonation attack on Lu et al.'s scheme

Step IA 1: $\mathcal{A}$ extracts the information stored in SC_a and computes:

$$h(PSK_{rs}) = X_a \oplus x_a \tag{2}$$

Step IA 2: $\mathcal{A}$ generates two random number n_a and P_a and computes:

$$M_{\bar{1}} = E_{Pub_{sj}}(ID_{ui}, n_a, P_a) \tag{3}$$

$$M_{\bar{2}} = h((X_a \oplus x_a)\|n_a\|P_a) \tag{4}$$

Step IA 3: $\mathcal{A}$ sends $M_{\bar{1}}$ and $M_{\bar{2}}$ as login message to $\mathcal{S}_j$.
Step IA 4: For the received login message, $\mathcal{S}_j$ decrypts $M_{\bar{1}}$ to obtain:

$$(ID_{ui}, n_a, P_a) = D_{Pri_{sj}}(M_{\bar{1}}) \tag{5}$$

Step IA 5: $\mathcal{S}_j$ further verifies $M_{\bar{2}} \overset{?}{=} h(h(PSK_{rs})\|n_a\|P_a)$ and finds it to be true.
Step IA 6: $\mathcal{S}_j$ further selects n_{sj} and computes:

$$M_3 = n_{sj} \oplus h(n_{ui}\|ID_{ui}\|P_a) \tag{6}$$

$$SK_{ji} = h(n_{ui}\|n_{sj}\|P_a) \tag{7}$$

$$M_4 = h(ID_{ui}\|n_{ui}\|SK_{ji}\|P_a) \tag{8}$$

Step IA 7: $\mathcal{S}_j$ sends M_3 and M_4 to $\mathcal{U}_i$ as response message.
Step IA 8: $\mathcal{A}$ intercepts the message and computes:

