A Novel Mutual Authentication Scheme Based on Fingerprint Biometric and Nonce Using Smart Cards

De-song Wang, Jian-ping Li

School of Computer Science and Engineering, University of Electronic Science and Technology of China
No.4, Section 2, North Jianshe Road, Chengdu 610054, Sichuan, P. R. China
desong.wangg@gmail.com

Abstract

In 2007, Khan-Zhang made an enhancement based on Lin-Lai's flexible biometrics remote user authentication scheme. The scheme has the merits of providing mutual authentication, no verification table, freely changing password and preventing the server spooling attack. However, this authentication scheme has been found to be vulnerable to the insider attack, the denial-of-service (DoS) attack and the clock synchronization problem. To overcome these weaknesses, a novel authentication scheme is proposed in this paper, which is based on nonce instead of timestamp and fresh tag to overcome the existing DoS attack and clock synchronization problem. The security analysis shows that the improved scheme not only inherits the merits of their scheme but also enhances the security of their scheme. Meantime the improved scheme does not add additional computation cost to the smart card. So the improved scheme is more secure, reliable and applicable with high potential to be used in the insecure network world than Khan-Zhang's scheme.

Keywords: *Authentication; Fingerprint verification; Security; Smart card; Attack; Nonce.*

1. Introduction

Remote user authentication scheme is a procedure which allows a server to authenticate a remote user through an insecure channel. Password-based authentication scheme is the most common method to check the validity of the login message and authenticate the user. In 1981, Lamport [1] proposed a password-based authentication scheme using password tables to authenticate remote users over an insecure network. In Lamport's scheme, password table was used to verify the legitimacy of users, but if this password table is compromised, stolen, or modified by an adversary, then the system could be partially or completely compromised. Later, Shimizu [2] pointed out the weakness of Lamport's scheme [1] and proposed a modified scheme. Then, many improved remote user authentication schemes [3-10] have been proposed.

Recently, some biometric-based remote user identity authentication schemes are also proposed in [11], [12], [13] and [14]. Among these schemes, Lee et al. [11] first proposed a fingerprint-based remote user authentication scheme using smart cards. Their scheme is based on ElGamal public key cryptosystem, which also does not require password table for authentication. Their scheme is novel because they used biometrics and two secret keys to improve the security, and to protect the system from the attacks. Unfortunately, their scheme faces some problems and Lin-Lai [12] pointed out that user have no way to choose and change his password in the system of Lee et al. Moreover, they also pointed out that the scheme of Lee et al. is vulnerable to masquerade attack. Later on, Ku et al. [13] also revealed a forgery attack on the scheme of Lee et al., in which an intruder can impersonate any legal user. In addition, Ku et al. also shown that the scheme of Lee et al. is not easily repairable. However, in 2007, Khan-Zhang [14] pointed out that Lin-Lai's scheme [12] performs only unilateral authentication (only user authentication), and user has no information either the authentication server is authentic or not. Hence, Lin-Lai's scheme is vulnerable to the server spoofing attack. To overcome this weakness, Khan-Zhang proposed an improved security patch, which performs mutual authentication between user and remote server and can withstand the server spoofing attack found in Lin-Lai's scheme.

In this paper, we state Khan-Zhang's scheme is vulnerable to the insider attack, the DoS attack and the existing clock synchronization problem. To remedy these pitfalls, this paper presents an improvement scheme. The improved scheme is based on nonce instead of timestamp and fresh tag to overcome the existing DoS attack and clock synchronization problem. The security analysis shows that the improved scheme not only inherits the merits of their scheme but also enhances the security of their scheme. Meantime the improved scheme is not add additional computation cost to the smart card.

The rest of the paper is organized as follows: Section 2 briefly reviews Khan-Zhang's scheme. Section 3 elaborates the weaknesses of Khan-Zhang's scheme. Section 4 presents an improvement scheme for Khan-Zhang's scheme. Section 5 demonstrates the security analysis of the proposed improvement. The conclusion is given in Section 6.

Table 1. Notations Used in this Paper

U_i	User
RS	Remote server
RC	Registration center
ID_i	Identity of user
PW_i	Password shared between U_i and RS
F_i	Fingerprint template of the user
$h(\cdot)$	Collision-free one way hash function
X_S	Secret key of the registration server
p	Large prime number
r	Random number using the minutiae extracted from the fingerprint template
R	64-bit random number
T_u	Timestamp of the login device

T_S	Timestamp of the remote server
DT	Expected valid time interval for transmission delay
N	Random nonce
Å	XOR operation

2. Review of Khan-Zhang's Scheme

There are four phases in Khan-Zhang's scheme [14], namely: registration, login, authentication, and change password. Figure 1illustrates Khan-Zhang's authentication scheme. In the following subsections, we briefly review their scheme. The notations in the Table 1 are used in this paper.

