An enhanced biometrics-based remote user authentication scheme using mobile devices

R. Madhusudhan*
Department of Mathematics,
Indian Institute of Technology Roorkee,
Roorkee 247667, India
and
Department of M.A.C.S.,
National Institute of Technology Karnataka,
Surathkal 575025, India
E-mail: madhu_nitks@yahoo.com
*Corresponding author

R.C. Mittal
Department of Mathematics,
Indian Institute of Technology Roorkee,
Roorkee 247667, India
E-mail: rcmmmfma@iitr.ernet.in

Abstract: Remote user authentication is a mechanism, in which the remote server verifies the legitimacy of a user over an insecure communication channel. Recently, Wang and Li proposed a fingerprint-based remote user authentication scheme using mobile devices. We demonstrate that their scheme is vulnerable and susceptible to many attacks and has some practical pitfalls. To solve these problems, we propose an efficient fingerprint-based remote user authentication scheme using mobile devices. Proposed scheme not only overcomes all the drawbacks and problems of Wang and Li’s scheme, but also provides secure and user-friendly fingerprint-based remote user authentication over insecure network.

Keywords: smart card; authentication; password; mobile devices; biometrics.


Biographical notes: R. Madhusudhan received his MTech degree in 2003 from NITK Surathkal, Karnataka (A Deemed University). He is currently an Associate Professor at the Mathematical and Computational Sciences Department of NITK Surathkal, India. He is pursuing his PhD at the Department of Mathematics, IIT Roorkee under the supervision of Professor R.C. Mittal. He is a life member of Computer Society of India and Indian Society for Technical Education. His current research includes network security, remote user authentication and mobile computing.
1 Introduction

The internet has become an integral part of everyday life. With rapid development of the internet technology, we are able to access any service from any place and at any time. Remote user authentication is a mechanism to authenticate remote users over insecure communication network. User authentication is a central component of any security infrastructure. Other security measures depend upon verifying the identity of the sender and receiver of information. Authorisation grants privileges based upon identity. Audit trails would not provide accountability without authentication. If we cannot reliably differentiate an authorised entity from an unauthorised entity, confidentiality and integrity are broken. To access resources at remote servers, users should have proper access rights. One of the simplest and most convenient security mechanisms is the use of a password authentication scheme. Examples of password authentication applications include remote login systems, automated teller machines (ATMs), personal digital assistants (PDAs) and database management systems, etc. To access these resources, each user should have an identifier (ID) and a password (PW). The ID and PW are maintained by the remote server. When a user wants to login to a remote server, he has to submit his ID and PW to the server. On receiving the login message, the remote server checks to see if it can identify the login message in the PW (verification) table. The PW table, also maintained by the server, covers all users’ IDs and PWs. If the submitted ID and PW match the corresponding pair stored in the server’s PW table, the user will be granted to access the server. Two problems are found in this traditional mechanism. One is that the revelation of the PWs can be seen by the administrator of the server, because, the PW table is in plain-text format. The other problem is that an intruder can impersonate a legal user by stealing the user’s ID and PW from the PW table. To make things worse, the current internet is vulnerable to various attacks such as denial of service (dos) attacks, forgery attacks, parallel session attacks, PW guessing attacks, replay attacks, and stolen-verifier attacks, etc., therefore, many PW authentication schemes without storing verifiers in the server have been proposed, and each has its pros and cons.

Due to their efficiency and one-way property, one-way hash functions have been used as the basis on which more and more cryptosystems including PW authentication systems are developed.

Recently, lots of PW authentication schemes based on one-way hash function have been proposed, because the computation cost is lower than those of RSA-based and ElGamal-based PW authentication schemes. Due to the low cost, portability, efficiency and the cryptographic capacity, smart cards have been widely adopted in remote user authentication schemes. Traditional remote user authentication methods mainly employ the possession of a token (smart cards, cell phones, personal digital assistant (PDA), and
An enhanced biometrics-based remote user authentication scheme

notebook computers, etc.) and/or the knowledge of a secret (PW, PIN, etc.) in order to establish the identity of an individual. To spread this technology, commercial companies are providing remote authentication of mobile users to access their resources remotely (e.g., online banking and mobile commerce). A token, however, can be lost, stolen, misplaced, or willingly given to an unauthorised user; and a secret can be forgotten, guessed, or unwillingly or willingly disclosed to an unauthorised user. The science of biometrics has emerged as a powerful tool for remote user authentication systems. Since it is based on the physiological and behavioural characteristics of an individual, biometrics does not suffer from disadvantages found in traditional methods. Also, biometrics and tokens have the potential to be a very useful combination. First, the security and convenience of biometrics allow for the implementation of high-security applications regarding tokens. Second, tokens represent a secure and portable way of storing biometric templates, which would otherwise need to be stored in a central database. Among the various biometric technological tools in use today, fingerprint recognition seems to be particularly suitable for token systems. Most recently, Li and Hwang (2010) proposed a remote user authentication scheme based on biometrics verification, smart card, one-way hash function and nonce, which is more efficient compared with other schemes.

