Expert Systems with Applications 41 (2014) 1411–1418

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics

Ming-Chin Chuang [a,*], Meng Chang Chen [a,b]

[a] Research Center for Information Technology Innovation, Academia Sinica, 128 Academia Road, Section 2, Nankang, Taipei 115, Taiwan
[b] Institute of Information Science, Academia Sinica, 128 Academia Road, Section 2, Nankang, Taipei 115, Taiwan

ARTICLE INFO

Keywords:
Authentication
Biometrics
Anonymous
Multi-server
Lightweight

ABSTRACT

Password-based remote user authentication schemes are widely investigated, with recent research increasingly combining a user's biometrics with a password to design a remote user authentication scheme that enhances the level of the security. However, these authentication schemes are designed for a single server environment and result in users needing to register many times when they want to access different application servers. To solve this problem, in this paper we propose an anonymous multi-server authenticating key agreement scheme based on trust computing using smart cards, password, and biometrics. Our scheme not only supports multi-server environments but also achieves many security requirements. In addition, our scheme is a lightweight authentication scheme which only uses the nonce and a hash function. From the subsequent analysis, the proposed scheme can be seen to resist several kinds of attacks, and to have more security properties than other comparable schemes.

1. Introduction

In recent years, wireless communications and network technologies have undergone rapid development (Dondi, Bertacchini, Brunelli, Larcher, & Benini, 2008; Gil-Castineira, Gonzalez-Castano, & Franck, 2008; Hwang, Chang, & Yu, 2007; Lazar & Carari, 2008; Liu, Xia, Chen, Rees, & Hu, 2007; Marino, Poza, Dominguez, & Otero, 2009), and many people now use mobile devices (e.g., PDAs, mobile phones, and notebooks) at anytime and from anywhere to access all kinds of application services from the Internet, such as network attached storage (NAS), Web-browsing, VoIP, video conferencing, and multimedia applications.

However, this mobile computing situation calls for an authentication mechanism to protect the valid user from attacks. The smart card based remote user authentication scheme is one of the simplest and most convenient authentication mechanisms for insecure networks. Lamport (1981) first introduced a password authentication scheme for communication through insecure channels, where the server has to maintain a password table. However, this scheme cannot prevent a stolen-verifier attack. Although many later papers (Fan, Chan, & Zhang, 2005; Juang, Chen, & Liaw, 2008; Sun et al., 2009) proposed improved password-based authentication schemes for resisting such attacks, password-based remote user authentication schemes are unfortunately still easily broken by simple dictionary attacks given the password's low entropy value. Therefore, more and more research (Chang & Lin, 2004; Fan & Lin, 2009; Khan & Zhang, 2006; Khan, Zhang, & Wang, 2008; Ku, Chang, & Chiang, 2005; Lee, Ryu, & Yoo, 2002; Li & Hwang, 2010; Lin & Lai, 2004; Mitchell & Tang, 2005; Xu, Zhu, & Feng, 2008) has combined a user's biometrics (e.g., fingerprints, irises, and hand geometry) with a password and a smart card to design a remote user authentication scheme that enhances the level of the security (i.e., a secret key that has a value of high entropy Fan, 2009). While Lee et al. (2002) put forward a fingerprint-based remote user authentication scheme using smart cards in 2002, a number of studies (Chang & Lin, 2004; Ku et al., 2005; Lin & Lai, 2004) thereafter pointed out that this scheme cannot resist masquerade attacks and server spoofing attacks. Lin and Lai (2004) thus combined password and fingerprint minutiae templates into super passwords and provided an off-line password change scheme, but Mitchell and Tang (2005) observed that the process of the password change is vulnerable because the smart card did not have enough information to check the correctness of the old passwords. Fan and Lin (2009) then suggested a three-factor authentication scheme which combines biometrics with a password and smart card to provide high-security remote authentication, and they proved the security of their scheme. Khan and Zhang (2006) proposed an improved scheme to enhance the security, but this scheme turned out to be susceptible to a parallel session attack (Khan et al., 2008; Xu et al., 2008), in which an adversary without knowing a legal user's password can impersonate the user by somehow crafting a valid login message from eavesdropped communications between the user and the server. Whereas Li and Hwang's (2010) biometric-based remote user authentication

* Corresponding author. Tel.: +886 2 27883799x1378.
E-mail addresses: speedboy@gmail.com, mcc@iis.sinica.edu.tw (M.C. Chen).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.08.040

scheme using smart cards was efficient, it used biometrics-based schemes (Chang & Lin, 2004; Fan & Lin, 2009; Khan & Zhang, 2006; Khan et al., 2008; Ku et al., 2005; Lee et al., 2002; Li & Hwang, 2010; Lin & Lai, 2004; Mitchell and Tang, 2005; Xu et al., 2008) that only supported a single server environment, which is a limitation insofar as there are many kinds of application servers on the Internet. Fig. 1 shows that a user accesses multiple application servers at the same time. If the designed authentication scheme does not consider the multi-server environment, the user performs the registration procedure many times and results in a high overhead at the registration center (RC) and the network. Some research (Chang & Lee, 2004; Juang, 2004; Liao & Wang, 2009; Tsai, 2008) has supported multi-server environments but since their schemes were only based on smart cards and passwords. The authentication system was insecure when both the user's smart card and password were stolen; moreover, the schemes (Chang & Lee, 2004; Juang, 2004; Tsai, 2008) did not provide anonymous authentication. More recently, Yang and Yang (2010) and Yoon and Yoo (2010) introduced biometric-based multi-server authentication schemes, but they still did not consider the user anonymity. Further, Yang's scheme (Yang & Yang, 2010) needs to perform exponential operations that entails high computational cost, while Yoon et al.'s scheme (Yoon & Yoo, 2010) was demonstrated by He (2011) to be vulnerable to privileged insider attacks, masquerade attacks and loss of smart card attacks.

In this paper, we propose an anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards, password, and biometrics. Our scheme not only is a lightweight authentication scheme which only uses the nonce and a hash function but also satisfies all of the following security properties: anonymity, absence of verification tables, mutual authentication, resistance to forgery attack, absence of clock synchronization problem, resistance to modification attacks, resistance to replay attacks, fast error detection, resistance to off-line guessing attacks, resistance to insider attacks, simple and secure password choice and modification, biometric template protection, and session key agreement.