$$n_{sj} = M_3 \oplus h(n_{ui}\|ID_{ui}\|P_a) \tag{9}$$

$$SK_{ij} = h(n_{ui}\|n_{sj}\|P_a) \tag{10}$$

$$M_{\bar{5}} = h(SK_{ij}\|ID_{ui}\|n_{sj}\|P_a) \tag{11}$$

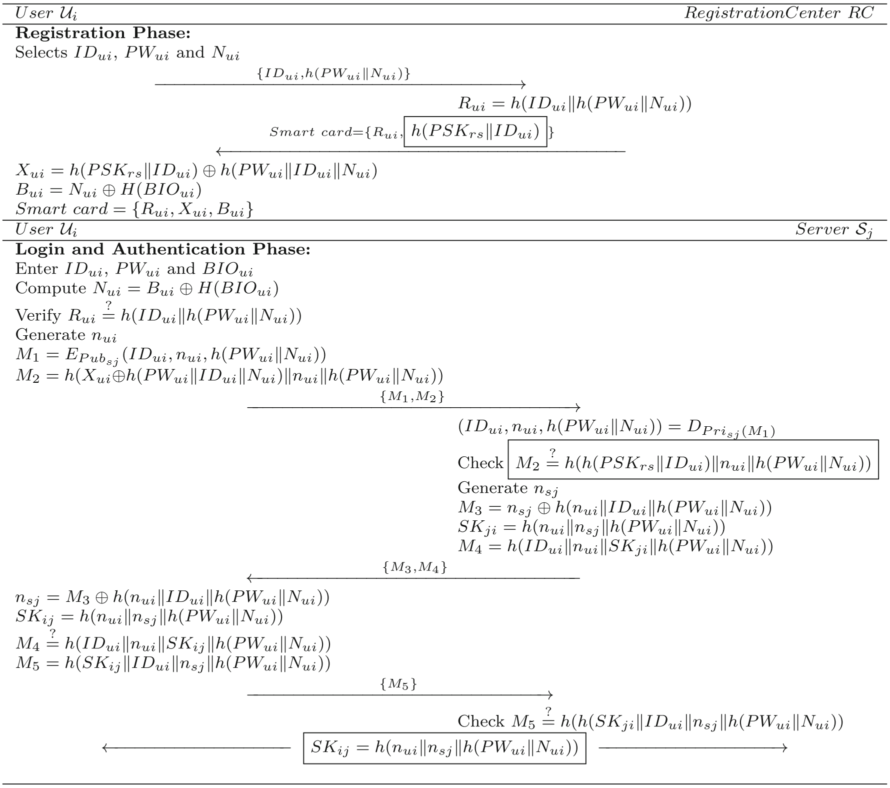

User $\mathcal{U}_i$
RegistrationCenter RC
Registration Phase:
Selects ID_{ui}, PW_{ui} and N_{ui}
$\{ID_{ui}, h(PW_{ui}\|N_{ui})\}$
$R_{ui} = h(ID_{ui}\|h(PW_{ui}\|N_{ui}))$
Smart card=$\{R_{ui}, \boxed{h(PSK_{rs}\|ID_{ui})}\}$
$X_{ui} = h(PSK_{rs}\|ID_{ui}) \oplus h(PW_{ui}\|ID_{ui}\|N_{ui})$
$B_{ui} = N_{ui} \oplus H(BIO_{ui})$
Smart card $= \{R_{ui}, X_{ui}, B_{ui}\}$
User $\mathcal{U}_i$
Server $\mathcal{S}_j$
Login and Authentication Phase:
Enter ID_{ui}, PW_{ui} and BIO_{ui}
Compute $N_{ui} = B_{ui} \oplus H(BIO_{ui})$
Verify $R_{ui} \overset{?}{=} h(ID_{ui}\|h(PW_{ui}\|N_{ui}))$
Generate n_{ui}
$M_1 = E_{Pub_{sj}}(ID_{ui}, n_{ui}, h(PW_{ui}\|N_{ui}))$
$M_2 = h(X_{ui} \oplus h(PW_{ui}\|ID_{ui}\|N_{ui})\|n_{ui}\|h(PW_{ui}\|N_{ui}))$
$\{M_1, M_2\}$
$(ID_{ui}, n_{ui}, h(PW_{ui}\|N_{ui})) = D_{Pri_{sj}(M_1)}$
Check $\boxed{M_2 \overset{?}{=} h(h(PSK_{rs}\|ID_{ui})\|n_{ui}\|h(PW_{ui}\|N_{ui}))}$
Generate n_{sj}
$M_3 = n_{sj} \oplus h(n_{ui}\|ID_{ui}\|h(PW_{ui}\|N_{ui}))$
$SK_{ji} = h(n_{ui}\|n_{sj}\|h(PW_{ui}\|N_{ui}))$
$M_4 = h(ID_{ui}\|n_{ui}\|SK_{ji}\|h(PW_{ui}\|N_{ui}))$
$\{M_3, M_4\}$
$n_{sj} = M_3 \oplus h(n_{ui}\|ID_{ui}\|h(PW_{ui}\|N_{ui}))$
$SK_{ij} = h(n_{ui}\|n_{sj}\|h(PW_{ui}\|N_{ui}))$
$M_4 \overset{?}{=} h(ID_{ui}\|n_{ui}\|SK_{ij}\|h(PW_{ui}\|N_{ui}))$
$M_5 = h(SK_{ij}\|ID_{ui}\|n_{sj}\|h(PW_{ui}\|N_{ui}))$
$\{M_5\}$
Check $M_5 \overset{?}{=} h(h(SK_{ji}\|ID_{ui}\|n_{sj}\|h(PW_{ui}\|N_{ui}))$
$\boxed{SK_{ij} = h(n_{ui}\|n_{sj}\|h(PW_{ui}\|N_{ui}))}$

Fig. 3 Proposed scheme

Step IA 9: $\mathcal{A}$ sends M_5 to $\mathcal{S}_j$.

Step IA 10: $\mathcal{S}_j$ checks $M_5 \overset{?}{=} h(h(SK_{ji}\|ID_{ui}\|n_{sj}\|P_a)$ and finds it to be true.

Hence, $\mathcal{A}$ successfully deceived $\mathcal{S}_j$ by impersonating himself as $\mathcal{U}_i$. The shared key between $\mathcal{A}$ and $\mathcal{S}_j$ is:

$$SK_{ji} = h(n_{ui}\|n_{sj}\|P_a) \tag{12}$$

5 Proposed scheme

This section elaborates the proposed improvement of Lu et al.'s scheme. The main problem of Lu et al.'s scheme is usage of secret parameter $h(PSK_{rs})$. This parameter is stored on smart card of each user. Therefore, an adversary after registering to the system can extract $h(PSK_{rs})$ from his own smart card. After obtaining $h(PSK_{rs})$, the adversary can easily impersonate himself any user of the system. In proposed scheme, we have alternated the generic secret $h(PSK_{rs})$ by a unique secret $h(PSK_{rs}\|ID_{ui})$. The proposed scheme as illustrated in Fig. 3 is described in following subsections.

5.1 Registration phase

Registration involves following three steps:

Step PR 1: $\mathcal{U}_i$ selects his identity ID_{ui}, password PW_{ui} and a random number N_{ui}. Then $\mathcal{U}_i$ scans his biometrics BIO_{ui}. Further $\mathcal{U}_i$ sends $\{ID_{ui}, h(PW_{ui}, N_{ui})\}$ to RC on a private channel.

Step PR 2: RC computes $R_{ui} = h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$ and personalizes the smart card SC_{ui} by $\{R_{ui}, h(PSK_{rs} \| ID_{ui})\}$, where PSK_{rs} is the shared secret key between RC and $\mathcal{S}_j$. RC using private channel sends SC_{ui} to $\mathcal{U}_i$.

Step PR 3: Upon receiving smart card, $\mathcal{U}_i$ computes $X_{ui} = h(PSK_{rs} \| ID_{ui}) \oplus h(PW_{ui} \| ID_{ui} \| N_{ui})$, $B_{ui} = N_{ui} \oplus H(BIO_{ui})$. Then $\mathcal{U}_i$ deletes $h(PSK_{rs} \| ID_{ui})$ from smart card (SC_{ui}) and stores X_{ui} and B_{ui} in the smart card (SC_{ui}). Finally, the smart card (SC_{ui}) contains $\{R_{ui}, X_{ui}, B_{ui}, h()\}$.

5.2 Login and authentication phase

During login and authentication phase, $\mathcal{U}_i$ inserts his SC_{ui} into card reader, imprints his biometrics (BIO_{ui}) and submits ID_{ui} and PW_{ui}. The steps performed by SC_{ui} and $\mathcal{S}_j$ are as follows:

Step LA1: SC_{ui} computes $N_{ui} = B_{ui} \oplus H(BIO_{ui})$ and $R'_{ui} = h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$.

Step LA2: SC_{ui} verifies $R_{ui} \overset{?}{=} h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$, if not true, SC_{ui} aborts the session.

Step LA3: SC_{ui} generates a random number n_{ui} and computes $M_1 = E_{Pub_{sj}}(ID_{ui}, n_{ui}, h(PW_{ui} \| N_{ui}))$ and $M_2 = h((X_{ui} \oplus h(PW_{ui} \| ID_{ui} \| N_{ui}) \| n_{ui} \| h(PW_{ui} \| N_{ui}))$.

Step LA4: Further, SC_{ui} sends login message $\{M_2, M_3\}$ to $\mathcal{S}_j$.