In this paper, we find that Wang and Li’s scheme fails to provide efficient authentication and session key agreement. It is also vulnerable to replay attack, denial of service attack and impersonation attack. To eliminate these weaknesses, we propose a new fingerprint-based remote user authentication scheme that removes these weaknesses and supports session key agreement.

This paper is organised as follows. In Section 2, we briefly review related work in this area. Section 3 reviews a scheme proposed by Wang and Li. Its security weaknesses are given in Section 4. Proposed scheme is presented in Section 5. Section 6 makes the security analysis of the proposed scheme. Performance comparison of the proposed scheme with related schemes is presented in Section 7. Finally, Section 8 concludes the paper.

2 Related work

In Lamport (1981) proposed a remote PW authentication scheme using a PW table to achieve user authentication. However, one of the weaknesses of Lamport’s scheme is that a verification table should be stored in the remote system in order to verify the legitimacy of a user. If an intruder can somehow break into the server, the contents of the verification table can be easily modified. In Hwang and Li (2000) pointed out that Lamport’s scheme suffers from the risk of a modified PW table and the cost of protecting and maintaining the PW table. Therefore, they proposed a new user authentication scheme using smart cards to eliminate the risk and cost, which was based on ElGamal (1985) public key encryption method. Theirs scheme can withstand replaying attacks and also authenticate users without maintaining a PW table.

Until now, there have been ample of remote user authentication schemes (Wang and Chang, 1996; Yang and Shieh, 1999; Sun, 2000; Lee et al., 2002; Shen et al., 2003; Tsai et al., 2006) based on smart card published in the literatures and each published scheme has its own merits and demerits. In these schemes, the smart card takes as input a PW from the user, creates a login message from the given PW, and sends the message to a
remote server, which then checks the validity of the login message before allowing access to any services or resources. This way the administrative overhead of the authentication server is reduced, and the user only needs to remember his PW. Besides creating and sending login messages, smart cards may also support mutual authentication. Although there are many smart card oriented schemes proposed to authenticate a legitimate user, none of them can solve all possible problems. In these schemes, an important problem is the lost smart card problem. If an intruder picks up a smart card lost by some legal user, then he/she can impersonate the user to login the system by guessing the PW of the card owner. On the contrary, the distinguishing feature of biometrics, such as fingerprints, faces and irises, provides the opportunity for a more reliable and automated method of identity verification based on measurable physiological or behavioural characteristics. These biometric characteristics are usually universal, unique and cannot be duplicated, lost or forgotten. In a generic biometric authentication system, feature information is extracted from the scanned biometric data and compared with the pre-stored template. The similarity between the inputted biometric and the template are computed taking into consideration the rotation, transformation and scaling. If the similarity is higher than a certain pre-defined threshold, the biometrics are claimed to be match, otherwise the biometrics are determined not to match (Ratha et al., 2001).

Recently, biometric-based authentication systems are becoming very popular because of their ability to differentiate between a legitimate user and imposter by verifying their physiological or behavioural characteristics (Jain and Uludag, 2003). Most commonly used biometric techniques are face, fingerprint, iris, voice, and palm print, etc., but fingerprint-based biometric authentication systems have attracted more attention and mostly deployed Jain et al. (1997). On the basis of excellent properties and easiness of fingerprint-based biometric authentication systems, recently Lee et al. (2002) proposed a remote user authentication scheme using fingerprint and smart cards. Their scheme is an extension of Hwang and Li’s (2000) scheme, which is based on ElGamal public key cryptosystem and does not require PW tables on the remote server. In contrast with Hwang and Li’s scheme, Lee et al. utilised two secret keys to strengthen the security and claimed that introducing two secret keys can protect the system from the impersonation attack and forgery attack. Unfortunately, Hsieh et al. (2003) identified that Lee et al.’s scheme suffers from the impersonation attack. They mentioned that Lee et al.’s scheme is insecure even using the fingerprint-based technique. More recently, Lin and Lai (2004) and Ku et al. (2005) also pointed out that Lee et al.’s scheme is susceptible to simple forgery and impersonation attacks, in which any intruder can impersonate a legitimate user without knowing the PWs and being verified by the fingerprint verification. Ku et al. identified that Lee et al.’s scheme is not repairable and has weakness; even they utilised two secret keys and fingerprint-biometric (Ku et al., 2005). Khan and Zhang (2006) also analysed Lee et al. and pointed out that their scheme only performs unilateral authentication (only client authentication) and user has no information about the authenticity of the remote server. Thus, their scheme is vulnerable to the server spoofing attack. They also identified that in their scheme, users can not choose and change their PWs, and PWs are computed by the servers and assigned to the users, which are large pseudorandom numbers and difficult to remember. Furthermore, cost of maintaining and protecting two secret keys in Lee et al.’s scheme is also high, which is not an ideal solution. Kim et al. (2003) proposed an ID-based PW authentication scheme using smart cards and fingerprints in 2003 and it has also been shown to be insecure by Scott (2004). Khan and Zhang (2007) pointed out that Lin and Lai’s (2004) biometrics-based
An enhanced biometrics-based remote user authentication scheme

