Hindawi Publishing Corporation
BioMed Research International
Volume 2013, Article ID 491289, 9 pages
http://dx.doi.org/10.1155/2013/491289

Hindawi

*Research Article*

# An Improved Biometrics-Based Remote User Authentication Scheme with User Anonymity

**Muhammad Khurram Khan[1] and Saru Kumari[2]**

[1] *King Saud University, P.O. Box 92144, Riyadh 11653, Saudi Arabia*
[2] *Department of Mathematics, Agra College, Agra, Dr. B. R. A. University, Agra, Uttar Pradesh 282002, India*

Correspondence should be addressed to Muhammad Khurram Khan; mkhurram@ksu.edu.sa

Received 4 August 2013; Accepted 2 September 2013

Academic Editor: Sabah Mohammed



The authors review the biometrics-based user authentication scheme proposed by An in 2012. The authors show that there exist loopholes in the scheme which are detrimental for its security. Therefore the authors propose an improved scheme eradicating the flaws of An's scheme. Then a detailed security analysis of the proposed scheme is presented followed by its efficiency comparison. The proposed scheme not only withstands security problems found in An's scheme but also provides some extra features with mere addition of only two hash operations. The proposed scheme allows user to freely change his password and also provides user anonymity with untraceability.

## 1. Introduction

In the last two decades, digital authentication has originated as a preferred method to authenticate remote users over insecure networks. After the first proposal of user authentication scheme by Lamport [1], considerable amount of research has been conducted in this field of which schemes [1–25] are few examples. In due course of time user authentication schemes underwent many changes. Initial schemes were based only on password [1–4], then schemes were based on smart card and password [5–13], and reliability of biometrics authentication over traditional password-based authentication gave rise to biometrics-based user authentication schemes [14–20].

In 2010, Li and Hwang [19] proposed a biometrics-based user authentication scheme. In 2011, Das [26] examined Li-Hwang's scheme and observed problems in login and authentication phase, in password change phase, and in biometrics verification mechanism of the scheme. Das depicted that user's smart card does not validate the inputted password during login phase which leads to useless computations in login and authentication phase. Owing to the same reason, Das further showed that the scheme suffers from incorrect password updating problem. Thus, Das proposed an improvement [26] of Li-Hwang's scheme and claimed their scheme to be free from problems observed in Li-Hwang's scheme. According to Das, their scheme [26] also provides mutual authentication. In 2012, An [27] pointed out that Das's scheme [26] deviates from the author's claim since an adversary can mount impersonation attacks and password guessing attack once he gets a chance to extract values from the smart card of the legal user. Thereby An [27] proposed an enhanced scheme to eradicate the flaws of Das's scheme.

In this paper, we review An's biometrics-based user authentication scheme. We show that An's scheme is vulnerable to the security problems to which Das's scheme is susceptible like online and offline password guessing attacks, user and server impersonation attacks, lack of mutual authentication, and lack of user anonymity. Besides, An's scheme lacks password change facility which is an important part of password-based user authentication schemes. We remove drawbacks from An's scheme by means of proposing an improved user authentication scheme. In addition, to resist various security threats, the proposed scheme incorporates features of password changing and user anonymity. The rest of this paper is arranged as follows. In Section 2, we review An's user authentication scheme. Section 3 is about cryptanalysis of An's scheme. In Section 4, we present our improved scheme. Section 5 is about security analysis of the improved

Table 1: Notations with their description.

| Notations | Description |
|---|---|
| $R$ | Trusted registration centre |
| $S_i$ | Server |
| $C_i$ | User |
| $\mathrm{ID}_i$ | Identity of $C_i$ |
| $\mathrm{PW}_i$ | Password of $C_i$ |
| $B_i$ | Biometric template of $C_i$ |
| $\mathrm{SC}_i$ | Smart card of $C_i$ |
| $K_i$ | Random number chosen by $C_i$ |
| $R_c$ | Random number generated by $\mathrm{SC}_i$ of $C_i$ |
| $R_s$ | Random number generated by $S_i$ |
| $U_a$ | Attacker |
| $x_s$ and $y_s$ | Secret keys maintained by $S_i$ |
| $h(\cdot)$ | One-way hash function |
| $\oplus$ | Bitwise XOR operator |
| $\|$ | Concatenation operator |

scheme. In Section 6, we compare the improved scheme with related schemes. Finally, the conclusion is presented in Section 7.

## 2. Review of An's Scheme

The notations useful in this paper are summarized along with their description in Table 1. In this section, we review An's scheme [27] which is an enhanced version of Das's scheme [26]. It has three phases: registration phase, login phase and authentication phase. Registration phase is carried over a secure channel whereas login phase, and authentication phase are carried over an insecure channel. There are three participants in the scheme, the user ($C_i$), the server ($S_i$), and the registration centre ($R$), where $R$ is assumed to be a trusted party. Details of each phase are given in the following subsections.

*2.1. Registration Phase.* In the beginning of scheme, the registration centre $R$ and the user $C_i$ carry out this phase involving the following steps.

(1) $C_i$ submits his identity $\mathrm{ID}_i$ and information $(\mathrm{PW}_i \oplus K_i)$ containing password to $R$ via a secure channel. $C_i$ also submits information $(B_i \oplus K_i)$ containing his biometrics via the specific device to $R$; here $K_i$ is a random number chosen by $C_i$.

(2) $R$computes $f_i = h(B_i \oplus K_i)$, $r_i = h(\mathrm{PW}_i \oplus K_i) \oplus f_i$, and $e_i = h(\mathrm{ID}_i \| x_s) \oplus r_i$, where $x_s$ is a secret key generated and maintained by $S_i$. Then $R$ stores $\{\mathrm{ID}_i, f_i, e_i, h(\cdot)\}$ in a smart card $\mathrm{SC}_i$ for user and provides it to $C_i$ via a secure channel.

(3) On receiving $\mathrm{SC}_i = \{\mathrm{ID}_i, f_i, e_i, h(\cdot)\}$, the user stores the random number $K_i$ into $\mathrm{SC}_i$ issued by $R$ so that now $\mathrm{SC}_i = \{\mathrm{ID}_i, f_i, e_i, h(\cdot)\}$.