The remainder of this paper is organized as follows. Section 2 introduces some preliminaries. In Section 3, we describe the proposed scheme in detail, and the analyses of the security, computation costs, and comparisons are presented in Section 4. Consequently, we summarize our conclusions in Section 5.

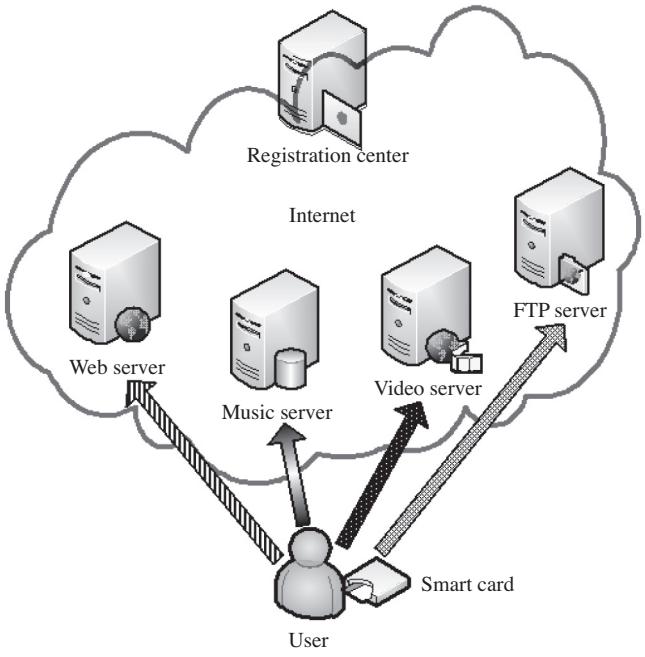

Fig. 1. The user accesses the multiple application servers.

2. Preliminaries

This section describes the common user requirements, the security requirements of the system, the advantage of using biometrics, and the feature of the hash function.

2.1. User requirements

Given that the designed authentication scheme should be user-friendly, the following user requirements need to be considered.

(1) **Simple and secure password choice and modification**: The system allows users to choose and changing their passwords easily and securely. In other words, the user can change the password without the help of any third trusted party after assuring the legality of cardholder.
(2) **Single registration**: The user only needs to register with the registration center once and then can access different application servers. Moreover, the single registration can reduce the overhead of the registration center and the network.
(3) **Anonymity**: The privacy of the user has attracted increasing attention from both industry and academia. Therefore, anonymous authentication involves verifying that a user does not use the real identity to execute the authentication procedure.

2.2. Security requirements

Since a remote user authentication scheme is susceptible to attack from adversaries, our objective is to design a scheme that is robust enough to resist such attacks. Based on related studies (Chang & Lin, 2004; Chang and Lee, 2004; Fan and Lin, 2009; Fan et al., 2005; He, 2011; Juang, 2004; Juang et al., 2008; Khan & Zhang, 2006; Khan et al., 2008; Ku et al., 2005; Lee et al., 2002; Li & Hwang, 2010; Liao & Wang, 2009; Lin & Hwang, 2011; Lin & Lai, 2004; Mitchell and Tang, 2005; Sun et al., 2009; Tsai, 2008; Xu et al., 2008; Yang & Yang, 2010; Yeh, Lo, & Li, 2011; Yoon & Yoo, 2010, 2011), we define the following key requirements for securing authentication.

(1) **Mutual authentication**: A mutual authentication process is required insofar as the server needs to verify that the user is a legal one, and the user needs to ensure that the server is not a forged one.
(2) **Efficiency**: Since the computational capacity of the smart card is limited, the computation and communication costs on smart cards must be as low as possible.
(3) **No verification table**: In most applications, the registration center stores the password table of the user resulting in the stolen-verifier attack, and as such, the designed scheme needs to avoid maintaining the password verification table of the user.
(4) **Integrity**: The message integrity means that data cannot be modified without detection.
(5) **Session key agreement**: After the authentication procedure, the session key is generated between the user and the server to provide a secure communication, and it can achieve forward secrecy.

2.3. Advantage of using biometrics

The weakness of a secret key is its low value of entropy, which can be guessed or cracked in polynomial time. For example, there is no way to prevent the attacker from impersonating the user if both

the user's smart card and password were stolen. For this reason, many schemes (Chang and Lee, 2004; Fan et al., 2005; Juang, 2004; Juang et al., 2008; Lamport, 1981; Liao & Wang, 2009; Sun et al., 2009; Tsai, 2008) guarantee the system security when either the smart card or his password is stolen, but not both. On the other hand, a strong secret key which combines passwords with biometrics and smart cards (called three-factor security) has the value of high entropy (Fan & Lin, 2009), which cannot be guessed in polynomial time. Moreover, the main feature of the biometric is uniqueness in that everyone has a different biometric, and it is difficult for the user's biometric to be stolen because only the user inputs his biometric into his own smart card.

2.4. The feature of hash function

The security property of the proposed scheme is based on a collision-free one-way hash function, such as MD5, SHA, RIPEMD. Generally speaking, the bit size is longer, and the system is more robust. For a one-way hash function $h()$, when the value of x is given, it is easy to compute $h(x)$; however, if the value of $h(x)$ is given, computing x is very difficult or it incurs a high computational cost.

3. Proposed scheme

This section describes the proposed anonymous multi-server authenticated key agreement scheme which involves five procedures: server registration, user registration, login, authentication, and change password. All notations are summarized in Table 1.

3.1. Server registration procedure

The application server sends the RC a join message if it would like to become an authorized server. Then, the RC replies with the key *PSK* to the server via the Internet Key Exchange Protocol version 2 (IKEv2) (Kaufman, 2005). Afterwards, the authorized server uses this key (i.e., *PSK*) to facilitate the user's authentication procedure.

3.2. User registration procedure

Initially, every user needs to perform the user registration procedure with the registration center via a secure channel. Moreover, we assume that the authorized application servers are trusted according to the trust computing (ISO/IEC 11889-1.:2009; ISO/IEC 11889-2.:2009; ISO/IEC 11889-3.:2009; ISO/IEC 11889-4.:2009; Mitchell, 2005) and that the *PSK* cannot be extracted from the RC and application servers. Fig. 2 depicts the user registration procedure. The steps of the procedure are described as follows.