authentication scheme is susceptible to the server spoofing attacks. In Fan and Lin (2009) proposed a provably secure remote truly three factor authentication scheme with privacy protection on biometrics. Very recently, Khan et al. (2008) proposed an efficient and practical chaotic hash-based fingerprint biometric remote user authentication scheme using mobile devices (e.g., cell phones and PDAs). However, Yoon and Yoo (2007) demonstrated that Khan et al.’s scheme was vulnerable to a privileged insider’s attack and an impersonation attack by using lost or stolen mobile devices. To isolate such problems, Yoon and Yoo proposed an improvement to Khan et al.’s scheme. The two authentication schemes mentioned above use timestamps to withstand the attack of replaying previously intercepted messages. Thus, the authentication schemes require the system clock synchronisation. Otherwise, the authentication scheme will not work properly. Since network environment and transmission delay is unpredictable (Gong, 1992), a potential replay attack exists in all schemes that employ the concept of timestamps. Li and Hwang (2010) proposed a remote user authentication scheme based on biometrics verification, smart card, one-way hash function and nonce. However, quite recently, Li et al. (2011) showed that Li and Hwang’s scheme does not provide proper authentication, thus it cannot resist the man-in-the-middle attack, and has a problem in biometrics authentication method. Li et al. proposed an improved biometrics-based remote user authentication scheme in order to remove the weaknesses existing in Li and Hwang’s scheme. Very recently Jeon et al. (2011) identified the weaknesses in Li and Hwang (2010) and Li et al. (2011). Wang and Li (2009) also proposed a remote user authentication scheme based on biometrics verification, smart card, one-way hash function and nonce, which is more efficient compared with other schemes. However, we show in Section 4, some weaknesses of Wang and Li’s scheme and propose a scheme in Section 5. This work is an extension of our earlier work in Madhusudhan and Mittal (2011).

3 Review of Wang and Li’s scheme

In Wang and Li (2009) proposed a fingerprint-based remote user authentication scheme. Their scheme is composed of four phases namely registration phase, login phase, verification phase and PW change phase. Before describing the different phases, the notations used in this paper are shown in Table 1.

Table 1 Notations

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>User</td>
</tr>
<tr>
<td>S</td>
<td>Remote server</td>
</tr>
<tr>
<td>ID</td>
<td>User’s ID</td>
</tr>
<tr>
<td>PW</td>
<td>User’s password</td>
</tr>
<tr>
<td>B</td>
<td>Biometric template of U</td>
</tr>
<tr>
<td>h()</td>
<td>One-way hash function</td>
</tr>
<tr>
<td>⊕</td>
<td>Bitwise XOR computation</td>
</tr>
<tr>
<td>x</td>
<td>A secret number maintained by S and stored in user’s mobile</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>SK</td>
<td>Session key</td>
</tr>
</tbody>
</table>
3.1 Registration phase

In the registration phase, the user $U_i$ imprints his/her fingerprint impression $B_i$ at the sensor and chooses his/her ID$_i$, password PW$_i$, and a large random number $r$. $U_i$ submits $\{\text{ID}_i, \text{PW}_i \oplus r, B_i\}$ to the server $S$ through a secure channel. The server $S$ performs the following operations.

Step R1 Computes $A_i = h(\text{ID}_i \oplus x)$ and $X_i = h(A_i)$.
Step R2 Computes $Y_i = h(\text{PW}_i \oplus r \oplus B_i)$, where $B_i$ is the extracted fingerprint template of $U_i$.
Step R3 Computes $R_i = A_i \oplus Y_i$.
Step R4 Store $\{\text{ID}_i, X_i, R_i, Y_i, B_i, h(.)\}$ on the user mobile device and sends it to the user via a secure channel.
Step R5 User $U_i$ enters $r$ into his/her mobile device.