*2.2. Login Phase.* When the user $C_i$ wishes to login the server $S_i$, the user and his smart card $\mathrm{SC}_i$ perform the following steps.

(1) $C_i$ inserts his smart card into a card reader and inputs his biometrics information $B_i$ on the specific device. $\mathrm{SC}_i$ computes $h(B_i \oplus K_i)$ and verifies if $f_i = h(B_i \oplus K_i)$ or not. If this biometrics information matches, $C_i$ passes the biometrics verification.

(2) $C_i$ inputs his $\mathrm{ID}_i$ and $\mathrm{PW}_i$; then $\mathrm{SC}_i$ generates a random number $R_c$ and computes the following equations:

$$\begin{aligned} r_i' &= h\left(\mathrm{PW}_i \oplus K_i\right) \oplus f_i, \\ M_1 &= e_i \oplus r_i', \\ M_2 &= M_1 \oplus R_c, \\ M_3 &= h\left(M_1 \| R_c\right). \end{aligned} \tag{1}$$

(3) $C_i$ sends the login request $= \{\mathrm{ID}_i, M_2, M_3\}$ to $S_i$.

*2.3. Authentication Phase.* On receiving the request login $= \{\mathrm{ID}_i, M_2, M_3\}$ from $C_i$, the server $S_i$ and the user $C_i$ perform the following steps to authenticate each other.

(1) $S_i$ first checks the format of $\mathrm{ID}_i$. If $\mathrm{ID}_i$ is valid, $S_i$ computes $M_4 = h(\mathrm{ID}_i \| x_s)$ and $M_5 = M_2 \oplus M_4$.

(2) $S_i$ checks if $M_3 = h(M_4 \| M_5)$ or not. If both are equal, it generates a random number $R_s$ and computes the following equations:

$$\begin{aligned} M_6 &= M_4 \oplus R_s, \\ M_7 &= h\left(M_4 \| R_s\right). \end{aligned} \tag{2}$$

Then, $S_i$ sends the reply message $= \{M_6, M_7\}$ for its authentication to $C_i$.

(3) On receiving $\{M_6, M_7\}$ from $S_i$, the user $C_i$ computes $M_8 = M_6 \oplus M_1$ and checks if $M_7 = h(M_1 \| M_8)$ or not. If both are equal, $C_i$ computes $M_9 = h(M_1 \| R_c \| M_8)$ and sends the reply message $\{M_9\}$ for its authentication to $S_i$.

(4) On receiving $\{M_9\}$ from $C_i$, the server checks if $M_9 = h(M_4 \| M_5 \| R_s)$ or not. If both are equal, $S_i$ accepts the login request $= \{\mathrm{ID}_i, M_2, M_3\}$ of $C_i$.

## 3. Cryptanalysis of An's Scheme

This section is about security problems in An's scheme. Here we show that an attacker $U_a$ can mount different types of attacks on the scheme. Independent researches by Kocher and Messerges [28, 29] show that it is possible to extract the values stored inside a smart card. So we assume that $U_a$ can extract out parameters stored inside a user's smart card.

*3.1. Online Password Guessing Attack.* If $U_a$ obtains the smart card $SC_i$ of user $C_i$ and extracts [28, 29] the values $\{ID_i, f_i, e_i, K_i, h(\cdot)\}$ stored inside it, then he can mount online password guessing attack as explained below.

(1) $U_a$ computes

$$\begin{aligned} e_i \oplus f_i &= [h(ID_i \parallel x_s) \oplus r_i] \oplus f_i \\ &= [h(ID_i \parallel x_s) \oplus h(PW_i \oplus K_i) \oplus f_i] \oplus f_i \\ &= [h(ID_i \parallel x_s) \oplus h(PW_i \oplus K_i)] \end{aligned} \quad (3)$$

to obtain $[h(ID_i \parallel x_s) \oplus h(PW_i \oplus K_i)]$.

(2) $U_a$ guesses $PW_a$ as user's possible password and computes $M_{1a} = [e_i \oplus f_i] \oplus h(PW_a \oplus K_i)$. Then $U_a$ computes $M_{2a} = M_{1a} \oplus R_{ca}$ and $M_{3a} = h(M_{1a} \parallel R_{ca})$, where $R_{ca}$ is the random number generated by the system of $U_a$. *He* sends $\{ID_i, M_{2a}, M_{3a}\}$ as login request to $S_i$.

(3) If $U_a$ does not receive any response from $S_i$ then he repeats step (2) with some other guess for user's password. But if $U_a$ receives response message from $S_i$, then it implies that his guessed password $PW_a$ is correct.

*3.2. Offline Password Guessing Attack.* In the scheme, $U_a$ can easily identify the login request corresponding to a smart card since both contain the identity of user. If $U_a$ extracts [28, 29] the values $\{ID_i, f_i, e_i, K_i, h(\cdot)\}$ from the smart card $SC_i$ of user $C_i$ and intercepts the login request = $\{ID_i, M_2, M_3\}$ from open network, then he can mount offline password guessing attack as explained below.

(1) $U_a$ computes

$$\begin{aligned} e_i \oplus f_i &= [h(ID_i \parallel x_s) \oplus r_i] \oplus f_i \\ &= [h(ID_i \parallel x_s) \oplus h(PW_i \oplus K_i) \oplus f_i] \oplus f_i \\ &= [h(ID_i \parallel x_s) \oplus h(PW_i \oplus K_i)] \end{aligned} \quad (4)$$

to obtain $[h(ID_i \parallel x_s) \oplus h(PW_i \oplus K_i)]$.

(2) $U_a$ guesses $PW_a$ as user's possible password and computes $M_{1a} = [e_i \oplus f_i] \oplus h(PW_a \oplus K_i)$.

(3) $U_a$ computes $R_{ca} = M_2 \oplus M_{1a}$ and $M_{3a} = h(M_{1a} \parallel R_{ca})$, and finally compares $M_{3a}$ with $M_3$. For $M_{3a} \neq M_3$, he repeats from step (2) with some other guess for user's password. But if $M_{3a} = M_3$, then it provides $U_a$ with the exact password $PW_i$ of $C_i$.