Table 1
Notations.

x	A secret value of the registration center
RC	The registration center
ID_i	The public identification of user i
SID_j	The public identification of server j
PW_i	The password of user i
BIO_i	The biometrics information of user i
AID_i	The anonymous identification of user i
$h()$	A one-way collision-resistant hash function
N_i	A random number
PSK	A secure pre-shared key among authorized application servers and the registration center
$\oplus$	The bitwise XOR operator
$\|\|$	The string concatenation operator
X → Y	User X sends a message to user Y through a secure channel
X → Y	User X sends a message to user Y through a common channel

Step 1: User → RC: The user sends his registration information (i.e., his identification ID_i and $h(PW_i \oplus BIO_i)$) to the RC via a secure channel.
Step 2: After receiving the information, the RC calculates the authentication parameters of the user as follows: $A_i = h(ID_i||x)$, $B_i = h^2(ID_i||x) = h(A_i)$, $C_i = h(PW_i \oplus BIO_i) \oplus B_i$, and $D_i = PSK \oplus A_i$.
Step 3: RC → User: The RC stores these authentication parameters $\{ID_i, B_i, C_i, D_i, h()\}$ in a smart card and delivers the smart card to the user via a secure channel.

Note that, the RC does not obtain the user's verification information (e.g., the password and biometrics information). Therefore, we can prevent the possibility of a stolen-verifier and insider attacks. In addition, the registered user cannot fabricate a valid user successfully when the user obtains these parameters (i.e., ID_i, B_i, C_i, D_i, $h()$). This is because the user does not know the secret value of the RC (i.e., x) and the *PSK*. In this paper, we also maintain the assumption that the biometric matching is exact matching.[1]

3.3. Login procedure

The login procedure is the first check point. The smart card detects an error event immediately if the user is not authorized to gain access (i.e., the user keys in the wrong identification, password, or biometrics information). Fig. 3 shows the steps of the login procedure.

Step 1: The user inserts his smart card into a card reader and keys in his ID_i and PW_i. Then, he scans his biometric information (e.g., fingerprint) BIO_i at the sensor.
Step 2: The smart card checks the ID_i and then verifies whether $h(PW_i \oplus BIO_i) \oplus C_i$ is equal to B_i. If the information is verified, then the smart card generates a nonce N_1, calculates the message M_1 as $h(B_i) \oplus N_1$, computes the alias AID_i as $h(N_1) \oplus ID_i$, and generates the message M_2 as $h(N_1||AID_i||D_i)$, where B_i and C_i are obtained from the user registration procedure.

3.4. Authentication procedure

The smart card sends the server an authentication message after the user finishes the login procedure. Note that the smart card never uses the real identity (i.e., ID_i) to perform the authentication procedure. Fig. 4 shows the steps of the authentication procedure.

Step 1: Smart card → Server: The smart card sends the server an authentication message (i.e., AID_i, M_1, M_2, D_i), where D_i is obtained from the user registration procedure.
Step 2: The server verifies the user: On receipt of the authentication request (i.e., AID_i, M_1, M_2, D_i), the server uses a secure pre-shared key (i.e., *PSK*) to obtain A_i (i.e., $A_i = D_i \oplus PSK$). The server retrieves the value of N_1 (i.e., $N_1 = M_1 \oplus h^2(A_i)$) and then checks whether $h(N_1||AID_i||D_i)$ is equal to M_2. The server rejects this authentication request and terminates this session if the result is not equal. This is because the authentication message has been modified. Next, the server generates a random number N_2 and calculates a session key SK_{ij} as $h(N_1||N_2)$. Finally, the server computes the authentication reply message (i.e., M_3, M_4), where M_3 as $N_2 \oplus h^2(N_1)$ and M_4 as $h(SID_j||N_2)$.

[1] The accuracy of the contemporary biometric recognition is very high. Although we do not discuss the biometric matching issue in this paper, we still respect this issue. We have thus added some related studies (Jea & Govindaraju, 2005; Liu, Zhao, & Zhang, 2011; Meenen, Ashrafi, & Adhami, 2006; Tong, Liu, Huang, & Tang, 2008; Yager and Amin, 2006a, 2006b; Zhu, Yin, & Zhang, 2005) about the biometric matching algorithm for extended reading, and we will discuss the biometric matching issue in our future work.

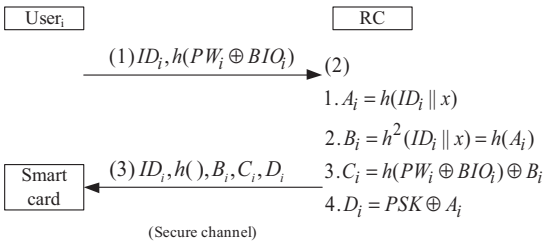

Fig. 2. The user registration procedure.

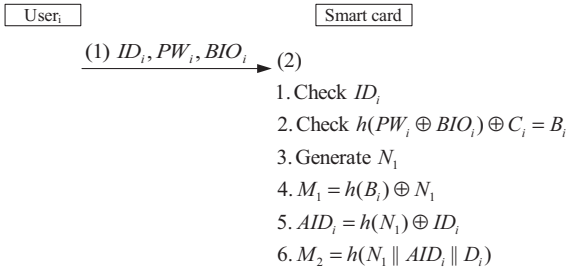

Fig. 3. The login procedure.

Step 3: Server → Smart card: The server sends back the authentication reply message (i.e., SID_j, M_3, M_4) to the smart card.
Step 4: The smart card verifies the server: The smart card computes the value of $h^2(N_1)$, retrieves the random number N_2 (i.e., $N_2 = M_3 \oplus h^2(N_1)$), and checks whether $h(SID_j||N_2)$ is equal to M_4. If the values are equal, the smart card computes the session key (i.e., $SK_{ij} = h(N_1||N_2)$).

Step 5: Smart card → Server: The smart card sends the message (i.e., $SK_{ij} \oplus h(N_2)$) to the server.
Step 6: The server uses the session key SK_{ij} to retrieve the value (i.e., $h(N_2)$), and then it checks this value to prevent an invalid user from executing the replay attack.

3.5. Password change procedure

This procedure is invoked whenever a user wants to change his password. In this procedure, the user can easily change his password without any assistance from the registration center. Fig. 5 illustrates the password change procedure, and the detailed steps are described as follows.