Step LA5: For the received login message, $\mathcal{S}_j$ using his private key decrypts M_1 to get $(ID_{ui}, n_{ui}, h(PW_{ui} \| N_{ui}))$.

Step LA6: $\mathcal{S}_j$ checks whether $M_2 \overset{?}{=} h(h(PSK_{rs} \| ID_{ui}) \| n_{ui} \| h(PW_{ui} \| N_{ui}))$, if not true $\mathcal{S}_j$ aborts the session. Otherwise, $\mathcal{S}_j$ selects a random number n_{sj} and computes $M_3 = n_{sj} \oplus h(n_{ui} \| ID_{ui} \| h(PW_{ui} \| N_{ui}))$, the session key $SK_{ji} = h(n_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$ and $M_4 = h(ID_{ui} \| n_{ui} \| SK_{ji} \| h(PW_{ui} \| N_{ui}))$. Further $\mathcal{S}_j$ sends $\{M_3, M_4\}$ to $\mathcal{U}_i$.

Step LA7: For the received login message, $\mathcal{U}_i$ computes $n_{sj} = M_3 \oplus h(n_{ui} \| ID_{ui} \| h(PW_{ui} \| N_{ui}))$ and session key $SK_{ij} = h(n_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$. $\mathcal{U}_i$ then checks $M_4 \overset{?}{=} h(ID_{ui} \| n_{ui} \| SK_{ij} \| h(PW_{ui} \| N_{ui}))$. If it holds, $\mathcal{U}_i$ ponders $\mathcal{S}_j$ as authenticated.

Step LA8: Finally, $\mathcal{U}_i$ computes and sends $M_5 = h(SK_{ij} \| ID_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$ to $\mathcal{S}_j$.

Step LA9: $\mathcal{S}_j$ checks $M_5 \overset{?}{=} h(h(SK_{ji} \| ID_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui}))$ if it holds, $\mathcal{S}_j$ ponders $\mathcal{U}_i$ as authenticated.

The computed shared key between $\mathcal{U}_i$ and $\mathcal{S}_j$ is:

$$SK_{ij} = h(n_{ui} \| n_{sj} \| h(PW_{ui} \| N_{ui})) \tag{13}$$

5.3 Password change phase

$\mathcal{U}_i$ inserts his smart card (SC_{ui}) in specialized reader. $\mathcal{U}_i$ then inputs ID_{ui}, PW_{ui} and BIO_{ui}. SC_{ui} computes $N_{ui} = B_{ui} \oplus H(BIO_{ui})$ and checks $R_{ui} = h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$, if it hold SC_{ui} asks for new password. $\mathcal{U}_i$ inputs new password PW_{ui}^{new}. SC_{ui} computes $R_{ui}^{new} = h(ID_{ui} \| h(PW_{ui}^{new} \| N_{ui}))$ and $X_{ui}^{new} = X_{ui} \oplus h(PW_{ui} \| ID_{ui} \| N_{ui}) \oplus h(PW_{ui}^{new} \| ID_{ui} \| N_{ui}^{new})$. Finally, SC_{ui} replaces R_{ui} and X_{ui} by R_{ui}^{new} and X_{ui}^{new}.

6 Security analysis

This section elaborates the security analysis of proposed scheme. Here, we prove that proposed scheme is robust and can with stand several attacks under the common adversarial model as mentioned in Sect. 2.3. The evidence is solicited in following subsections.

6.1 Formal security

To demonstrate that proposed scheme is provably secure, we adopted the same analysis as mentioned in [33,34]. For analysis, we define Reveal oracle as follows:

- *Reveal*: This oracle results an input string S from the hash code $T = h(S)$.

Theorem 1 *The proposed scheme is provably secure against an adversary $\mathcal{A}$ for stemming $\mathcal{U}_i$'s identity ID_{ui}, password PW_{ui}, the session key SK_{ij} and the shared key PSK_{rs} between Registration center RC and the server $\mathcal{S}_j$ considering one way hash function as a random oracle.*

Proof 1 For the proof purpose, we construct an attacker $\mathcal{A}$ with capabilities to derive a legal user $\mathcal{U}_i$'s ID_{ui}, PW_{ui}, the session key SK_{ij} between $\mathcal{U}_i$ and $\mathcal{S}_j$ and the shared key PSK_{rs} between $\mathcal{S}_j$ and RC. $\mathcal{A}$ simulates *Reveal* oracle to executes algorithmic experiment $EXPE1_{\mathcal{A},MSBTFAS}^{HASH}$ against our proposed multiserver biometric-based three factor authentication scheme (*MSBTFAS*). The success probability for $EXPE1_{\mathcal{A},MSBTFAS}^{HASH}$ is defined as $Succe_1 = |Pr[EXPE1_{\mathcal{A},MSBTFAS}^{HASH} = 1] - 1|$. The adversary advantage is defined as $Advt_{\mathcal{A},MSBTFAS}^{HASH}(t_{e1}, q_{rv}) = max_{\mathcal{A}}(Succe_1)$, where t_{e1} is the maximum execution time for polynomial bound adversary $\mathcal{A}$ and q_{rv} are the maximum number of *Reveal* queries. Referring to the experiment, $\mathcal{A}$ can derive ID_{ui}, PW_{ui}, SK_{ij} and PSK_{rs} if he can invert hash value (i.e. find S out of $h(S)$), which is infeasible as per Definition 1. Therefore, $Advt_{A}^{HASH}(t_{e1}) \leq \epsilon$ for sufficiently small value $\epsilon > 0$. The advantage $Advt_{\mathcal{A},,MSBTFAS}^{HASH}(t_{e1}, q_{rv})$ relies on $Advt_{A}^{HASH}(t_{e1})$. Hence,