2.1. Registration Phase

Before the remote user logins to the remote system, the user needs to perform the following steps.

R1: First, the user U_i chooses his/her ID_i , password PW_i and inputs his/her personal fingerprint biometric F_i on the fingerprint device to the registration center in person.

R2: Next, the registration center computes $PW_i¢$ and Y_i as follows:

$$PW_i¢ = h(PW_i \text{ Å } F_i) \bmod p$$

$$Y_i = (ID_i^{X_s} \bmod p \text{ Å } PW_i¢)$$

R3: Lastly, the registration center stores $\{h(\cdot), p, Y_i, F_i, ID_i\}$ on the user's smart card and issues it to the user via a secure channel.

2.2. Login Phase

Whenever the user wants to logon to the remote server, he/she must perform the following steps.

L1: First, U_i inserts his/her smart card into the card reader and inputs the personal fingerprint biometric F_i on the fingerprint device to verify the user's fingerprint biometrics.

L2: If U_i does not pass the fingerprint verification, then remote user authentication scheme is terminated. On the contrary, If U_i passes the fingerprint verification, then the smart card generates a random number r using the minutiae extracted from the fingerprint template and U_i enters PW_i to perform the following operations in L3.

L3: After receiving U_i 's password, the smart card will compute the following messages:

$$PW_i¢¢ = h(PW_i \text{ Å } F_i) \bmod p$$

$$Y_i¢ = Y_i \text{ Å } PW_i¢¢$$

$$C_1 = (ID_i)^r \bmod p$$

$$M = h(Y_i¢ \text{ Å } T_u) \bmod p$$

$$C_2 = (Y_i¢)^r M \bmod p$$

L4: Finally, U_i sends the login message $C = \{ID_i, C_1, C_2, T_u\}$ to RS for the authentication process.

2.3. Authentication Phase

After receiving the request login message from the user at current time T_S , RS will perform the following steps to authenticate that the user is legal or not.

A1: First, RS checks whether the format of ID_i is valid or not.

A2: If the format is not valid, RS rejects the login request. On the contrary, if the format is valid, RS further checks T_S - T_u < DT or not.