3.2 Login phase

If $U_i$ wants to login into the system, he/she opens the login application software, enters ID$_i$ and PW$_i$, and imprints a fingerprint biometric $B_i$ at the sensor. If $U_i$ is successfully verified by his/her fingerprint biometric, the mobile device performs further the following operations; otherwise, it terminates the operation.

Step L1 Computes $K_i = R_i \oplus h(\text{PW}_i \oplus r \oplus B_i)$, and verifies whether $h(K_i)$ is equal to the stored $X_i$ or not. If the two values are equal, the user’s mobile device performs further operations; otherwise, it terminates the operation.
Step L2 Computes $C_1 = R_i \oplus Y_i$.
Step L3 Computes $C_2 = C_1 \oplus N_1$, where $N_1$ is a random nonce, which is randomly selected by user $U_i$.
Step L4 At the end of the login phase, user $U_i$ sends the login message $\{\text{ID}_i, C_2\}$ to the server $S$ over an insecure network.

3.3 Authentication phase

In the authentication phase, the remote server $S$ receives the login message from the user and performs the following operations.

Step A1 Checks the validity of ID$. If ID$ is invalid, it rejects the login request.
Step A2 Computes $C_1' = h(\text{ID}_i \oplus x), \ N_1' = C_2 \oplus C_1', \ C_3 = C_1' \oplus N_2$ and $V_1 = h(C_2 \oplus N_1')$, where $N_2$ is a random nonce selected by the server.
Step A3 Sends the message $\{V_1, C_3\}$ to $U_i$.

Upon receiving the messages $\{V_1, C_3\}$, the user $U_i$ performs the following operations.
Step A4 Computes $V_{hCN} = \oplus$ and checks whether $V_{hCN}'$ is equal to the received $V_1$ or not. If they are equal, the user $U_i$ believes that the remote server $S$ is authenticated; otherwise $U_i$ terminates the operation.

Step A5 Computes $N_2' = C_3 \oplus C_1$ and $V_2 = h(C_3 \oplus N_2')$ in sequence.

Step A6 Sends the message $\{V_2\}$ to the remote server $S$.

Upon receiving the message $\{V_2\}$, the remote server $S$ performs the following operation.

Step A7 Computes $V_2' = h(C_3 \oplus N_2)$ and checks whether $V_2'$ is equal to the received $V_2$ or not. If they are equal, the remote server $S$ believes that $U_i$ is authenticated; otherwise, the remote server $S$ terminates the operation.

3.4 PW-change phase

Whenever user $U_i$ wants to change or update his/her old password $PW_i$ to the new password $PW_{new}$, he/she opens the login application on his/her mobile device and enters his/her ID and $PW_i$, and also imprints a fingerprint biometric at the sensor. If $U_i$ is successfully verified, the mobile device performs further the following operations, without any help from the remote server $S$.

Step C1 Computes $K_i = R_i \oplus h(PW_i \oplus r \oplus B_i) = h(ID_i \oplus x)$.

Step C2 Verifies whether $h(K_i)$ is equal to the stored $X_i$ or not. If the two values are equal, the mobile device performs further operations; otherwise, it terminates the operation.

Step C3 Computes $X'_i = K_i \oplus h(PW_{new} \oplus r \oplus B_i)$.

Step C4 Replaces the old value of $X_i$ with the newly computed value $X'_i$. The new PW is successfully updated, and this phase is terminated.

4 Cryptanalysis of Wang and Li’s scheme

In this section, we will demonstrate that Wang and Li’s scheme fails to provide efficient authentication and session key agreement. We will also prove that the scheme is vulnerable to replay attack, denial of service attack and impersonation attack.

4.1 Authentication is not efficient

In login phase of Wang and Li’s scheme, it is noted that user $U_i$, first enters his/her personal biometrics on the specific device to verify whether his/her biometrics passes or not. If this verification passes, then $U_i$ enters his/her password $PW_i$ and identity $ID_i$. It checks the entered PW is correct or not by checking $h(K_i)$ is equal to the stored $X_i$ Where $K_i = R_i \oplus h(PW_i \oplus r \oplus B_i)$. If it is true, then it computes $C_1$ as $C_1 = R_i \oplus Y_i$, which is not required, since $K_i$ already contains the value it has calculated in $C_1$. At the server side no
proper validation of the client. If the adversary modifies the value of $ID_i$ to $ID_a$, still the server calculates all the values as follows:

$$K_a = h(ID_b \oplus x),$$

$$N'_a = h(C_2 \oplus K_a),$$

$$C_{a3} = K_a \oplus N_2$$

$$V_a = h(C_2 \oplus N'_a)$$

And sends $\{V_a, C_{a3}\}$ to the user. The user $U_i$ checks whether $V_a = h(C_2 \oplus N_1)$, which is obviously not true. It concludes that the server is not authentic, even though the server is authentic. Therefore, the authentication is not efficient.