*3.3. User Impersonation Attack.* As just discussed in previous subsections, $U_a$ can guess a user's password if he obtains the smart card of user. It is noticeable that the successful process of password guessing (online or offline manner) also yields $M_{1a} = h(ID_i \parallel x_s)$. In fact, $h(ID_i \parallel x_s)$ is the key value required to compute a valid login request or valid reply messages. Further, $U_a$ has easy access to user's identity $ID_i$ from $SC_i = \{ID_i, f_i, e_i, K_i, h(\cdot)\}$ or from the login request = $\{ID_i, M_2, M_3\}$ of $C_i$. Having $h(ID_i \parallel x_s)$ and $ID_i$ in hand, $U_a$ can impersonate the user $C_i$ as explained below.

(1) $U_a$ generates a random number $R_{ca}$ in his system and computes

$$\begin{aligned} M_{2a} &= M_{1a} \oplus R_{ca}, \\ M_{3a} &= h(M_{1a} \parallel R_{ca}). \end{aligned} \quad (5)$$

Then $U_a$ sends the login request = $\{ID_i, M_{2a}, M_{3a}\}$ to $S_i$.

(2) On receiving $\{ID_i, M_{2a}, M_{3a}\}$, the server $S_i$ first checks the format of $ID_i$. Clearly, $S_i$ would proceed further because $ID_i$ is the identity of a legitimate registered user and hence it is in valid format.

(3) $S_i$ computes $M_4 = h(ID_i \parallel x_s)$ and $M_5 = M_{2a} \oplus M_4$ and checks if $M_{3a} = h(M_4 \parallel M_5)$; clearly it would hold. Therefore $S_i$ believes that the login request = $\{ID_i, M_{2a}, M_{3a}\}$ is from the legitimate user.

(4) $S_i$ generates a random number $R_s$ and computes $M_6 = M_4 \oplus R_s$ and $M_7 = h(M_4 \parallel R_s)$. Then $S_i$ transmits the reply message $\{M_6, M_7\}$.

(5) On receiving $\{M_6, M_7\}$ from $S_i$, the attacker $U_a$ first obtains the random number $R_s$ by computing $M_{8a} = M_6 \oplus M_{1a}$. Next, it computes $M_{9a} = h(M_{1a} \| R_{ca} \| M_8)$ and sends $\{M_{9a}\}$ to $S_i$.

(6) On receiving $\{M_9\}$, the server $S_i$ checks if $M_9 = h(M_4 \| M_5 \| R_s)$ or not. Clearly, this would hold, so $S_i$ will accept the login request = $\{ID_i, M_{2a}, M_{3a}\}$.

*3.4. Server Impersonation Attack.* $U_a$ can easily impersonate the legal server $S_i$ to cheat the user $C_i$ whose information $\{ID_i$ and $M_{1a} = h(ID_i \parallel x_s)\}$ he possesses as described in Section 3.3. To masquerade as $S_i$ the attacker proceeds in the following manner.

(1) $U_a$ can easily recognize the login request = $\{ID_i, M_2, M_3\}$ of $C_i$ transmitted over open channel as he possesses the identity $ID_i$ of $C_i$. So when $C_i$ sends his login request = $\{ID_i, M_2, M_3\}$ to $S_i$, the attacker $U_a$ intercepts and blocks it from reaching $S_i$.

(2) $U_a$ first obtains the random number $R_c$ by computing $M_{5a} = M_2 \oplus M_{1a}$. Next, he generates a random number $R_{sa}$ in his system and computes $M_{6a} = M_{1a} \oplus R_{sa}$ and $M_{7a} = h(M_{1a} \parallel R_{sa})$. Then $U_a$ transmits the reply message $\{M_{6a}, M_{7a}\}$ to $C_i$.

(3) On receiving $\{M_{6a}, M_{7a}\}$, the user $C_i$ first obtains the random number $R_{sa}$ by computing $M_8 = M_{6a} \oplus M_1$, where $M_1 = h(ID_i \parallel x_s)$. Next, he checks if $M_{7a} = h(M_1 \parallel M_8)$ or not. Clearly, this equivalence will hold and hence $C_i$ will believe that he is communicating with the intended server. However, it is the clever attacker $U_a$ who is deceiving $C_i$.

*3.5. Lack of Mutual Authentication.* Like Das's scheme [26], the enhanced scheme by An also fails to resist user impersonation attack and server impersonation attack as described in Sections 3.3 and 3.4. In fact, if $U_a$ extracts values $\{\mathrm{ID}_i, f_i, e_i, K_i, h(\cdot)\}$ from the smart card $\mathrm{SC}_i$ of user $C_i$ and successfully obtains the secret value $h(\mathrm{ID}_i \parallel x_s)$, then he can easily craft valid login request and reply messages so as to deceive the legal user or the legal server. Therefore, the scheme loses mutual authentication feature.

*3.6. Lack of User Anonymity.* In An's scheme, $C_i$ sends $\{\mathrm{ID}_i, M_2, M_3\}$ as his login request to $S_i$ through an insecure channel. User's identity $\mathrm{ID}_i$ is openly available if an attacker $U_a$ intercepts the login request of $C_i$ from the open channel. Moreover, identity $\mathrm{ID}_i$ is also stored inside user's smart card $\mathrm{SC}_i$. Having $\mathrm{ID}_i$ in hand, it is easy for $U_a$ to craft threats against $C_i$. To the worst, $U_a$ may be able to compromise user's biometrics information which would result in serious consequences. Thus, the scheme does not provide user anonymity.

## 4. The Proposed Scheme

In this section, we propose a new user authentication scheme which is an improvement of An's scheme. In addition to resist the security problems found in An's scheme, it also provides password change phase with which user can change his password at his will. It has four phases: registration phase, login phase, authentication phase and password change phase. Registration phase, and password change phase are carried over a secure channel whereas login phase and authentication phase are carried over an insecure channel. It also consists of three participants, the user ($C_i$), the server ($S_i$), and the registration centre ($R$). In the proposed scheme, the server maintains two secret keys $x_s$ and $y_s$. Details of each phase along with Figure 1 are given in the following.