$Advt^{HASH}_{\mathcal{A},MSBTFAS}(t_{e1}, q_{rv}) \leq \epsilon$. Therefore, the proposed scheme is secure against $\mathcal{A}$ for deriving ID_{ui}, PW_{ui}, SK_{ij} and PSK_{rs}. □

Algorithm 1 $EXPE1^{HASH}_{\mathcal{A},MSBTFAS}$

```
Eavesdrop the login request $(M_1, M_2)$, Where $M_1 = E_{Pub_{sj}}(ID_{ui}, n_{ui}, h(PW_{ui} \| N_{ui}))$ and $M_2 = h(h(PSK_{rs} \| ID_{ui}) \| n_{ui} \| h(PW_{ui} \| N_{ui}))$
Call reveal oracle on $M_2$ to get $(h(PSK_{rs} \| ID_{ui})' \| n'_{ui} \| h(PW_{ui} \| N_{ui})') \leftarrow Reveal(M_2)$
Eavesdrop the challenge message $(M_3, M_4)$, Where $M_3 = n_{sj} \oplus h(n_{ui} \| ID_{ui} \| h(PW_{ui} \| N_{ui}))$ and $M_4 = h(ID_{ui} \| n_{ui} \| SK_{ji} \| h(PW_{ui} \| N_{ui}))$
Call reveal oracle on $M_4$ to get $(ID'_{ui} \| n''_{ui} \| SK'_{ji} \| h(PW_{ui} \| N_{ui})'') \leftarrow Reveal(M_4)$
if $n'_{ui} = n''_{ui}$ then
    call Reveal on $h(PW_{ui} \| N_{ui})'$ to obtain $(PW'_{ui} \| N'_{ui}) \leftarrow Reveal(h(PW_{ui} \| N_{ui})')$
    call Reveal on $h(PW_{ui} \| N_{ui})''$ to obtain $(PW''_{ui} \| N''_{ui}) \leftarrow Reveal(h(PW_{ui} \| N_{ui})'')$
    call Reveal on $h(PSK_{rs} \| ID_{ui})'$ to obtain $(PSK'_{rs} \| ID''_{ui}) \leftarrow Reveal(h(PSK_{rs} \| ID_{ui})')$
    if $N'_{ui} = N''_{ui}$ then
        Accept $PW'_{ui}$, $ID'_{ui}$ and $SK'_{ji}$ as $\mathcal{U}_i$'s password, identity and the shared session key respectively.
        if $ID'_{ui} = ID''_{ui}$ then
            Accept $PSK'_{rs}$ as the shared key between $RC$ and $\mathcal{S}_j$
            return Success
        else
            return Fail
        end if
    else
        return Fail
    end if
else
    return Fail
end if
```

6.2 Further security discussion

In this subsection, we informally describes the security functionalities provided by proposed scheme.

6.2.1 Anonymity and privacy

In proposed scheme, $\mathcal{U}_i$'s identity (ID_{ui}) is not transmitted in plain text, rather it is encrypted by intended server $\mathcal{S}_j$'s public key. Hence, only $\mathcal{S}_j$ can know the real identity of the sender. Furthermore, the message M_1 contains session specific n_{ui}. Hence, no adversary can predict whether two sessions are initiated by same user.

6.2.2 Mutual authentication

$\mathcal{S}_j$ authenticates $\mathcal{U}_i$ by verifying $M_2 \overset{?}{=} h(h(PSK_{rs} \| ID_{ui}) \| n_{ui} \| h(PW_{ui} \| N_{ui}))$. To compute valid M_2 the adversary needs $h(PSK_{rs} \| ID_{ui})$ which can only be computed by involving both $\mathcal{U}_i$'s password and smart card. Similarly $\mathcal{S}_j$ is authenticated by verifying $M_4 \overset{?}{=} h(ID_{ui} \| n_{ui} \| SK_{ij} \| h(PW_{ui} \| N_{ui}))$. $\mathcal{U}_i$ sends M_1 and M_2 to server as authentication request. The session specific information n_{ui} and $\mathcal{U}_i$'s password, identity and secret number N_{ui} can be extracted by decrypting M_1. As M_1 is decrypted by using public key of $\mathcal{S}_j$. Hence to decrypt one needs private key of $\mathcal{S}_j$. Hence, only legal user can generate valid (M_1, M_2) pair. Similarly, only legal server can respond by M_4. Therefore, proposed scheme posses mutual authentication between $\mathcal{U}_i$ and $\mathcal{S}_j$.

6.2.3 *User and server impersonation attacks*

As described earlier in Sect. 6.2.2, only legal user can generate valid request (M_1, M_2) pair and only valid intended server can generate valid response M_4 and no adversary can generate either of the mentioned messages. Hence, proposed scheme resists user as well as server impersonation attacks.

6.2.4 *Smart card theft/stolen attack*

In proposed scheme, even if an adversary becomes able to acquire $\mathcal{U}_i$'s smart card. The adversary can further extracts $R_{ui} = h(ID_{ui} \| h(PW_{ui} \| N_{ui}))$, $X_{ui} = h(PSK_{rs} \| ID_{ui}) \oplus h(PW_{ui} \| ID_{ui} \| N_{ui})$ and $B_{ui} = N_{ui} \oplus H(BIO_{ui})$. Then to obtain $h(PSK_{rs} \| ID_{ui})$ and N_{ui} he needs $\mathcal{U}_i$'s password as well as biometrics. Hence, no forgery attack is possible with theft smart card.