4.2 Vulnerable to denial of service attack

At step C3 of PW change phase, mobile device computes $X'_i = K_i \oplus h(PW_i \oplus r \oplus B_i)$ and replaces the value $X_i$ with $X'_i$. The $X_i$ value previously stored in the mobile device was $X_i = h(A_i)$, where $A_i = h(ID_i \oplus x)$. If the PW change phase replaces the value of $X_i$ with $X'_i = K_i \oplus h(PW_i \oplus r \oplus B_i)$, then the legal user cannot get authenticated. He/she is denied service forever. Due to this weakness the legal user cannot be able to login, after he/she changes the PW. Therefore, the scheme is vulnerable to denial of service attack.

4.3 Vulnerable to replay attack

An adversary $U_a$ without knowing users’ PWs wants to masquerade as a legal user $U_i$ by creating a valid login message from the eavesdropped communication between $S$ and $U_i$. The adversary applies the following steps and can successfully make a valid login request to masquerades as the legal user $U_i$. Intercepts the login message $\{ID_a, C\}$, which is sent by the user $U_i$ to $S$. The response message $\{V_1, C_3\}$, which is sent by $S$ to $U_i$ is also intercepted. Once $U_a$ intercepts these messages, he/she masquerades as the legal user $U_i$ to start a new session with $S$ by sending $V_2$ to $S$, where $V_2 = V_1$. Upon receiving the $V_2$, the server verifies by computing $V_2 = h(C_3 \oplus N_2)$, since both are equal, adversary $U_a$ masquerades as a legal user $U_i$. Therefore, the scheme is vulnerable to replay attack. Similarly, we can prove that the scheme is vulnerable to reflection attack and parallel session attack.

4.4 No session key agreement

Wang and Li’s scheme does not have provision to provide the session key and its agreement between the $U_i$ and the remote server $S$. It is pertinent that after the successful authentication process, both parties will communicate some secret messages, which should be encrypted to provide the confidentiality and secrecy of transmitted data, e.g., online money transfer or secure-order placement. For providing this confidentiality, a shared-session key is required, but Wang and Li’s scheme, does not provide the session key.
4.5 Vulnerable to impersonation attack by using lost or stolen mobile device

Wang and Li’s scheme is vulnerable to impersonation attacks using lost or stolen mobile device. Namely, a user can be authenticated to a remote system even if he/she does not have the valid password $PWi$. Precisely, if an attacker gets a user’s mobile device and extracts secure values \{ID, Xi, Yi, Bi, h(.), r\}, then he/she can simply be authenticated without the user’s PW as follows. He/she computes $C_1 = R_i \oplus Y_i$ and $C_2 = C_1 \oplus N_i$, where $N_i$ is a random nonce, which is randomly selected by attacker. Attacker sends the login message \{ID, C_2\} to the server $S$ over an insecure network. Server $S$ authenticates the attacker as the legal user. So, the scheme is vulnerable to impersonation attack by using lost or stolen mobile device.

5 The proposed scheme

In this section, we will propose an improved user friendly remote authentication scheme with mobile device. The scheme is also separated into four phases: registration phase, login phase, authentication phase and PW change phase.

5.1 Registration phase

In the registration phase, the user $Ui$ chooses his/her ID, password $PW_i$ and a large random number $r$. The user $Ui$ interactively submits \{ID, PW_i \oplus r, Bi\} to the registration centre through a secure channel, where $\oplus$ is a bit-wise exclusive-or operation. User $Ui$ also imprints his/her fingerprint impression at the sensor, and then the registration system performs the following operations.

Step R1 Computes $A_i = h(ID_i \oplus x)$ and $X_i = h(A_i)$.

Step R2 Computes $R_i = A_i \oplus h(PW_i \oplus r \oplus B_i)$, where $B_i$ is the extracted fingerprint template of $Ui$.

Step R3 Remote server $S$ personalises the secure information \{ID, $X_i, R_i, B_i, h(.)\} and saves it into the system of $Ui$.

Step R4 User $Ui$ enters $r$ into his/her mobile device.

5.2 Login phase

If $Ui$ wants to login the system, he/she opens the login application software, enters ID, and PW, and imprints a fingerprint biometric at the sensor. If $Ui$ is successfully verified by his/her fingerprint biometric, the mobile device performs further the following operations; otherwise it terminates the operation.