*4.1. Registration Phase.* Before starting the scheme, the registration centre $R$ and the user $C_i$ carry out this phase involving the following steps.

(1) $C_i$ submits his identity $\mathrm{ID}_i$ and information $(\mathrm{PW}_i \oplus K_i)$ containing password to $R$ via a secure channel. $C_i$ also submits information $(B_i \oplus K_i)$ containing his biometrics via a specific device to $R$; here $K_i$ is a random number chosen by $C_i$.

(2) $R$ computes the following values:

$$\begin{aligned} f_i &= h(B_i \oplus K_i), \\ r_i &= h(\mathrm{PW}_i \oplus K_i) \oplus f_i, \\ c_i &= h(x_s \parallel y_s) \oplus f_i, \\ e_i &= h(\mathrm{ID}_i \parallel x_s) \oplus r_i, \end{aligned} \tag{6}$$

where $R$ stores $\{c_i, e_i, h(\cdot)\}$ in a smart card $\mathrm{SC}_i$ for user. Then $R$ provides $\mathrm{SC}_i = \{c_i, e_i, h(\cdot)\}$ and $f_i$ to the user $C_i$ via a secure channel.

(3) On receiving $[\mathrm{SC}_i = \{c_i, e_i, h(\cdot)\} \ \& \ f_i]$, the user computes the following values:

$$\begin{aligned} g_i &= (\mathrm{ID}_i \parallel \mathrm{PW}_i) \oplus f_i, \\ j_i &= (\mathrm{ID}_i \parallel \mathrm{PW}_i) \oplus K_i, \end{aligned} \tag{7}$$

where $C_i$ inserts $g_i$ and $j_i$ into $\mathrm{SC}_i$ issued by $R$ so that now $\mathrm{SC}_i = \{c_i, e_i, g_i, j_i, h(\cdot)\}$.

*4.2. Login Phase.* When the user $C_i$ wishes to login the server $S_i$, the user and his smart card $\mathrm{SC}_i$ perform the following steps.

(1) $C_i$ inserts his smart card into a card reader, keys in his identity $\mathrm{ID}_i$, and password $\mathrm{PW}_i$ and inputs his biometrics information $B_i$ on the specific device.

(2) $\mathrm{SC}_i$ retrieves $f_i \leftarrow (\mathrm{ID}_i \parallel \mathrm{PW}_i) \oplus g_i$ and $K_i \leftarrow (\mathrm{ID}_i \parallel \mathrm{PW}_i) \oplus j_i$. It then checks if $f_i = h(B_i \oplus K_i)$ or not. If this biometrics information matches, $C_i$ passes the biometrics verification; otherwise $\mathrm{SC}_i$ terminates the sesion. This process also verifies the correctness of inserted $\mathrm{ID}_i$ and $\mathrm{PW}_i$.

(3) $\mathrm{SC}_i$ generates a random number $R_c$ and computes the following equations:

$$\begin{aligned} r_i &= h(\mathrm{PW}_i \oplus K_i) \oplus f_i, \\ M_1 &= c_i \oplus f_i \quad (\text{which is indeed } h(x_s \parallel y_s)), \\ M_2 &= e_i \oplus r_i \quad (\text{which is indeed } h(\mathrm{ID}_i \parallel x_s)), \\ M_3 &= M_1 \oplus R_c \quad (\text{which is indeed } h(x_s \parallel y_s) \oplus R_c), \\ M_4 &= (M_1 \parallel R_c) \oplus \mathrm{ID}_i \\ &\quad (\text{which is indeed } [(h(x_s \parallel y_s) \parallel R_c) \oplus \mathrm{ID}_i]), \\ M_5 &= h(M_2 \parallel R_c), \\ &\quad (\text{which is indeed } h(h(\mathrm{ID}_i \parallel x_s) \parallel R_c)). \end{aligned} \tag{8}$$

(4) $C_i$ sends the login request $= \{M_3, M_4, M_5\}$ to $S_i$.

*4.3. Authentication Phase.* On receiving the request login $= \{M_3, M_4, M_5\}$ from $C_i$, the server $S_i$ and the user $C_i$ perform the following steps to authenticate each other.

(1) $S_i$ computes the following values:

$$\begin{aligned} M_6 &= h(x_s \parallel y_s), \\ M_7 &= M_3 \oplus M_6 \quad (\text{which is indeed } R_c), \\ \mathrm{ID}_i &= M_4 \oplus (M_6 \parallel M_7). \end{aligned} \tag{9}$$

(2) $S_i$ checks the format of $\mathrm{ID}_i$. If $\mathrm{ID}_i$ is valid, $S_i$ computes $M_8 = h(\mathrm{ID}_i \parallel x_s)$. It then checks if $M_5 = h(M_8 \parallel M_7)$.

| User ($C_i$) | | Registration centre ($R$) |
|---|---|---|
| *Registration phase* | | Computes $f_i = h(B_i \bigoplus K_i)$, |
| Chooses $ID_i$, $PW_i$ & $K_i$ | $\xrightarrow{\{ID_i, (PW_i \bigoplus K_i), (B_i \bigoplus K)\}}$ | $r_i = h(PW_i \bigoplus K_i) \bigoplus f_i$, |
| | $\xleftarrow{[SC_i = \{c_i, e_i, h(\cdot)\}\ \&\ f_i]}$ | $c_i = h(x_s \| \| y_s) \bigoplus f_i$, and |
| | | $e_i = h(ID_i \| \| x_s) \bigoplus r_i$ |
| Computes | | |
| $g_i = (ID_i \| \| PW_i) \bigoplus f_i$ and | | |
| $j_i = (ID_i \| \| PW_i) \bigoplus K_i$ | | |
| Insert $g_i$ & $j_i$ into $SC_i$ so that $SC_i = \{c_i, e_i, g_i, j_i, h(\cdot)\}$ | | |