Step 1: A user keys in his ID_i and PW_i, and then he imprints BIO_i at the sensor.
Step 2: The smart card checks the ID_i and verifies whether $h(PW_i \oplus BIO_i) \oplus C_i$ is equal to B_i. If the smart card determines that they are equal, then the user can key in the new password PW_i^*. Otherwise, the smart card rejects the password change request. The smart card computes $C_i^* = C_i \oplus h(PW_i \oplus BIO_i) \oplus h(PW_i^* \oplus BIO_i)$ and replaces C_i by C_i^*. The password has now been changed.

4. Analysis

4.1. Security analysis

We use the same scheme as (Chang and Lee, 2004; Crypto++ Library; Juang, 2004; Khan et al., 2008; Li & Hwang, 2010; Liao & Wang, 2009; Lin & Lai, 2004; Tsai, 2008; Xu et al., 2008; Yang & Yang, 2010; Yoon & Yoo, 2010) to present the security analysis.

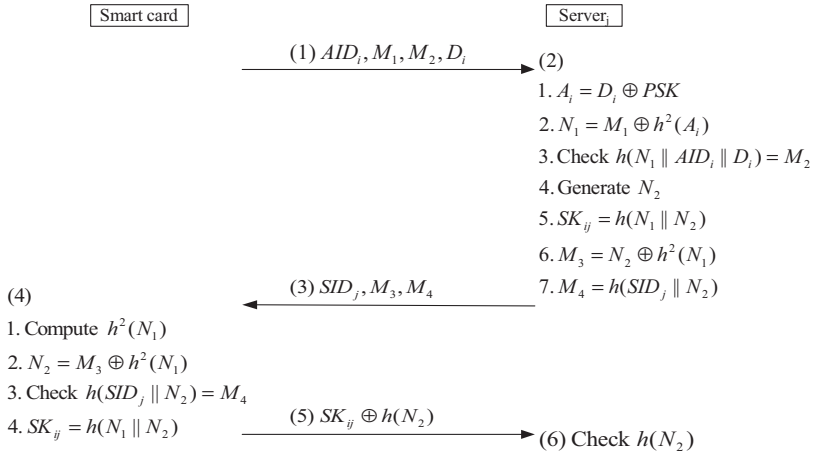

Fig. 4. The authentication procedure.

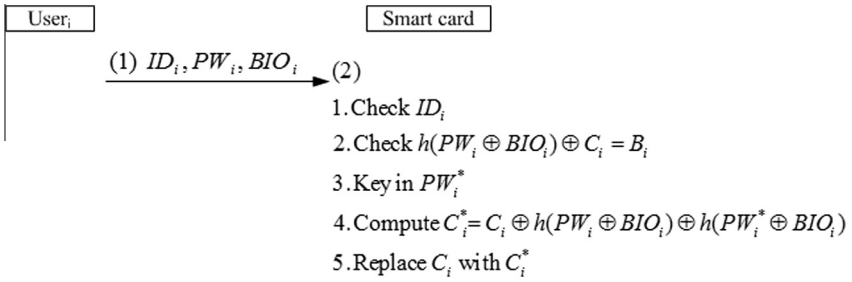

Fig. 5. The password change procedure.

(1) **Anonymity**: Under the proposed scheme, the original identity of a user is always converted into an alias that is based on a random number (i.e., $AID_i = h(N_1) \oplus ID_i$). Therefore, an adversary cannot determine the original identity of the user without knowing the random number N_1 chosen by the smart card. In addition, the unauthorized server cannot decrypt the user's authentication message successfully since it does not obtain the *PSK*. As a result, it cannot retrieve the user's real identity, i.e., the unauthorized server cannot get N_1 because it cannot retrieve A_i from the communication messages. Our anonymity mechanism is a dynamic identification process.

(2) **No verification table**: The registration center and application servers do not store the password table and the biometrics database of the user. Therefore, even if an intruding adversary accesses the database of the RC, he still cannot obtain the authentication information of users.

(3) **Mutual authentication**: A mutual authentication process is required. The server needs to verify that the user is a legal one, and the user needs to ensure that the server is not a forged one. In the authentication procedure, Step 2 shows that the server authenticates the user, and Step 4 shows that the user authenticates the server. If the attacker intercepts the messages and wants to forge a valid server/user, it must generate a valid reply message to the user/server. However, the attacker cannot compute a valid message because he does not know the secure key (i.e., *PSK*) and the random number (i.e., N_1 and N_2).

(4) **Resistance to replay attacks**: To protect the proposed scheme from replay attacks, we add a random number into the message. Hence, if an adversary intercepts the previous authentication message (i.e., AID_i, M_1, M_2, D_i) and tried to impersonate the valid user by immediately replaying the message, the server would obviously reject the request because the invalid random number (i.e., N_1^*) will be detected in Step 2 of the authentication procedure. Moreover, the user also checks the random number which is sent from the server to prevent the replay attack.

(5) **Session key agreement**: We only use one round trip between the user and the server to generate the session key. Then, we can use the session key to encrypt the following packets to ensure the communications are confidential. Moreover, the session key is generated by the random number and a one-way hash function (i.e., $SK_{ij} = h(N_1||N_2)$). Hence, this session key is different in each session, and it is difficult for the adversary to derive the session key from the intercepted messages.

(6) **Not requiring clock synchronization**: In timestamp-based authentication schemes, the clocks of all devices must be synchronized. In our scheme, we provide a nonce-based authentication mechanism, instead of the timestamps that cause serious time synchronization problems.

(7) **Resistance to modification attacks**: An adversary can attempt to modify the authentication message of the user. In this paper, we use a one-way hash function to ensure that information cannot be modified (e.g., M_2 and M_4 in Fig. 4). Therefore, this attack will be detected because an attacker has no way to obtain the value of the random number to generate the legitimate message. If an attacker transmits a modified packet to the server, the packet can be easily identified by checking the hash values.

(8) **Resistance to forgery attacks**: If a valid user attempts to forge another valid user (i.e., AID_i^*), it will authenticate unsuccessfully (i.e., Step 2 of Fig. 4). Even if the user knows his parameters (i.e., ID_i, B_i, C_i, D_i, $h()$) and forges an alias identification (i.e., $AID_i^* = h(N_1) \oplus ID_i^*$), the user cannot figure out the valid authentication parameter (i.e., $D_i^* = PSK \oplus A_i^*$) to pass the authentication. This is because the malicious user does not know the secret key of the RC (i.e., x) and other user's real identity, which will result in the smart card incorrectly computing the value of A_i^* (i.e., $A_i^* = h(ID_i^*||x^*)$). The secret key of the RC (i.e., x) is protected by the one-way hash function $h()$, and thus it is computationally infeasible to derive x from the value $h(ID_i||x)$.