Step L1 Computes $K_i = R_i \oplus h(PW_i \oplus r \oplus B_i)$, and verifies whether $h(K_i) is equal to the stored $X_i or not. If the two values are equal, the user’s mobile device performs further operations; otherwise, it terminates the operation.

Step L2 Computes $C_1 = K_i \oplus N_i$, where $N_i$ is a random nonce, which is randomly selected by user $Ui$. 
Step L3 At the end of the login phase, the user $U_i$ sends the login message $\{ID_i, C_1\}$ to the remote server $S$ over an insecure network.

5.3 Authentication phase

In the authentication phase, the remote server $S$ receives the login message from the user and performs the following operations.

Step A1 Checks the validity of $ID_i$, if $ID_i$ is invalid, it rejects the login request.

Step A2 Computes $K'_i = h(ID_i \oplus x)$ and $N'_i = h(C_1 \oplus K'_i)$.

Step A3 Checks whether $C_1 = ?(K'_i \oplus N'_i)$, if it holds, continue, otherwise, rejects the login attempt.

Step A4 Computes $C_2 = K'_i \oplus N_2$, where $N_2$ is a random nonce selected by the server.

Step A5 Sends the message $\{C_2\}$ to $U_i$.

Upon receiving the message $\{C_2\}$, the user $U_i$ performs the following operations.

Step A6 Computes $N'_2 = C_2 \oplus K_i$ and checks whether $C_2 = ?(K_i \oplus N'_2)$, if they are equal, the user $U_i$ believes that the remote server $S$ is authenticated; otherwise $U_i$ terminates the operation.

Step A7 Computes $SK = h(K_i || N_i || N'_2)$ and $C_3 = SK \oplus h(N_i \oplus N'_2)$.

Step A8 Sends the message $\{C_3\}$ to the remote server $S$.

Upon receiving the message $\{C_3\}$, the remote server $S$ performs the following operation.

Step A9 Computes $SK = h(K'_i || N'_i || N_2)$ and checks whether $C_3 = ?SK \oplus h(N'_i || N_2)$. If they are equal, the remote server $S$ believes that $U_i$ is authenticated; otherwise, the remote server $S$ terminates the operation.

5.4 PW-change phase

Whenever the user $U_i$ wants to change or update his/her old password $PW_i$ to the new password $PW_i^{new}$, he/she opens the login application on his/her mobile device and enters his/her $ID_i$ and $PW_i$, and also imprints his/her fingerprint biometric at the sensor. If $U_i$ is successfully verified, the mobile device performs further the following operations without any help from the remote server $S$.

Step C1 Computes $K_i = R_i \oplus h(PW_i \oplus r \oplus B_i) = h(ID_i \oplus x)$.

Step C2 Verifies whether $h(K_i)$ is equal to the stored $X_i$ or not. If the two values are equal, the mobile device performs further operations; otherwise, it terminates the operation.

Step C3 Computes $R'_i = K_i \oplus h(PW_i^{new} \oplus r \oplus B_i)$. 
An enhanced biometrics-based remote user authentication scheme

343

Step C4 Replaces the old value of \( R_i \) with the new \( R'_i \). The new PW is successfully updated, and this phase is terminated.

6 Security analysis of the proposed scheme

This section describes the security analysis of the proposed scheme.

6.1 Authentication is efficient

In the login phase of Wang and Li’s scheme, it is noted that user \( U_i \) first enters his/her personal biometrics on the specific device to verify whether his/her biometrics passes or not. If this verification passes, then \( U_i \) enters his/her password \( PW_i \) and identity \( ID_i \). It checks the entered PW is correct or not by checking \( h(K_i) \) is equal to the stored \( X_i \), where \( K_i = R_i \oplus h(PW_i \oplus r \oplus B_i) \). If it is true, then it computes \( C_1 = K_i \oplus N_i \), and sends \( \{ID_i, C_1\} \) to the server. It has calculated the value of \( C_1 \) with the already computed value of \( K_i \). It reduces the computation burden of the mobile device. At the server side proper validation of the client is done as follows.

Server computes \( K'_i = h(ID_i \oplus x) \) and \( N'_i = (C_i \oplus K'_i) \). S checks whether \( C_i = h(K'_i \oplus N'_i) \), if it holds, continue; otherwise, rejects the login attempt. If the adversary modifies the value of \( ID \) to \( ID_a \), the server will reject the login request as follows.