| User ($C_i$) | | Server ($S_i$) |
|---|---|---|
| *Login and authentication phase* | | |
| $U$: inserts $ID_i$, $PW_i$ & $B_i$ | | Computes $M_6 = h(x_s \| \| y_s)$, |
| SC : $f_i \leftarrow (ID_i \| \| PW_i) \bigoplus g_i$ and | | $M_7 = M_3 \bigoplus M_6$, and |
| $K_i \leftarrow (ID_i \| \| PW_i) \bigoplus j_i$ | | $ID_i = M_4 \bigoplus (M_6 \| \| M_7)$ |
| For $f_i = h(B_i \bigoplus K_i)$ computes | | For correct $ID_i$ format computes |
| $r_i = h(PW_i \bigoplus K_i) \bigoplus f_i$, | $\xrightarrow{\{M_3, M_4, M_5\}}$ | $M_8 = h(ID_i \| \| x_s)$ |
| $M_1 = c_i \bigoplus f_i, M_2 = e_i \bigoplus r_i$, | | For $M_5 = h(M_8 \| \| M_7)$ computes |
| $M_3 = M_1 \bigoplus R_c, M_4 = (M_1 \| \| R_c) \bigoplus ID_i$, | | $M_9 = M_8 \bigoplus R_s$ and |
| $M_5 = h(M_2 \| \| R_c)$. | | $M_{10} = h(M_8 \| \| R_s)$ |
| | $\xleftarrow{\{M_9, M_{10}\}}$ | |
| Computes $M_{11} = M_9 \bigoplus M_2$ | | |
| For $M_{10} = h(M_2 \| \| M_{11})$ | | |
| Computes $M_{12} = h(M_2 \| \| R_c \| \| M_{11})$ | | |
| | $\xrightarrow{\{M_{12}\}}$ | For $M_{12} = h(M_8 \| \| M_7 \| \| R_s)$ |
| | | Accepts login request |

| User ($C_i$) | | Smart card ($SC_i$) |
|---|---|---|
| *Password change phase*: | | $f_i \leftarrow (ID_i \| \| PW_i) \bigoplus g_i$ and |
| $U$: inserts $ID_i$, $PW_i$ & $B_i$ | $\xrightarrow{\{ID_i, PW_i\ \&\ B_i\}}$ | $K_i \leftarrow (ID_i \| \| PW_i) \bigoplus j_i$ |
| | | For $f_i = h(B_i \bigoplus K_i)$ asks for new password |
| | $\xrightarrow{(PW_i)_{new}}$ | Computes $(g_i)_{new} = (ID_i \| \| (PW_i)_{new}) \bigoplus f_i$, |
| | | $(j_i)_{new} = (ID_i \| \| (PW_i)_{new} \bigoplus K_i$, and |
| | | $(e_i)_{new} =$ |
| | | $e_i \bigoplus h(PW_i \bigoplus K_i) \bigoplus h((PW_i)_{new} \bigoplus K_i)$. |
| | | $(e_i)_{new} \leftarrow e_i, (g_i)_{new} \leftarrow g_i, (j_i)_{new} \leftarrow j_i$ |

Figure 1: The proposed scheme.

If both are equal, $S_i$ generates a random number $R_s$ and computes:

$$\begin{aligned} &M_9 = M_8 \oplus R_s \\ &\left(\text{which is indeed } h\left(\text{ID}_i \parallel x_s\right) \oplus R_s\right) \\ &M_{10} = h\left(M_8 \parallel R_s\right) \\ &\left(\text{which is indeed } h\left(h\left(\text{ID}_i \parallel x_s\right) \parallel R_s\right)\right). \end{aligned} \tag{10}$$

Then, $S_i$ sends the reply message = $\{M_9, M_{10}\}$ for its authentication to $C_i$.

(3) On receiving $\{M_9, M_{10}\}$ from $S_i$, the user $C_i$ computes $M_{11} = M_9 \oplus M_2$ (which is indeed $R_s$). It then checks if $M_{10} = h(M_2 \parallel M_{11})$ or not. If both are equal, $C_i$ computes $M_{12} = h(M_2 \parallel R_c \parallel M_{11})$ (which is indeed $h[h(\text{ID}_i \parallel x_s) \parallel R_c \parallel R_s]$). Then $C_i$ sends the reply message $\{M_{12}\}$ for its authentication to $S_i$.

(4) On receiving $\{M_{12}\}$ from $C_i$, the server checks if $M_{12} = h(M_8 \parallel M_7 \parallel R_s)$ or not. If both are equal, $S_i$ accepts the login request = $\{M_3, M_4, M_5\}$ of $C_i$.

*4.4. Password Change Phase.* When the user wishes to change his old password $\text{PW}_i$, he invokes this phase. Details of the steps required to update the smart card $\text{SC}_i$ with new password $(\text{PW}_i)_{\text{new}}$ are as follows.

(1) $C_i$ inserts his smart card into a card reader, keys in his identity $\text{ID}_i$, and password $\text{PW}_i$ and inputs his biometrics information $B_i$ on the specific device.

(2) $\text{SC}_i$ retrieves $f_i \leftarrow (\text{ID}_i \parallel \text{PW}_i) \oplus g_i$ and $K_i \leftarrow (\text{ID}_i \parallel \text{PW}_i) \oplus j_i$. It then checks if $f_i = h(B_i \oplus K_i)$ or not. If this biometrics information matches, $C_i$ passes the biometrics verification, otherwise terminates the session. This process also verifies the correctness of inserted $\text{ID}_i$ and $\text{PW}_i$. Then $\text{SC}_i$ allows the user to enter the new password $(\text{PW}_i)_{\text{new}}$.

(3) $\text{SC}_i$ computes the following equations:

$$\begin{aligned} (g_i)_{\text{new}} &= \left(\text{ID}_i \parallel (\text{PW}_i)_{\text{new}}\right) \oplus f_i, \\ (j_i)_{\text{new}} &= \left(\text{ID}_i \parallel (\text{PW}_i)_{\text{new}}\right) \oplus K_i, \\ (e_i)_{\text{new}} &= e_i \oplus h\left(\text{PW}_i \oplus K_i\right) \oplus h\left((\text{PW}_i)_{\text{new}} \oplus K_i\right). \end{aligned} \tag{11}$$

(4) $\text{SC}_i$ replaces $e_i$, $g_i$, and $j_i$ with $(e_i)_{\text{new}}$, $(g_i)_{\text{new}}$ and $(j_i)_{\text{new}}$, respectively.