6.2.5 *Replay attack*

An adversary after intercepting a previous message request (M_1, M_2) can replay it later on. But he will not be able to compute session specific (n_{ui}, n_{sj}) and password-related $h(PW_{ui} \| N_{ui})$. Furthermore, adversary will not be able to generate valid response message M_5. Hence no replay attack is feasible on proposed scheme.

6.2.6 *Perfect forward secrecy*

In proposed scheme, the session key contains session specific n_{ui} contributed by $\mathcal{U}_i$ and n_{sj} putted by $\mathcal{S}_j$. If some session key or long term private key of the server or user's password is exposed to the adversary it will have no effect on established session keys.

6.2.7 *Insider and stolen verifier attacks*

In proposed scheme, the user's password is not sent in plain text to the server. Furthermore, the server does not store any verifier table for user authentication. Hence, no insider or stolen verifier attack is possible on proposed scheme.

6.2.8 Password guessing attack

In proposed scheme, the information relating to $\mathcal{U}_i$'s password is protected by N_{ui} and one way hash function. Furthermore, there is no parameter to verify correctness of user's password. Hence password guessing attack is not feasible on proposed scheme.

6.2.9 No clock synchronization

The proposed scheme made use of session specific random number n_{ui} and n_{sj} for authentication. There is no time stamp involved in any message. Hence in proposed scheme, there is no need to perform clock synchronization.

7 Verification through ProVerif

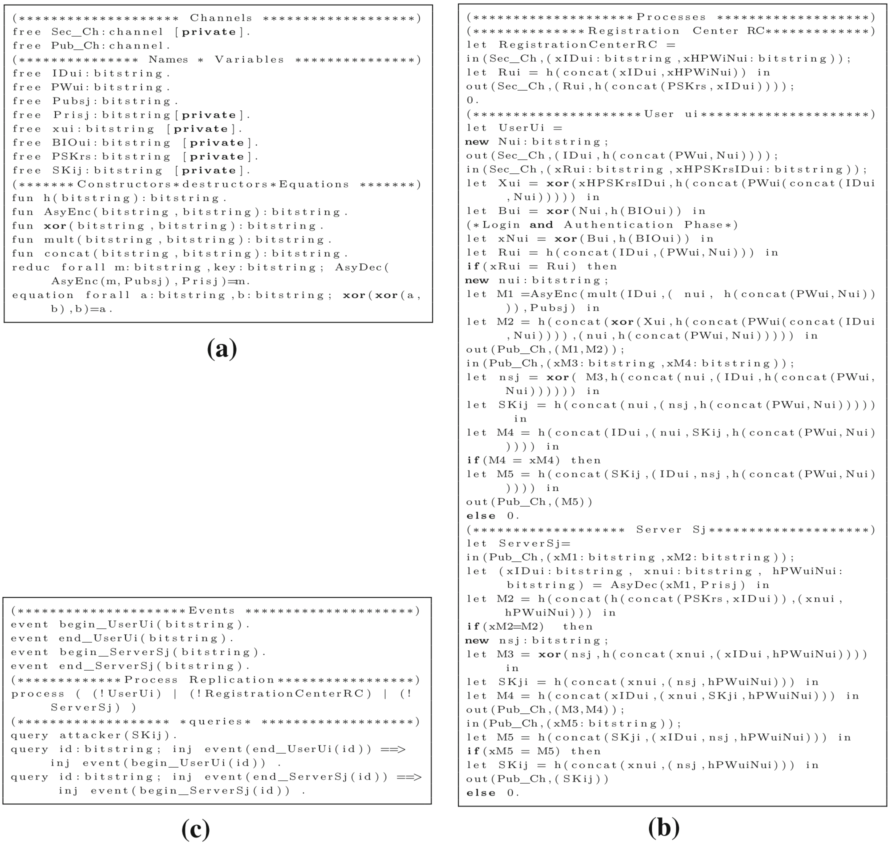
Cryptographic verification aims to examine the robustness of protocols against strong active attackers such as insiders who know some of the cryptographic parameters. ProVerif as designed is an automated verification tool to analyze security protocols against strong adversaries. Based on applied π calculus, ProVerif can verify numerous security aspects like: secrecy, reachability, and authentication [43–48]. To analyze the security of proposed scheme, we model the mentioned steps of Sect. 5, which are also illustrated in Fig. 3. The formal model of ProVerif can be described by following three parts: (1) declaration; (2) process; and (3) main. Declaration part is reserved for defining variables, constants and cryptographic primitives. As shown in Fig. 4a, we define two channels, the variables and constants. We also model the primitives used in proposed scheme as constructors, destructors and equations in declaration part. We define three processes for each registration center, server and user in processes part as shown in Fig. 4b. In main part, we simulate parallel execution of the three processes. To verify reachability property, we define start and end events of server and user. Finally, we applied three queries as shown in Fig. 4c. Following are the results:

1. RESULT inj-event(end_ServerSj(id)) ==> inj-event(begin_ServerSj(id)) is true.
2. RESULT inj-event(end_UserUi(id_3409)) ==>inj-event(begin_UserUi(id_3409)) is true.
3. RESULT not attacker(SKij[]) is true.

Results (1) and (2) confirms both the user and server processes initiated and terminated successfully which verifies that proposed scheme is correct and posses the reachability property. Result (3) confirms that the attacker is not able to compute the session key ($SKij$[]). Hence, proposed scheme is correct and fulfills reachability as well as secrecy and authentication properties.