\( S \) computes \( K_a = h(ID_a \oplus x) \), \( N'_a = h(C_i \oplus K_a) \), \( S \) checks whether \( C_i = h(K_a \oplus N'_a) \). Since this condition is obviously false, the server rightly takes the conclusion that the user is not authentic. Therefore, the authentication is efficient and user friendly.

6.2 Resistance to denial of service attack

Our scheme rightly modifies the value of \( R_i \) rather than \( X_i \) in the PW change phase. If \( U_i \) is successfully verified, the mobile device performs the following operations without any help from the remote server \( S \). Computes \( K_i = R_i \oplus h(PW_i \oplus r \oplus B_i) = h(ID_i \oplus x) \). Verifies whether \( h(K_i) \) is equal to the stored \( X_i \) or not. If the two values are equal, the mobile device performs further the following operations; otherwise, it terminates the PW change phase. Computes \( R' = R_i \oplus h(PW_i^{new} \oplus r \oplus B_i) \) and replaces the old value of \( R_i \) with new \( R'_i \). The new PW is successfully updated. Thus, it eliminates denial of service attack in the Wang and Li’s scheme.

6.3 Resistance to replay attack

An adversary \( U_a \) without knowing users’ PWs wants to masquerade as a legal user \( U_i \) by creating a valid user message from the eavesdropped communication messages between \( S \) and \( U_i \). The adversary applies the following steps. Intercepts the login message \( \{ID_i, C_2\} \), which is sent by the user \( U_i \) to \( S \). The response message \( \{C_2\} \), which is sent by \( S \) to \( U_i \) is also intercepted. Even though, \( U_a \) intercepts these messages, he/she cannot be able to
compute $C_3 = SK \oplus h(N_1 \parallel N_2')$, where $SK = h(K'_r \parallel N'_1 \parallel N_2)$, because the adversary does not have the information about $N_1$, $N_2$ and $x$ to compute $C_3$. Therefore, $U_a$ cannot compute the session key agreement during the authentication phase. At step A7 of the authentication phase $U_i$ computes the session key as $SK = h(K'_r \parallel N'_1 \parallel N_2')$. Similarly, at step A9 of the authentication phase $S$ computes $SK = h(K'_r \parallel N'_1 \parallel N_2')$. Thus, a session key is shared between $U_i$ and $S$. Hence, $U_i$ and $S$ can use $SK$ to securely perform encryption and decryption of subsequent messages.

6.4 Session key agreement

The proposed scheme provides session key agreement during the authentication phase. At step A7 of the authentication phase $U_i$ computes the session key as $SK = h(K'_r \parallel N'_1 \parallel N_2')$. Similarly, at step A9 of the authentication phase $S$ computes $SK = h(K'_r \parallel N'_1 \parallel N_2')$. Thus, a session key is shared between $U_i$ and $S$. Hence, $U_i$ and $S$ can use $SK$ to securely perform encryption and decryption of subsequent messages.

6.5 Resistance to impersonation attack by using lost or stolen mobile device

The proposed scheme is not vulnerable to impersonation attacks using lost or stolen mobile device. Namely, a user cannot be authenticated to a remote system even if he/she does not have the valid password $PW_i$. Precisely, if an attacker gets a user’s mobile device and extracts secure values $\{ID_i, X_i, R_i, B_i, h(), r\}$, then he/she cannot simply be authenticated without the user’s PW as follows. He/she needs to compute $C_1 = K'_i \oplus N_i$. Even though the attacker picks up a random number $N_i$, he/she cannot get $K'_i$ since it is not stored in the mobile device. So, the attacker cannot able to send the login message $\{ID_i, C_2\}$ to the server $S$ over an insecure network. Therefore, the proposed scheme is not vulnerable to Impersonation attack by using lost or stolen mobile device.

7 Performance comparison

In this section, we compare the performance of the proposed scheme with related schemes based on security requirements (Madhusudhan and Mittal, 2012), functional requirements and computation cost.

Security requirements comparison of the proposed scheme with related schemes is shown in Table 2. It demonstrates that our scheme is more secure and robust than the other related schemes and achieves more security features, which were not considered in their scheme and are essentially required in implementing a practical and universal remote user authentication scheme using smart cards or mobile devices. It is clear from the Table 2 that our proposed scheme satisfies all the security requirements, other schemes Yoon and Yoo (2007), Khan et al. (2008), Li and Hwang (2010) and Wang and Li (2009) satisfies 5, 4, 5 and 2 security requirements respectively. All the schemes provide resistance to guessing attack. Even though, attacker somehow finds the PW, he/she cannot login into the system, because, proper biometric of the user is not available for the attacker.
Table 2  Security requirements comparison of the proposed scheme with related schemes