(9) **Resistance to off-line password guessing attacks**: Since PW_i, *PSK*, BIO_i, and x are unknown to the adversary, the system is secure even if the stored information B_i, C_i, and D_i are revealed. The password and the biometrics of the user (i.e., PW_i and BIO_i) are protected by the one-way hash function $h()$, which means that the adversary cannot check whether or not each of his guessed password (i.e., PW_i^*) is correct (i.e., $h(PW_i^* \oplus BIO_i) = C_i \oplus B_i$). Moreover, it is impossible for any two people to have the same biometrics template, such as a fingerprint. Therefore, our scheme can defeat the off-line password guessing attack.

(10) **Fast error detection**: In the login or password change procedures, the smart card detects the error immediately if the attacker keys in the wrong identification, password, or biometrics information (i.e., Step 2 in the login procedure and Step 2 in the password change procedure).

(11) **Simple and secure password choice and modification**: The user can change their password at liberty so that it is easy for the user to remember passwords. Moreover, our password change procedure does not need any assistance from the RC. In addition, any attacker cannot update the password even if he obtains the smart card and the password, which is because the incorrect biometric template BIO_i^* will be detected (i.e., $h(PW_i \oplus BIO_i^*) \oplus C_i$ is not equal to B_i).

(12) **Biometric template protection**: As the user may use his biometrics in other biometric applications, we prevent the smart card from being cracked if it is lost, which can result in the biometrics information of the user being obtained by the adversary. In our scheme, the biometrics information of the user (i.e., BIO_i) is not directly stored into the smart card, but is rather protected by the one-way hash function $h()$ (i.e., Step 1 in Fig. 2), which makes it impossible for the adversary to obtain the user's biometrics information.

(13) **Resistance to insider attacks**: In the registration procedure, the user sends RC a registration message, which is $h(PW_i \oplus BIO_i)$ instead of PW_i and BIO_i. The RC cannot obtain the user's password and biometrics directly. Moreover, it is difficult for the attacker to retrieve the user's information since the hash function has the one-way property. As such, the proposed scheme can resist insider attacks.

(14) **Support multi-server environment**: If the designed authentication scheme does not consider the multi-server environment, the user performs the registration procedure many times, which results in a high overhead at the RC and the network. In our scheme, the user only needs to register with the RC once and then can access multiple different application servers at the same time. As long as the application server is authorized, it can get the key *PSK* and perform the authentication procedure for the registered user. Moreover, the RC does not need to participate in the authentication procedure of the user. Therefore, our scheme belongs to the multi-server environment.

4.2. Computational cost analysis

In the analysis of the computational cost, we use the following notations: "-" means there is no computational cost in that phase; n: the number of users; m: the number of application servers; C_h:

the cost of executing the one-way hash function; C_F: the cost of executing the fuzzy extractor; C_{SYM}: the cost of executing the symmetric encryption/decryption operation; C_{ECC}: the cost of executing the elliptic curve encryption/decryption operation; and C_{EXP}: the cost of executing the exponential operation. Generally, the cost (i.e., time complexity) associated with the different operations can be roughly described as $C_{EXP} >> C_{ECC} > C_{SYM} > C_h$ (Crypto++ Library).

Table 2 compares the computational costs of the proposed scheme and those of other biometrics-based schemes (Lin & Lai, 2004; Khan et al., 2008; Li & Hwang, 2010; Xu et al., 2008; Yang & Yang, 2010; Yoon & Yoo, 2010). From this comparison, we can see that our scheme is an efficient authentication scheme in computational cost because it does not use asymmetric cryptography. Hence, our proposed scheme is very useful in environments of limited computation and communication resources to access remote information systems.

Table 3 shows the performance comparisons among our scheme and other multi-server schemes. Ours, Tsai's (2008), and Liao and Wang's (2009) have low computational cost because they are hash-based authentication schemes. Although Tsai (2008) provides two hash-based authentication schemes, those schemes both need the registration center to assist in performing the authentication procedure, which create more authentication delays and the heavier workloads at the registration center.

We use Crypto++ Library to evaluate the computing time of the operation; see Table 4 for the computing time of each operation. According to (Chuang & Lee, 2011, 2012, 2013; Chuang, Lee, & Chen, 2013; The SANS Technology Institute-Security Laboratory, 2008), for bulk encryption, symmetric encryption is about 1000 times faster than asymmetric encryption and the hash operation is faster than symmetric encryption. Therefore, our scheme is clearly a lightweight authentication scheme. In practice, the ARM CPU of the smartphone already supports these cryptograph operations.

4.3. Comparisons with other schemes

The comparisons of the security property among our proposed scheme and other biometrics-based schemes (Khan et al., 2008; Li & Hwang, 2010; Lin & Lai, 2004; Xu et al., 2008; Yang & Yang, 2010; Yoon & Yoo, 2010) are summarized in Table 5. We can see that our scheme provides the most security properties. Furthermore, the schemes (Khan et al., 2008; Lin & Lai, 2004; Xu et al., 2008; Yoon & Yoo, 2010) store the biometrics information of the user in the smart card directly with the possible result that the user's biometrics information will be obtained by the adversary when the user's smart card is lost. Next, we compare our scheme with multi-server schemes (Chang and Lee, 2004; Juang, 2004; Liao & Wang, 2009; Tsai, 2008; Yang & Yang, 2010; Yoon & Yoo, 2010), as shown in Table 6. This overview demonstrates that our scheme achieves the most security properties. The schemes of Juang (2004), Chang and Lee (2004), Tsai (2008), Yang and Yang (2010), and Yoon and Yoo (2010) do not support the user anonymity, and the password-based schemes (i.e., low entropy) (Chang and Lee, 2004; Juang, 2004; Liao & Wang, 2009; Tsai, 2008) lead to higher system risk.

4.4. Extended discussions

(1) **Traceable problem:** In cryptography, the user's privacy includes anonymity and untraceability, where anonymity means that an adversary cannot obtain the user's real identity, and untraceability means that an adversary cannot acquire the user's behavior trajectory. In fact, our scheme supports many famous physical layer methods (e.g., *frequency-hopping spread spectrum* (*FHSS*) and time-reversal (TR) Wang, Wu, Han, Yang, & Ray Liu, 2011) to reduce the probability of the eavesdropping during the authentication procedure (i.e., it is hard for the attacker to collect the user's

Table 2
Performance comparisons of biometrics-based schemes.