8 Performance and security comparisons

This section elaborates the performance and security comparisons of proposed and related recent schemes [32–34]. We illustrate the security comparison of proposed scheme with related schemes in Table 2 under the mentioned adversarial model in



```
(******************** Channels *******************)
free Sec_Ch:channel [private].
free Pub_Ch:channel.
(*************** Names * Variables ***************)
free IDui:bitstring.
free PWui:bitstring.
free Pubsj:bitstring.
free Prisj:bitstring[private].
free xui:bitstring [private].
free BIOui:bitstring [private].
free PSKrs:bitstring [private].
free SKij:bitstring [private].
(*******Constructors*destructors*Equations *******)
fun h(bitstring):bitstring.
fun AsyEnc(bitstring,bitstring):bitstring.
fun xor(bitstring,bitstring):bitstring.
fun mult(bitstring,bitstring):bitstring.
fun concat(bitstring,bitstring):bitstring.
reduc forall m:bitstring,key:bitstring; AsyDec(
      AsyEnc(m,Pubsj),Prisj)=m.
equation forall a:bitstring,b:bitstring; xor(xor(a,
      b),b)=a.
```

(a)

```
(*********************Events *********************)
event begin_UserUi(bitstring).
event end_UserUi(bitstring).
event begin_ServerSj(bitstring).
event end_ServerSj(bitstring).
(*************Process Replication*****************)
process ( (!UserUi) | (!RegistrationCenterRC) | (!
      ServerSj) )
(******************* *queries* *******************)
query attacker(SKij).
query id:bitstring; inj event(end_UserUi(id)) ==>
      inj event(begin_UserUi(id)) .
query id:bitstring; inj event(end_ServerSj(id)) ==>
       inj event(begin_ServerSj(id)) .
```

(c)

```
(********************Processes *******************)
(**************Registration Center RC*************)
let RegistrationCenterRC =
in(Sec_Ch,(xIDui:bitstring,xHPWiNui:bitstring));
let Rui = h(concat(xIDui,xHPWiNui)) in
out(Sec_Ch,(Rui,h(concat(PSKrs,xIDui))));
0.
(*********************User ui*********************)
let UserUi =
new Nui:bitstring;
out(Sec_Ch,(IDui,h(concat(PWui,Nui))));
in(Sec_Ch,(xRui:bitstring,xHPSKrsIDui:bitstring));
let Xui = xor(xHPSKrsIDui,h(concat(PWui(concat(IDui
      ,Nui))))) in
let Bui = xor(Nui,h(BIOui)) in
(*Login and Authentication Phase*)
let xNui = xor(Bui,h(BIOui)) in
let Rui = h(concat(IDui,(PWui,Nui))) in
if(xRui = Rui) then
new nui:bitstring;
let M1 =AsyEnc(mult(IDui,( nui, h(concat(PWui,Nui))
      )),Pubsj) in
let M2 = h(concat(xor(Xui,h(concat(PWui(concat(IDui
      ,Nui)))),(nui,h(concat(PWui,Nui))))) in
out(Pub_Ch,(M1,M2));
in(Pub_Ch,(xM3:bitstring,xM4:bitstring));
let nsj = xor( M3,h(concat(nui,(IDui,h(concat(PWui,
      Nui)))))) in
let SKij = h(concat(nui,(nsj,h(concat(PWui,Nui)))))
       in
let M4 = h(concat(IDui,(nui,SKij,h(concat(PWui,Nui)
      )))) in
if(M4 = xM4) then
let M5 = h(concat(SKij,(IDui,nsj,h(concat(PWui,Nui)
      )))) in
out(Pub_Ch,(M5))
else 0.
(******************* Server Sj*********************)
let ServerSj=
in(Pub_Ch,(xM1:bitstring,xM2:bitstring));
let (xIDui:bitstring, xnui:bitstring, hPWuiNui:
      bitstring) = AsyDec(xM1,Prisj) in
let M2 = h(concat(h(concat(PSKrs,xIDui)),(xnui,
      hPWuiNui))) in
if(xM2=M2)  then
new nsj:bitstring;
let M3 = xor(nsj,h(concat(xnui,(xIDui,hPWuiNui))))
      in
let SKji = h(concat(xnui,(nsj,hPWuiNui))) in
let M4 = h(concat(xIDui,(xnui,SKji,hPWuiNui))) in
out(Pub_Ch,(M3,M4));
in(Pub_Ch,(xM5:bitstring));
let M5 = h(concat(SKji,(xIDui,nsj,hPWuiNui))) in
if(xM5 = M5) then
let SKij = h(concat(xnui,(nsj,hPWuiNui))) in
out(Pub_Ch,(SKij))
else 0.
```

(b)

Fig. 4 ProVerif validation. **a** Declarations, **b** processes, **c** main

Table 2 Comparison of security parameters

Scheme	Proposed	[34]	[33]	[32]
Anonymity and privacy	Yes	No	Yes	Yes
Mutual authentication and key agreement	Yes	Yes	Yes	Yes
Resists impersonation attack	Yes	No	No	No
Resists smart card theft attack	Yes	Yes	Yes	No
Resists replay attack	Yes	Yes	Yes	Yes
Provides forward secrecy	Yes	Yes	No	Yes
Resists insider and stolen verifier attacks	Yes	Yes	Yes	Yes
Resists password guessing attack	Yes	Yes	Yes	Yes
Provides no clock synchronization	Yes	Yes	Yes	Yes

Sect. 2.3. The vulnerability to impersonation attack of Lu et al.'s scheme was due to the fact, they used a generic value $h(PSK_{rs})$ to authenticate any user. Hence a registered user can extract this value from his smart card and then can impersonate