## 5. Security Analysis of the Proposed Scheme

In this section, we analyze security of the proposed scheme. We show that the scheme remains unaffected even if an attacker $U_a$ extracts [28, 29] all the values stored inside a user's smart card.

*5.1. Online Password Guessing Attack.* On having access to user's smart card $\text{SC}_i$ an attacker $U_a$ can extract [28, 29] all values $\{c_i, e_i, g_i, j_i, h(\cdot)\}$ from it. In order to compute $e_i \oplus f_i$ and obtain $[h(\text{ID}_i \parallel x_s) \oplus h(\text{PW}_i \oplus K_i)]$, he requires $f_i$. But $U_a$ cannot obtain $f_i$ from $g_i = (\text{ID}_i \parallel \text{PW}_i) \oplus f_i$ as he does not know about user's identity $\text{ID}_i$ and password $\text{PW}_i$. The attacker $U_a$ can obtain $f_i \oplus K_i$ by performing $g_i \oplus j_i = [(\text{ID}_i \parallel \text{PW}_i) \oplus f_i] \oplus [(\text{ID}_i \parallel \text{PW}_i) \oplus K_i]$. Next, he can compute

$$\begin{aligned} &e_i \oplus \left(f_i \oplus K_i\right) \\ &\quad = \left[h\left(\text{ID}_i \parallel x_s\right) \oplus r_i\right] \oplus \left(f_i \oplus K_i\right) \\ &\quad = \left[h\left(\text{ID}_i \parallel x_s\right) \oplus h\left(\text{PW}_i \oplus K_i\right) \oplus f_i\right] \oplus \left(f_i \oplus K_i\right) \\ &\quad = h\left(\text{ID}_i \parallel x_s\right) \oplus h\left(\text{PW}_i \oplus K_i\right) \oplus K_i. \end{aligned} \tag{12}$$

But $U_a$ cannot compute forged $M_{2a}$ $(= h(\text{ID}_i \parallel x_s)) = [e_i \oplus f_i \oplus K_i] \oplus h(\text{PW}_a \oplus K_i)$ using a guessed password $\text{PW}_a$ because it requires knowledge of $K_i$. It is troublesome for $U_a$ to obtain $K_i$ because $K_i$ is not stored in plaintext inside user's smart card but is stored securely in $j_i = (\text{ID}_i \parallel \text{PW}_i) \oplus K_i$. Further $U_a$ cannot obtain $K_i$ from $j_i$ without knowing $\text{ID}_i$ and password $\text{PW}_i$. Besides, $U_a$ cannot compute $M_{1a}$ $(= h(x_s \parallel y_s)) = (c_i \oplus f_i)$ as he does not have access to $f_i$. Moreover, $U_a$ does not have $\text{ID}_i$ of $C_i$ as $\text{ID}_i$ is not stored in plaintext inside user's smart card. Thus, $U_a$ cannot compute a login request $\{M_{3a}, M_{4a}, M_{5a}\}$ in a way so as to guess user's password in an online manner. Hence, the proposed scheme withstands online password guessing attack.

*5.2. Offline Password Guessing Attack.* Suppose $U_a$ obtains the smart card of some user. Though $U_a$ can intercept login message of any user from open channel, he cannot relate a user's smart card with its corresponding login request. This is due to the fact that, unlike An's scheme, in the proposed scheme user's identity in plaintext is neither stored inside user's smart card nor transmitted in login request. As a result, $U_a$ cannot combine values extracted from a user's smart card with values of corresponding login request to guess user's password in an offline manner. If we consider the situation that $U_a$ somehow happens to get the correct combination of user's smart card and login request, we show that still $U_a$ cannot mount offline password guessing attack. To guess password of $C_i$ and then verify the guess, $U_a$ can use $M_5 = h(M_2 \parallel R_c)$ provided that he possesses the values $\{[h(\text{ID}_i \parallel x_s) \oplus h(\text{PW}_i \oplus K_i) \oplus K_i], K_i \text{ and } R_c\}$ in hand. As explained in Section 5.1, $U_a$ can obtain $[h(\text{ID}_i \parallel x_s) \oplus h(\text{PW}_i \oplus K_i) \oplus K_i]$ using $\{g_i, j_i \text{ and } e_i\}$ extracted [28, 29] from $\text{SC}_i$, but he cannot obtain the random number $K_i$. Besides, $U_a$ cannot obtain the random number $R_c$ using $M_3 = M_1 \oplus R_c$ without having $M_1$ $(= h(x_s \parallel y_s))$ and $U_a$ fails to obtain $M_1$ $(= h(x_s \parallel y_s))$ as discussed in Section 5.1. Thus an attacker $U_a$ cannot guess user's password in an offline manner.

*5.3. User Impersonation and Server Impersonation Attack.* To impersonate a legal user, $U_a$ should possess $M_1 = h(x_s \parallel y_s)$ and $M_2 = h(\text{ID}_i \parallel x_s)$; otherwise he cannot compute a valid login request $\{M_{3a}, M_{4a}, M_{5a}\}$ or a valid reply message $\{M_{12a}\}$. The value $h(\text{ID}_i \parallel x_s)$ is equally important if $U_a$ wishes to masquerade as legal server. Unlike An's scheme, in the proposed scheme $U_a$ is not able to obtain $M_2$ $(= M_8) = h(\text{ID}_i \parallel x_s)$ while making attempts of guessing user's

Table 2: Comparison of security attributes.

| Security attributes | Schemes | | | |
|---|---|---|---|---|
| | Li-Hwang's [19] | Das's [26] | An's [27] | Ours |
| Resist online $PW_i$ guessing attack | No | No | No | Yes |
| Resist offline $PW_i$ guessing attack | No | No | No | Yes |
| Resist user impersonation attack | No | No | No | Yes |
| Resist server impersonation attack | No | No | No | Yes |
| Provides mutual authentication | No | No | No | Yes |
| Provides $PW_i$ change facility | Yes | Yes | No | Yes |
| Provides user anonymity | No | No | No | Yes |

password. This is due to the fact that password guessing is not feasible as explained in Sections 5.1 and 5.2. Moreover, $U_a$ cannot obtain $M_1 = h(x_s \parallel y_s)$ (i) from $M_3 = M_1 \oplus R_c$ obtained by intercepting the login request of $C_i$ because of not having random number $R_c$ and (ii) from $c_i = h(x_s \parallel y_s) \oplus f_i$ extracted from user's smart card without knowing $f_i$. Thus, the proposed scheme resists impersonation attacks.