		Our	Lin and Lai (2004)	Khan et al. (2008)	Xu et al. (2008)	Li and Hwang (2010)	Yang and Yang (2010)	Yoon and Yoo (2010)
Registration	User	C_h	–	–	–	–	–	C_h
	Server	–	–	–	–	–	–	–
	RC	$n(2C_h)$	$nm(C_h + C_{EXP})$	$nm(2C_h)$	$nm(2C_h + C_{EXP})$	$nm(3C_h)$	$n(3C_h + C_{EXP} + C_F)$	$(n + m)C_h$
Login	User	$4C_h$	$2C_h + 2C_{EXP}$	$2C_h$	$3C_h + 2C_{EXP}$	$2C_h$	$4C_h + C_{EXP} + C_F$	$2C_h + C_{ECC}$
	Server	–	–	–	–	–	–	–
Authentication	User	$5C_h$	–	C_h	C_h	$2C_h$	$C_h + C_{EXP}$	$3C_h + C_{ECC}$
	Server	$8C_h$	$C_h + 2C_{EXP}$	$4C_h$	$2C_h + C_{EXP}$	$3C_h$	$3C_h + 2C_{EXP}$	$5C_h + 2C_{ECC}$
	RC	–	–	–	–	–	–	$7C_h$
Password change	User	$3C_h$	$2C_h$	$2C_h$	$3C_h + C_{EXP}$	$2C_h$	$3C_h + C_F$	$2C_h$
	RC	–	–	–	–	–	–	–

Table 3
Performance comparisons of multi-server schemes.

		Our	Juang (2004)	Chang and Lee (2004)	Tsai (2008)	Liao and Wang 2009)	Yang and Yang (2010)	Yoon and Yoo (2010)
Registration	User	C_h	–	–	–	–	–	C_h
	Server	–	–	–	–	–	–	–
	RC	$n(2C_h)$	$n(C_h)$	$n(2C_h)$	$n(2C_h)$, $n(2C_h)$	$n(5C_h)$	$n(3C_h + C_{EXP} + C_F)$	$(n + m)C_h$
Login	User	$4C_h$	$2C_h + C_{SYM}$	$2C_h + C_{SYM}$	C_h, C_h	$6C_h$	$4C_h + C_{EXP} + C_F$	$2C_h + C_{ECC}$
	Server	–	–	–	–	–	–	–
Authentication	User	$5C_h$	$C_h + 2C_{SYM}$	$2C_h + 2C_{SYM}$	$4C_h$, $4C_h$	$3C_h$	$C_h + C_{EXP}$	$3C_h + C_{ECC}$
	Server	$8C_h$	$2C_h + 4C_{SYM}$	$4C_h + 3C_{SYM}$	$6C_h$, $4C_h$	$7C_h$	$3C_h + 2C_{EXP}$	$5C_h + 2C_{ECC}$
	RC	–	–	–	$6C_h$, $2C_h$	–	–	$7C_h$
Password change	User	$3C_h$	X	X	X	$3C_h$	$3C_h + C_F$	$2C_h$
	RC	–	X	X	X	–	–	–

X: no discussion.

Table 4
Computing time.

Operations	Microseconds/operation
RSA 1024 Encryption	3010
RSA 1024 Decryption	130
RSA 1024 Signature	3020
RSA 1024 Verification	130
AES 256 Encryption	0.801
AES 256 Decryption	0.801
SHA-1	0.5
SHA-512	0.76

Table 5
Comparison with other biometrics-based schemes.

	Our	Lin and Lai (2004)	Khan et al. (2008)	Xu et al. (2008)	Li and Hwang (2010)	Yang and Yang (2010)	Yoon and Yoo (2010)
C1	Yes	No	No	No	No	No	No
C2	Yes	No	Yes	Yes	Yes	Yes	Yes
C3	Yes	No[a]	Yes	Yes	Yes	Yes	Yes
C4	Yes	No	No	No	No	Yes	Yes
C5	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C6	Yes	No	No	No	Yes	Yes	No
C7	Yes	No	No	No	No	Yes	Yes
C8	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C9	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C10	Yes	No	No	No	Yes	No	Yes
C11	Yes	Yes	Yes	Yes	Yes	Yes	No[a]
C12	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C13	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C14	Yes	No	No	No	No	No	No[a]

C1: anonymity; C2: mutual authentication; C3: simple and secure password modification; C4: single registration; C5: fast error detection; C6: protection of the biometric; C7: session key agreement; C8: off-line guessing attack resistance; C9: replay attack resistance; C10: no time synchronization; C11: forgery attack resistance; C12: modification attack resistance; C13: stolen-verifier attack resistance; C14: insider attack resistance.

[a] Insecure.

communication message). In addition, the authentication procedure is not performed frequently, and once the user is authenticated successfully, the later communication is encrypted by the session key generated by random numbers.

(2) **Authorized servers fully trust each other:** We explain this assumption from two aspects which include business and hardware. In the business aspect, the application servers can issue the agreements to cooperate with each other as an alliance and then provide the PSK to the RC. In the registration phase, the RC and the user need to sign a contract which includes roles, service-level agreement (SLA), and access rights. If the server or the user violates the contract (e.g., the server betrays the user's information or the user does a malicious behavior), he must be punished. In the hardware aspect, Trusted Computing (TC), which is a mature issue, is a technology developed and promoted by the Trusted Computing Group (TCG) (Mitchell, 2005) that ensures the security of the hardware. The TCG is an initiative started by AMD, IBM, Intel, and Microsoft to implement trusted computing. Moreover, many specifications (ISO/IEC 11889-1.:2009; ISO/IEC 11889-2.:2009; ISO/IEC 11889-3.:2009; ISO/IEC 11889-4.:2009) have already been defined. In the future, this is a trend that the hardware of Internet Service Providers (ISPs) will follow, and for this reason, we think that this assumption is accepted.