Table 3 Computational cost comparison

Participant	Scheme			
	Proposed	[34]	[33]	[32]
User	$1t_{aen} + 7t_h$	$1t_{aen} + 7t_h$	$10t_h$	$8t_h$
Server	$1t_{aen} + 5t_h$	$1t_{aen} + 5t_h$	$7t_h$	$8t_h$
Total	$2t_{aen} + 12t_h$	$2t_{aen} + 12t_h$	$17t_h$	$16t_h$

himself as any user of the system provided he obtains the public identity of the user. We alternated the use of generic value $h(PSK_{rs})$ by user specific value $h(PSK_{rs} \| ID_{ui})$, which prevents impersonation and other attacks. Refer to Table 2, Proposed scheme is robust against all attacks, while all other schemes are vulnerable to impersonation attacks. In addition, Mishra et al. scheme does not provide forward secrecy, while Chaung et al.'s scheme can not resist smart card lost/theft attack. Following are the notations used for performance comparison:

- t_h: time to compute hash code
- t_{pml}: time to perform point multiplication
- t_{aen}: time to perform asymmetric operations

Table 3 illustrates the computational cost comparisons. It can be seen that only Mishra et al. and Chaung et al.'s schemes are having least cost because both schemes are based on hash function. Proposed scheme is having same computation cost as of Lu et al.'s scheme. Hence proposed scheme not only improved the security but is also having same computation cost as of Lu et al.'s scheme.

9 Conclusion

In this paper, we analyzed Lu et al.'s biometric-based authentication scheme for multiserver environments. We proved that Lu et al.'s scheme cannot withstand user impersonation attack. Then we proposed an improved and robust biometric-based authentication scheme to enhance security. We also proved the security of proposed scheme using popular automated tool ProVerif. The improved scheme did not change the computation and communication costs of original scheme while ensuring security and privacy of the remote user.

References

1. Lamport L (1981) Password authentication with insecure communication. Commun ACM 24(11):770–772
2. He D (2012) An efficient remote user authentication and key agreement protocol for mobile client–server environment from pairings. Ad Hoc Netw 10(6):1009–1016
3. Farash MS, Attari MA (2014) A secure and efficient identity-based authenticated key exchange protocol for mobile client-server networks. J Supercomput 69(1):395–411
4. Farash MS, Attari MA (2014) An anonymous and untraceable password-based authentication scheme for session initiation protocol using smart cards. Int J Commun Syst. doi:10.1002/dac.2848
5. Farash MS, Attari MA (2014) Cryptanalysis and improvement of a chaotic map-based key agreement protocol using Chebyshev sequence membership testing. Nonlinear Dyn 76(2):1203–1213

6. Irshad A, Sher M, Faisal MS, Ghani A, Ul Hassan M, Ch SA (2013) A secure authentication scheme for session initiation protocol by using ECC on the basis of the Tang and Liu scheme. Secur Commun Netw 7(8):1210–1218. doi:10.1002/sec.834
7. Irshad A, Sher M, Rehman E, Ch SA, Hassan MU, Ghani A (2013) A single round-trip sip authentication scheme for voice over internet protocol using smart card. Multimed Tools Appl 74(11):3967–3984. doi:10.1007/s11042-013-1807-z
8. Islam S, Khan M (2014) Cryptanalysis and improvement of authentication and key agreement protocols for telecare medicine information systems. J Med Syst. doi:10.1007/s10916-014-0135-9
9. Chaudhry S, Naqvi H, Shon T, Sher M, Farash M (2015) Cryptanalysis and improvement of an improved two factor authentication protocol for telecare medical information systems. J Med Syst. doi:10.1007/s10916-015-0244-0
10. Jiang Q, Ma J, Tian Y (2014) Cryptanalysis of smart-card-based password authenticated key agreement protocol for session initiationprotocol of zhang et al. Int J Commun Syst. doi:10.1002/dac.2767
11. Zhang L, Tang S, Cai Z (2014) Robust and efficient password authenticated key agreement with user anonymity for session initiation protocol-based communications. IET Commun 8(1):83–91
12. He D, Kumar N, Chen J, Lee C-C, Chilamkurti N, Yeo S-S (2015) Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks. Multimedia Syst 21(1):49–60. doi:10.1007/s00530-013-0346-9
13. He D, Kumar N, Chilamkurti N (2015) A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks. Inf Sci 321:263–274. doi:10.1016/j.ins.2015.02.010
14. He D, Zeadally S (2015) Authentication protocol for an ambient assisted living system. Commun Mag IEEE 53(1):71–77
15. Farash MS, Chaudhry SA, Heydari M, Sajad Sadough SM, Kumari S, Khan MK (2015) A lightweight anonymous authentication scheme for consumer roaming in ubiquitous networks with provable security. Int J Commun Syst. doi:10.1002/dac.3019
16. Mehmood Z, Uddin N, Ch SA, Nasar W, Ghani A (2012) An efficient key agreement with rekeying for secured body sensor networks. In: 2012 second international conference on digital information processing and communications (ICDIPC). IEEE, pp 164–167
17. Chaudhry SA, Farash MS, Naqvi H, Islam SH, Shon T, Sher M (2015) A robust and efficient privacy aware handover authentication scheme for wireless networks. Wirel Pers Commun. doi:10.1007/s11277-015-3139-y
18. Heydari M, Sadough S, Farash M, Chaudhry S, Mahmood K (2015) An efficient password-based authenticated key exchange protocol with provable security for mobile client–client networks. Wirel Pers Commun. doi:10.1007/s11277-015-3123-6
19. Guo P, Wang J, Geng XH, Kim CS, Kim J-U (2014) A variable threshold-value authentication architecture for wireless mesh networks. J Internet Technol 15(6):929–935. doi:10.6138/JIT.2014.15.6.05
20. Amin R, Biswas G (2015) A novel user authentication and key agreement protocol for accessing multi-medical server usable in TMIS. J Med Syst 39(3):1–17
21. Amin R, Islam SH, Biswas G, Khan MK, Kumar N (2015) An efficient and practical smart card based anonymity preserving user authentication scheme for TMIS using elliptic curve cryptography. J Med Syst 39(11):1–18
22. Lu Y, Li L, Peng H, Yang Y (2015) An enhanced biometric-based authentication scheme for telecare medicine information systems using elliptic curve cryptosystem. J Med Syst 39(3):1–8
23. Awasthi AK, Srivastava K (2013) A biometric authentication scheme for telecare medicine information systems with nonce. J Med Syst 37(5):1–4
24. Li X, Niu J, Khan MK, Liao J, Zhao X (2014) Robust three-factor remote user authentication scheme with key agreement for multimedia systems. Secur Commun Netw. doi:10.1002/sec.961
25. Zhang M, Zhang J, Zhang Y (2015) Remote three-factor authentication scheme based on fuzzy extractors. Secur Commun Netw 8(4):682–693. doi:10.1002/sec.1016
26. Mishra D, Kumari S, Khan MK, Mukhopadhyay S (2015) An anonymous biometric-based remote user-authenticated key agreement scheme for multimedia systems. Int J Commun Syst. doi:10.1002/dac.2946
27. Das AK (2015) A secure and effective biometric-based user authentication scheme for wireless sensor networks using smart card and fuzzy extractor. Int J Commun Syst. doi:10.1002/dac.2933