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Resistance to impersonation attack</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Resistance to parallel session attack</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Resistance to password guessing attack</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Resistance to replay attack</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Resistance to mobile device loss attack</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Resistance to reflection attack</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Resistance to insider attack</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Functionality requirements comparison of the proposed scheme with related schemes is shown in Table 3. It is clear from the table that the proposed scheme satisfies all the functional requirements, other schemes Yoon and Yoo (2007), Khan et al. (2008), Li and Hwang (2010) and Wang and Li (2009) satisfies 4, 3, 4 and 5 functional requirements respectively. Functional requirements such as no verification table, freely chosen PW and mutual authentication are achieved by all the schemes, but, efficient for wrong PW login, session key agreement and securely update PW are not achieved by all the related schemes except our proposed scheme.

Table 3  Functionality requirements comparison of the proposed scheme with related schemes

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>No verification table</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Freely chosen password</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Securely update password</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>No password reveal</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Mutual authentication</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Session key agreement</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Efficient for wrong password login</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Time synchronisation not needed</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Due to the resources constraints of mobile devices, the authentication scheme must take efficiency evaluation into consideration. In general, the efficiency evaluation usually is done by computing the computational cost. The computation cost of each phase is defined as the total time of various operations executed in that phase. Computational cost comparison of the proposed scheme and several related schemes is shown in Table 4, where $T_h$ represents the execution time of the one-way hash function $h()$, the execution
time of the XOR operation is $T_\oplus$ and $T_\|\|$ represents execution time of the concatenation operation.

### Table 4 Computational cost comparison of the proposed scheme with related schemes

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration</td>
<td>$3T_h + 2T_\oplus$</td>
<td>$2T_h + 3T_\oplus$</td>
<td>$2T_|| + 3T_h + 1T_\oplus$</td>
<td>$3T_h + 5T_\oplus$</td>
<td>$2T_|| + 4T_h + 4T_\oplus$</td>
</tr>
<tr>
<td>Login and Verification</td>
<td>$7T_h + 4T_\oplus$</td>
<td>$6T_h + 8T_\oplus$</td>
<td>$6T_|| + 7T_h + 5T_\oplus$</td>
<td>$7T_h + 13T_\oplus$</td>
<td>$8T_h + 14T_\oplus$</td>
</tr>
<tr>
<td>Password change</td>
<td>$3T_h + 6T_\oplus$</td>
<td>$2T_h + 4T_\oplus$</td>
<td>$2T_|| + 3T_h + 2T_\oplus$</td>
<td>$3T_h + 6T_\oplus$</td>
<td>$3T_h + 5T_\oplus$</td>
</tr>
<tr>
<td>Total cost</td>
<td>$13T_h + 12T_\oplus$</td>
<td>$10T_h + 15T_\oplus$</td>
<td>$10T_|| + 13T_h + 8T_\oplus$</td>
<td>$13T_h + 24T_\oplus$</td>
<td>$2T_|| + 15T_h + 23T_\oplus$</td>
</tr>
</tbody>
</table>

It can be seen from the table that the proposed scheme needs two concatenation operations and two more hashing operations than Wang and Li’s scheme. This is because our scheme provides efficient authentication, session key computation and it is vulnerable to different security attacks, which require more hashing operations to enhance the security of authentication system. Hence, the computational cost of the proposed scheme is almost same as Wang and Li’s scheme with several enhanced security features, which are indispensable for implementing a reliable and trustworthy remote user authentication system.

### 8 Conclusions

In this paper, we have presented cryptanalysis and weaknesses of Wang and Li’s fingerprint-based remote user authentication scheme. Firstly, we showed that Wang and Li’s scheme fails to provide efficient authentication, correct PW change phase and no provision for session key agreement. It is also vulnerable to replay attack, denial of service attack and impersonation attack. To overcome the identified problems, we have proposed an enhanced fingerprint-based authentication scheme, which eliminates all the identified weaknesses of Wang and Li’s scheme. The proposed scheme provides efficient authentication because, it removes the redundant computation in the mobile device during the login phase. It is user friendly because it provides validation of the client at the server. The PW change phase, which was responsible for DoS attack has been modified to remove DoS attack completely. Similarly, the proposed scheme is carefully designed to remove replay and impersonation attack. In addition, it provides session key $SK = h(K_i \| N_i \| N_i')$, so that both the user and the server can communicate further securely. Therefore, the proposed scheme is more efficient and robust for real-life use.

### Acknowledgements

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions. This research was supported by the Ministry of Human Resources Development (MHRD) India, under the Quality Improvement Programme (QIP) sponsored by the All India Council for Technical Education (AICTE).
References