(3) **The distribution of PSK:** The distribution of the PSK is a trade-off issue. If the PSK is only kept in the RC, the server's compromise problem will not happen. However, all of the users cannot be authenticated successfully if the RC crashes (i.e., a single point failure problem). Further, the authentication delay and the communication cost between the RC and the servers will increase substantially because the server needs to obtain the PSK from the RC to perform the authentication procedure every time. On the other hand, the servers provide a fault tolerant capability (i.e., if the RC crashes, the authentication procedure can still be executed because the servers can perform this procedure) if they keep the PSK. This scheme can also reduce the authentication delay and the computation load of the RC. As a result, the key distribution method can be decided by ISPs themselves according to their security policy.

(4) **The discussion of three factors:** The two-factor authentication scheme (i.e., smart card and password) is currently the most common authentication mode. Unfortunately, many two-factor schemes only guarantee the system security when either the smart card or his password is stolen, but not both. Furthermore, the password lacks the feature of uniqueness. In this paper, we add the additional factor of biometrics in order to increase the system's entropy. The main feature of the biometric is uniqueness and the user's biometric does not be stored in external device. If there is another unique authentication factor, this factor can be used instead of a biometric. However, this scheme is inconvenient for the user because the user needs to keep the extra information.

Table 6
Comparison with other multi-server schemes.

	Our	Juang (2004))	Chang and Lee (2004)	Tsai (2008)	Liao and Wang (2009)	Yang and Yang (2010)	Yoon and Yoo (2010)
C1	Yes	No	No	No	Yes	No	No
C2	Yes	Yes	Yes	Yes	No[a]	Yes	Yes
C3	Yes	No	Yes	No	Yes	Yes	Yes
C4	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C5	Yes	No	No	No	Yes	Yes	Yes
C6	Yes	No	No	No	No	Yes	Yes
C7	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C8	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C9	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C10	Yes	Yes	Yes	Yes	Yes	No	Yes
C11	Yes	Yes	Yes	Yes	Yes	Yes	No[a]
C12	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C13	Yes	Yes	Yes	No	Yes	Yes	No
C14	Yes	No	No	No	No	No	No[a]

C1: anonymity; C2: mutual authentication; C3: simple and secure password modification; C4: stolen-verifier attack resistance; C5: fast error detection; C6: three-factor security; C7: session key agreement; C8: off-line guessing attack resistance; C9: replay attack resistance; C10: no time synchronization; C11: forgery attack resistance; C12: modification attack resistance; C13: no registration center assistance; C14: insider attack resistance.

[a] Insecure.

5. Conclusions and future work

In this paper, we propose a secure remote user authentication scheme which not only supports the multi-server environment to reduce the overhead of the RC but also possesses high security properties to protect the valid user against attacks with minimal computational cost. Our scheme is suitable for real-life applications because it is a true lightweight authentication scheme that only uses the hash function. Moreover, our scheme satisfies the following security properties: anonymity, no verification tables, mutual authentication, resistance to forgery attacks, no clock synchronization problem, resistance to modification attacks, resistance to replay attacks, fast error detection, resistance to off-line guessing attacks, resistance to insider attacks, simple and secure choice and change of passwords, biometric template protection,

and session key agreement. We compare the proposed scheme with other existing schemes, and the comparison results clearly show that our scheme has more security properties than the others.

In the future, we will propose a cryptanalysis scheme to prove that our authentication mechanism is secure and discuss the biometric matching issue in detail. Moreover, we will build a biometric-based authentication testbed and extend our scheme for micropayment services.

References

Chang, C. C., & Lin, I. C. (2004). Remarks on fingerprint-based remote user authentication scheme using smart cards. *ACM SIGOPS Operating Systems Review, 38*(4), 91–96.

Chang, C. C., & Lee, J. S. (2004). An efficient and secure multi-server password authentication scheme using smart cards. In *IEEE international conference on cyberworlds (CW)* (pp. 417–422).

Chuang, Ming-Chin., & Lee, Jeng-Farn. (2013). TEAM: Trust-extended authentication mechanism for vehicular ad hoc networks. *IEEE Systems Journal* (in press). http://dx.doi.org/10.1109/JSYST.2012.2231792.

Chuang, Ming-Chin, & Lee, Jeng-Farn (2011). A lightweight mutual authentication mechanism for network mobility in IEEE 802.16e wireless networks. *Computer Networks, 55*(16), 3796–3809.

Chuang, Ming-Chin, & Lee, Jeng-Farn (2012). SF-PMIPv6: A secure fast handover mechanism for proxy mobile IPv6 networks. *Journal of Systems and Software*, 437–448.

Chuang, Ming-Chin, Lee, Jeng-Farn, & Chen, Meng-Chang (2013). SPAM: A secure password authentication mechanism for seamless handover in proxy mobile IPv6 networks. *IEEE Systems Journal, 7*, 102–113.

Crypto++ Library 5.6.1. (2013). Available at: http://www.cryptopp.com/.

Dondi, D., Bertacchini, A., Brunelli, D., Larcher, L., & Benini, L. (2008). Modeling and optimization of a solar energy harvester system for self-powered wireless sensor networks. *IEEE Transactions on Industrial Electronics, 55*(7), 2759–2766.

Fan, C., Chan, Y., & Zhang, Z. (2005). Robust remote authentication scheme with smart cards. *Computer Security, 24*(8), 619–628.

Fan, Chun-I, & Lin, Yi-Hui (2009). Provably secure remote truly three-factor authentication scheme with privacy protection on biometrics. *IEEE Transactions on Information Forensics and Security, 4*(4), 933–945.

Gil-Castineira, F., Gonzalez-Castano, F. J., & Franck, L. (2008). Extending vehicular CAN fieldbuses with delay-tolerant networks. *IEEE Transactions on Industrial Electronics, 55*(9), 3307–3314.

He, Diabao. (2011). Security flaws in a biometrics-based multi-server authentication with key agreement scheme. In *IACR Cryptology* (pp. 1–9).

Hwang, C. L., Chang, L. J., & Yu, Y. S. (2007). Network-based fuzzy decentralized sliding-mode control for car-like mobile robots. *IEEE Transactions on Industrial Electronics, 54*(1), 574–585.

ISO/IEC 11889-1:2009, (2009). Information technology – trusted platform module. Part 1: Overview.

ISO/IEC 11889-2:2009, (2009). Information technology – trusted platform module. Part 2: Design principles.

ISO/IEC 11889-3:2009, (2009). Information technology – trusted platform module. Part 3: Structures.

ISO/IEC 11889-4:2009, (2009). Information technology – trusted platform module. Part 4: Commands.