28. Li X, Khan M, Kumari S, Liao J, Liang W (2014) Cryptanalysis of a robust smart card authentication scheme for multi-server architecture. In: 2014 international symposium on biometrics and security technologies (ISBAST), pp 120–123. doi:10.1109/ISBAST.2014.7013106
29. He D, Kumar N, Lee J-H, Sherratt R (2014) Enhanced three-factor security protocol for consumer USB mass storage devices. IEEE Trans Consum Electron 60(1):30–37. doi:10.1109/TCE.2014.6780922
30. Yoon E-J, Yoo K-Y (2013) Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem. J Supercomput 63(1):235–255
31. He D, Wang D (2014) Robust biometrics-based authentication scheme for multiserver environment. IEEE Syst J 99:1–9. doi:10.1109/JSYST.2014.2301517
32. Chuang M-C, Chen MC (2014) An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics. Expert Syst Appl 41(4):1411–1418
33. Mishra D, Das AK, Mukhopadhyay S (2014) A secure user anonymity-preserving biometric-based multi-server authenticated key agreement scheme using smart cards. Expert Syst Appl 41(18):8129–8143
34. Lu Y, Li L, Peng H, Yang Y (2015) A biometrics and smart cards-based authentication scheme for multi-server environments. Secur Commun Netw. doi:10.1002/sec.1246
35. Jin ATB, Ling DNC, Goh A (2004) Biohashing: two factor authentication featuring fingerprint data and tokenised random number. Pattern Recognit 37(11):2245–2255
36. Lumini A, Nanni L (2007) An improved biohashing for human authentication. Pattern Recognit 40(3):1057–1065
37. Belguechi R, Rosenberger C, Ait-Aoudia S (2010) Biohashing for securing minutiae template. In: 2010 20th international conference on pattern recognition (ICPR). IEEE, pp 1168–1171
38. Eisenbarth T, Kasper T, Moradi A, Paar C, Salmasizadeh M, Shalmani M (2008) On the power of power analysis in the real world: a complete break of the KeeLoq code hopping scheme. In: Wagner D (ed) Advances in cryptology, CRYPTO 2008, vol 5157 of lecture notes in computer science. Springer, Berlin, pp 203–220. doi:10.1007/978-3-540-85174-5_12
39. Dolev D, Yao AC (1983) On the security of public key protocols. IEEE Trans Inf Theory 29(2):198–208. doi:10.1109/TIT.1983.1056650
40. Cao X, Zhong S (2006) Breaking a remote user authentication scheme for multi-server architecture. IEEE Commun Lett 10(8):580–581. doi:10.1109/LCOMM.2006.1665116
41. Kocher P, Jaffe J, Jun B (1999) Differential power analysis. In: Advances in cryptology CRYPTO 99. Springer, pp 388–397
42. Messerges TS, Dabbish EA, Sloan RH (2002) Examining smart-card security under the threat of power analysis attacks. IEEE Trans Comput 51(5):541–552
43. Xie Q, Dong N, Wong DS, Hu B (2014) Cryptanalysis and security enhancement of a robust two-factor authentication and key agreement protocol. Int J Commun Syst. doi:10.1002/dac.2858
44. Chaudhry SA, Mahmood K, Naqvi H, Khan MK (2015) An improved and secure biometric authentication scheme for telecare medicine information systems based on elliptic curve cryptography. J Med Syst. doi:10.1007/s10916-015-0335-y
45. Kumari S, Chaudhry SA, Wu F, Li X, Farash MS, Khan MK (2015) An improved smart card based authentication scheme for session initiation protocol. Peer-to-Peer Netw Appl. doi:10.1007/s12083-015-0409-0
46. Chaudhry SA, Naqvi H, Sher M, Farash MS, Hassan Mu (2015) An improved and provably secure privacy preserving authentication protocol for sip. Peer-to-Peer Netw Appl. doi:10.1002/ppna.1299
47. Chaudhry SA, Farash MS, Naqvi H, Kumari S, Khan MK (2015) An enhanced privacy preserving remote user authentication scheme with provable security. Netw Secur Commun. doi:10.1002/sec.1299
48. Chaudhry SA, Farash M, Naqvi H, Sher M (2015) A secure and efficient authenticated encryption for electronic payment systems using elliptic curve cryptography. Electron Commer Res. doi:10.1007/s10660-015-9192-5