*5.4. Supporting Mutual Authentication.* The success of mutual authentication in the proposed scheme follows directly from resistance against user impersonation attack and server impersonation attack as described in Section 5.3. In fact, $U_a$ has many hurdles before him to act as a legal user or a legal server: (i) the secret keys $x_s$ and $y_s$ maintained by the server are unknown for $U_a$ and (ii) $U_a$ has no access to the identity $ID_i$ of user $C_i$. As a result, $U_a$ cannot compute $h(x_s \parallel y_s)$ and $h(ID_i \parallel x_s)$ required to mount impersonation attacks. Besides, $U_a$ has no method to retrieve these values either from the parameters extracted out of user's smart card or from the login request or using both. Therefore, the proposed scheme provides proper mutual authentication.

*5.5. Providing User Anonymity and User Untraceability.* In the proposed scheme, user's plaintext identity $ID_i$ is completely out of scene; it is neither stored in user's smart card $SC_i$ nor sent in any of the login-authentication messages transmitted over insecure network. If $U_a$ extracts [28, 29] the values $\{c_i, e_i, g_i, j_i, h(\cdot)\}$ from $SC_i$, we explain in the following that he cannot obtain $ID_i$ of $C_i$. To guess $ID_i$ from $g_i = (ID_i \parallel PW_i) \oplus f_i$ and from $j_i = (ID_i \parallel PW_i) \oplus K_i$, the attacker must have the knowledge of $\{PW_i, f_i\}$ and $\{PW_i, K_i\}$, respectively. $U_a$ cannot guess out $ID_i$ from $e_i = h(ID_i \parallel x_s) \oplus r_i$ without knowing $r_i$ and $x_s$. If $U_a$ intercepts a login request $\{M_3, M_4, M_5\}$ or the reply message $\{M_9, M_{10}\}/\{M_{12}\}$, he cannot guess out $ID_i$ using $\{M_5, M_{10}, M_{12}\}$ without the knowledge of $\{x_s, R_c$ and $R_s\}$. Besides, it is not feasible for $U_a$ to retrieve $ID_i$ out of $\{e_i, M_5, M_{10}, M_{12}\}$ due to one-way property of hash function. Moreover, each value $\{M_3, M_4, M_5, M_9, M_{10}, M_{12}\}$ transmitted over insecure network is dynamic in nature by virtue of random numbers $R_c$ and $R_s$ which are different for each session. Thus, $U_a$ can neither obtain user's identity $ID_i$ nor can he trace the legal user by means of observing and analyzing some fixed parameter in the login request or the reply messages. Hence, the scheme provides user anonymity as well as user untraceability.

Table 3: Comparison of computational load in terms of hash functions.

| Phases | Schemes | | | |
|---|---|---|---|---|
| | Li-Hwang's [19] | Das's [26] | An's [27] | Ours |
| Registration phase | 3 $h(\cdot)$ | 3 $h(\cdot)$ | 3 $h(\cdot)$ | 4 $h(\cdot)$ |
| Login phase | 2 $h(\cdot)$ | 2 $h(\cdot)$ | 3 $h(\cdot)$ | 3 $h(\cdot)$ |
| Authentication phase | 5 $h(\cdot)$ | 8 $h(\cdot)$ | 6 $h(\cdot)$ | 7 $h(\cdot)$ |
| Total | 10 $h(\cdot)$ | 13 $h(\cdot)$ | 12 $h(\cdot)$ | 14 $h(\cdot)$ |

*5.6. Providing Password Change Facility.* In An's scheme, once user chooses his password during registration phase, it is fixed forever as user cannot change his password at his will. Probably the author might have opined that in the presence of biometrics verification procedure there is no need of password change facility. Undoubtedly, it is very difficult to forge copy or compromise biometrics, but once compromised then biometrics cannot be changed like passwords. So we opine that if password is employed in user authentication scheme then there should be the provision to facilitate the user to freely change his password. The proposed scheme provides password changing facility with which a user can freely (without interacting with server) change his old password to a new one whenever he feels to do so. Before updating stored values with the new password $(PW_i)_{new}$, the smart card verifies the correctness of identity $ID_i$ old password $PW_i$ along with verifying the biometrics information $f_i = h(B_i \oplus K_i)$. Thus the proposed scheme provides secure and easy password changing facility.

## 6. Comparison

In this section, we examine the proposed scheme by means of comparing its efficiency with Li-Hwang's scheme [19], Das's scheme [26], and An's scheme [27]. Table 2 displays comparison of security attributes and Table 3 displays comparison of computational load in terms of hash functions. Comparison in Table 2 shows that the proposed scheme resists various attacks possible on schemes [19, 26, 27] and provides additional feature of user anonymity with untraceability. Besides, it also restores password change facility which is provided by original versions [19, 26] but is missing in An's scheme [27]. As Table 3 shows, the proposed scheme carries only two additional hash operations over its immediate

predecessor scheme [27]. The important aspect about the proposed scheme is minor increase of two hash functions in computational load to achieve higher efficiency as compared to other schemes [19, 26, 27].

## 7. Conclusion

This paper shows that the recently proposed biometrics-based user authentication scheme by An is susceptible to many threats. Once an attacker obtains the smart card of a legal user, he can guess user's password and impersonate the user. Further, the attacker can also cheat the user by masquerading as the legal server. Consequently, the scheme fails to provide mutual authentication. Besides, the scheme also suffers from the restriction of static password. We have proposed a new scheme based on the design of An's scheme so as to fix the problems identified in An's scheme. In the proposed scheme an attacker cannot figure out the identity of user either from the smart card or by intercepting all login-authentication messages transmitted over insecure network. Analysis and comparison show improved performance of the proposed scheme.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgment

The authors would like to extend their sincere appreciation to the Deanship of Scientific Research at King Saud University for its funding of this research through the Research Group Project no. RGP-VPP-288.