Jea, Tsai-Yang, & Govindaraju, Venu (2005). Aminutia-based partial fingerprint recognition system. *Pattern Recognition, 38*(10), 1672–1684.

Juang, Wen-Shenq (2004). Efficient multi-server password authenticated key agreement using smart cards. *IEEE Transaction on Consumer Electronic, 50*(1), 251–255.

Juang, W. S., Chen, S. T., & Liaw, H. T. (2008). Robust and efficient password-authenticated key agreement using smart cards. *IEEE Transactions on Industrial Electronics, 55*(6), 2551–2556.

Kaufman, C. (2005). Internet Key Exchange (IKEv2) Protocol. RFC 4306, December 2005.

Khan, M. K., & Zhang, J. (2006). An efficient and practical fingerprint-based remote user authentication scheme with smart cards. *Springer Lecture Notes in Computer Science*, 260–268.

Khan, M. K., Zhang, J., & Wang, X. (2008). Chaotic hash-based fingerprint biometric remote user authentication scheme on mobile devices. *Chaos, Solitons and Fractals, 35*(3), 519–524.

Ku, W. C., Chang, S. T., & Chiang, M. H. (2005). Further cryptanalysis of fingerprint-based remote user authentication scheme using smartcards. *Electronics Letters, 41*(5), 240–241.

Lamport, L. (1981). Password authentication with insecure communication. *ACM Communication, 24*(11), 770–772.

Lazar, C., & Carari, S. (2008). A remote-control engineering laboratory. *IEEE Transactions on Industrial Electronics, 55*(6), 2368–2375.

Lee, J. K., Ryu, S. R., & Yoo, K. Y. (2002). Fingerprint-based remote user authentication scheme using smart cards. *Electronics Letters, 38*(12), 554–555.

Li, Chun-Ta, & Hwang, Min-Shiang (2010). An efficient biometrics-based remote user authentication scheme using smart cards. *Journal of Network and Computer Applications, 33*(1), 1–5.

Liao, Y. P., & Wang, S. S. (2009). A secure dynamic ID based remote user authentication scheme for multi-server environment. *Computer Standards & Interfaces, 31*(1), 24–29.

Lin, Ching-Ying, & Hwang, Tzone-lih (2011). On 'a simple three-party password-based key exchange protocol'. *International Journal of Communication Systems (IJCS), 24*, 1520–1532.

Lin, C. H., & Lai, Y. Y. (2004). A flexible biometrics remote user authentication scheme. *Computer Standards & Interfaces, 27*(1), 19–23.

Liu, G. P., Xia, Y., Chen, J., Rees, D., & Hu, W. (2007). Networked predictive control of systems with random network delays in both forward and feedback channels. *IEEE Transactions on Industrial Electronics, 54*(3), 1282–1297.

Liu, Feng, Zhao, Qijun, & Zhang, David (2011). A novel hierarchical fingerprint matching approach. *Pattern Recognition, 44*(8), 1604–1613.

Marino, P., Poza, F., Dominguez, M. A., & Otero, S. (2009). Electronics in automotive engineering: A top–down approach for implementing industrial fieldbus technologies in city buses and coaches. *IEEE Transactions on Industrial Electronics, 56*(2), 589–600.

Meenen, Peter, Ashrafi, Ashkan, & Adhami, Reza (2006). the utilization of a Taylor series-based transformation in fingerprint verification. *Pattern Recognition Letters, 27*(14), 1606–1618.

Mitchell, Chris. (2005). Trusted computing, Institution of Electrical Engineers, 2005.

Mitchell, C. J., & Tang, Q. (2005). Security of the Lin–Lai smart card based user authentication scheme. Technical report (Online). Available: http://www.rhul.ac.uk/mathematics/techreports.

Sun, Da-Zhi, Huai, Jin-Peng, Sun, Ji-Zhou, Li, Jian-Xin, Zhang, Jia-Wan, & Feng, Zhi-Yong (2009). Improvements of Juang's password-authenticated key agreement scheme using smart cards. *IEEE Transactions on Industrial Electronics, 56*(6), 2284–2291.

The SANS Technology Institute-Security Laboratory, 2008. Hash Functions. http://www.sans.edu/resources/securitylab/hash_functions.php, January 2008.

Tong, Xifeng, Liu, Songbo, Huang, Jianhua, & Tang, Xianglong (2008). Local relative location error descriptor-based fingerprint minutiae matching. *Pattern Recognition Letters, 29*(3), 286–294.

Tsai, J. L. (2008). Efficient multi-server authentication scheme based on one-way hash function without verification table. *Computers & Security, 27*(3–4), 115–121.

Wang, B. B., Wu, Y. L., Han, F., Yang, Y. H., & Ray Liu, K. J. (2011). Green wireless communications: A time-reversal paradigm. *IEEE Journal on Selected Areas in Communications (JSAC), 29*(8), 1698–1710.

Xu, Jing, Zhu, Wen-Tao, & Feng, Deng-Guo. (2008). Improvement of a fingerprint-based remote user authentication scheme. In *IEEE international conference on information security and assurance (ISA)* (pp. 87–92).

Yager, Neil, & Amin, Adnan (2006a). Fingerprint alignment using a two stage optimization. *Pattern Recognition Letters, 27*(5), 317–324.

Yager, Neil, & Amin, Adnan (2006b). Dynamic registration selection for fingerprint verification. *Pattern Recognition, 39*(11), 2141–2148.

Yang, Dexin, & Yang, Bo. (2010). A biometric password-based multi-server authentication scheme with smart card. In *IEEE international conference on computer design and applications (ICCDA)* (pp. 554–559).

Yeh, Kuo-Hui, Lo, N. W., & Li, Yingjiu (2011). Cryptanalysis of Hsiang–Shih's authentication scheme for multi-server architecture. *International Journal of Communication Systems (IJCS), 24*, 829–836.

Yoon, E.-J., & Yoo, K.-Y. (2010). Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem. *Journal of Supercomputing*, 1–21.

Yoon, Eun-Jun, & Yoo, Kee-Young (2011). Cryptanalysis of a simple three-party password-based key exchange protocol. *International Journal of Communication Systems (IJCS), 24*, 532–542.

Zhu, En., Yin, Jianping, & Zhang, Guomin (2005). Fingerprint matching based on global alignment of multiple reference minutiae. *Pattern Recognition, 38*(10), 1685–1694.