## References

[1] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, vol. 24, no. 11, pp. 770–772, 1981.

[2] N. M. Haller, "The S/KEY one-time password system," RFC1760, February 1995.

[3] G. Horng, "Password authentication without using a password table," *Information Processing Letters*, vol. 55, no. 5, pp. 247–250, 1995.

[4] J.-K. Jan and Y.-Y. Chen, "'Paramita wisdom' password authentication scheme without verification tables," *The Journal of Systems and Software*, vol. 42, no. 1, pp. 45–57, 1998.

[5] M.-S. Hwang and L.-H. Li, "A new remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 28–30, 2000.

[6] W.-C. Ku and S.-M. Chen, "Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 204–207, 2004.

[7] C.-I. Fan, Y.-C. Chan, and Z.-K. Zhang, "Robust remote authentication scheme with smart cards," *Computers and Security*, vol. 24, no. 8, pp. 619–628, 2005.

[8] J.-Y. Liu, A.-M. Zhou, and M.-X. Gao, "A new mutual authentication scheme based on nonce and smart cards," *Computer Communications*, vol. 31, no. 10, pp. 2205–2209, 2008.

[9] M. Kumar, M. K. Gupta, and S. Kumari, "An improved efficient remote password authentication scheme with smart card over insecure networks," *International Journal of Network Security*, vol. 13, no. 3, pp. 167–177, 2011.

[10] M. Kumar, M. K. Gupta, and S. Kumari, "An improved smart card based remote user authentication scheme with session key agreement during the verification phase," *Journal of Applied Computer Science & Mathematics*, vol. 11, no. 5, pp. 38–46, 2011.

[11] S. Kumari, M. K. Gupta, and M. Kumar, "Cryptanalysis and security enhancement of Chen *et al.'s* remote user authentication scheme using smart card," *Central European Journal of Computer Science*, vol. 2, no. 1, pp. 60–75, 2012.

[12] S. Kumari, F. B. Muhaya, M. K. Khan, and R. Kumar, "Cryptanalysis of 'a robust smart-card-based remote user password authentication scheme'," in *Proceedings of the International Symposium on Biometrics and Security Technologies*, Chengdu, China, July 2013.

[13] S. Kumari and M. K. Khan, "Cryptanalysis and improvement of 'a robust smart-card-based remote user password authentication scheme'," *International Journal of Communication Systems*, 2013.

[14] J. K. Lee, S. R. Ryu, and K. Y. Yoo, "Fingerprint-based remote user authentication scheme using smart cards," *Electronics Letters*, vol. 38, no. 12, pp. 554–555, 2002.

[15] C.-H. Lin and Y.-Y. Lai, "A flexible biometrics remote user authentication scheme," *Computer Standards and Interfaces*, vol. 27, no. 1, pp. 19–23, 2004.

[16] M. K. Khan and J. Zhang, "Improving the security of 'a flexible biometrics remote user authentication scheme'," *Computer Standards and Interfaces*, vol. 29, no. 1, pp. 82–85, 2007.

[17] M. K. Khan, J. Zhang, and X. Wang, "Chaotic hash-based fingerprint biometric remote user authentication scheme on mobile devices," *Chaos, Solitons & Fractals*, vol. 35, no. 3, pp. 519–524, 2008.

[18] M. K. Khan, "Fingerprint biometric-based self-authentication and deniable authentication schemes for the electronic world," *IETE Technical Review*, vol. 26, no. 3, pp. 191–195, 2009.

[19] C.-T. Li and M.-S. Hwang, "An efficient biometrics-based remote user authentication scheme using smart cards," *Journal of Network and Computer Applications*, vol. 33, no. 1, pp. 1–5, 2010.

[20] M. K. Khan, S. Kumari, and M. K. Gupta, "More efficient key-hash based fingerprint remote authentication scheme using mobile device," *Computing*, 2013.

[21] M. K. Khan, S.-K. Kim, and K. Alghathbar, "Cryptanalysis and security enhancement of a 'more efficient & secure dynamic ID-based remote user authentication scheme'," *Computer Communications*, vol. 34, no. 3, pp. 305–309, 2011.

[22] M. Kumar, M. K. Gupta, and S. Kumari, "Cryptanalysis of enhancements of a password authentication scheme over insecure networks," in *Proceedings of the 4th International Conference on Contemporary Computing (IC3 '11)*, vol. 168, pp. 524–532, Noida, India, 2011.

[23] M. K. Khan, S. Kumari, and M. K. Gupta, "Further cryptanalysis of 'a remote authentication scheme using mobile device'," in *Proceedings of the 4th International Conference on Computational Aspects of Social Networks (CASoN '12)*, pp. 234–237, Sao Carlos, Brazil, November 2012.

[24] S. Kumari, M. K. Gupta, M. K. Khan, and F. T. B. Muhaya, "Cryptanalysis of 'an improved timestamp-based remote user authentication scheme'," in *Proceedings of the International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE '12)*, pp. 1439–1442, Chengdu, China, June 2012.

[25] S. Kumari, M. K. Khan, and R. Kumar, "Cryptanalysis and improvement of 'a privacy enhanced scheme for telecare medical information systems'," *Journal of Medical Systems*, vol. 37, no. 4, article 9952, 2013.

[26] A. K. Das, "Analysis and improvement on an efficient biometric-based remote user authentication scheme using smart cards," *IET Information Security*, vol. 5, no. 3, pp. 145–151, 2011.

[27] Y. An, "Security analysis and enhancements of an effective biometric-based remote user authentication scheme using smart cards," *Journal of Biomedicine and Biotechnology*, vol. 2012, Article ID 519723, 6 pages, 2012.

[28] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO' 99*, pp. 388–397, Springer, Berlin, Germany, 1999.

[29] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Transactions on Computers*, vol. 51, no. 5, pp. 541–552, 2002.