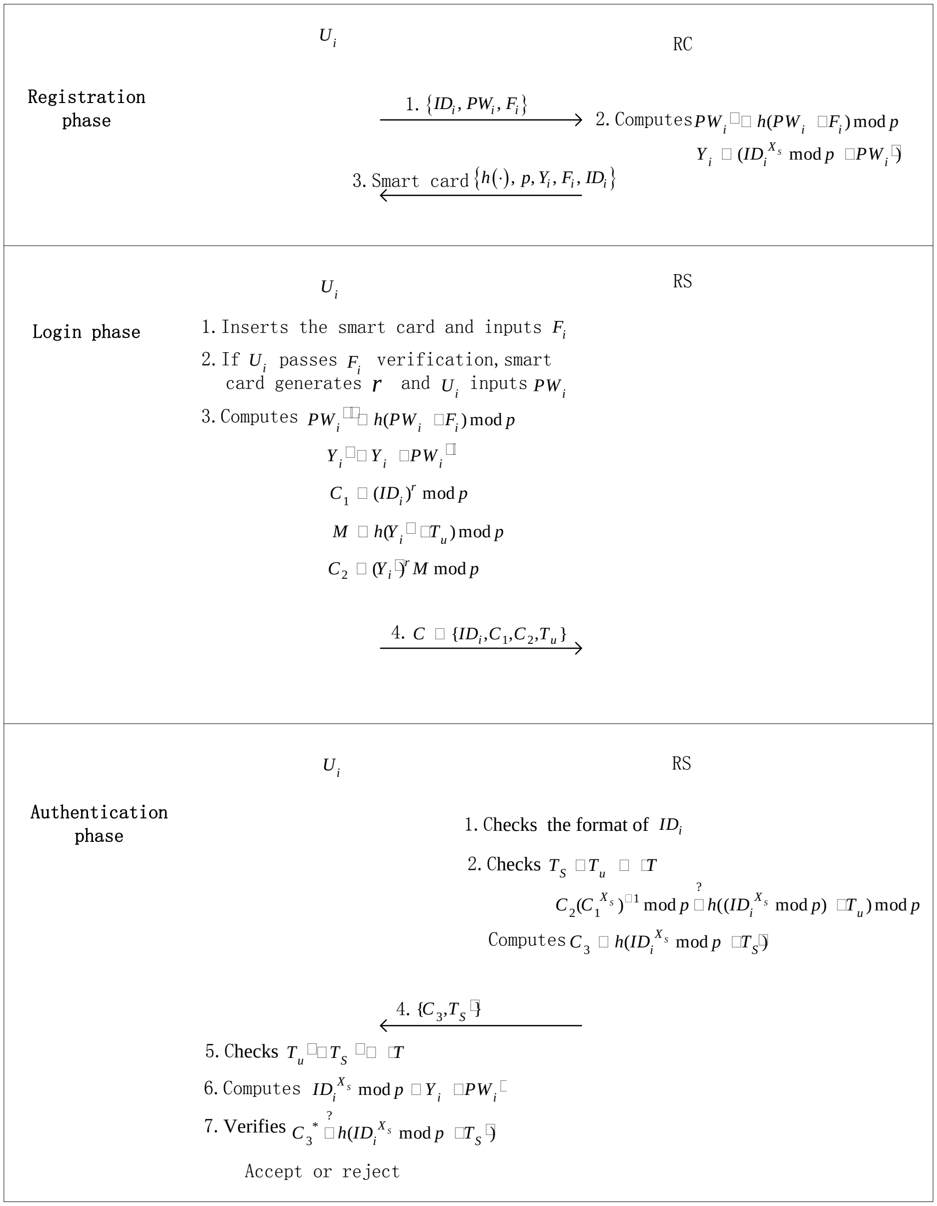
U_i
RC
Registration phase
1. $\{ID_i, PW_i, F_i\}$
2. Computes PW_i□□ $h(PW_i$ □$F_i)$ mod p
Y_i □ $(ID_i^{X_s}$ mod p □PW_i□)
3. Smart card $\{h(\cdot), p, Y_i, F_i, ID_i\}$
U_i
RS
Login phase
1. Inserts the smart card and inputs F_i
2. If U_i passes F_i verification, smart card generates r and U_i inputs PW_i
3. Computes PW_i□□ $h(PW_i$ □$F_i)$ mod p
Y_i□□ Y_i □PW_i□
C_1 □ $(ID_i)^r$ mod p
M □ $h(Y_i$□ □$T_u)$ mod p
C_2 □ $(Y_i$□$)^r M$ mod p
4. C □ $\{ID_i, C_1, C_2, T_u\}$
U_i
RS
Authentication phase
1. Checks the format of ID_i
2. Checks T_S □T_u □ □T
$C_2(C_1^{X_s})^{□1}$ mod $p \overset{?}{□} h((ID_i^{X_s}$ mod $p)$ □$T_u)$ mod p
Computes C_3 □ $h(ID_i^{X_s}$ mod p □T_S□)
4. $\{C_3, T_S$□$\}$
5. Checks T_u□□T_S □□ □T
6. Computes $ID_i^{X_s}$ mod p □Y_i □PW_i□
7. Verifies $C_3^* \overset{?}{□} h(ID_i^{X_s}$ mod p □T_S□)
Accept or reject

Figure 1. Khan-Zhang's Scheme

A3: If it holds, RS then verifies whether $C_2(C_1^{X_s})^{-1} \bmod p \overset{?}{=} h((ID_i^{X_s} \bmod p)\, Å\, T_u) \bmod p$ or not. If it does not hold true, then the login request is rejected; otherwise, RS accepts the login request. And RS computes the following message:

$$C_3 = h(ID_i^{X_s} \bmod p\, Å\, T_S¢)$$

A4. Then, RS sends the message $\{C_3, T_S¢\}$ to U_i .

A5. After receiving RS's message at current time $T_u¢$, U_i first checks $T_u¢ - T_S¢ < \mathrm{D}T$ or not.

A6. If it holds, U_i then computes the following message:

$$ID_i^{X_s} \bmod p = Y_i\, Å\, PW_i¢$$

Where Y_i is stored in U_i 's smart card and $PW_i¢$ is the password of the user.

A7. Finally, U_i computes C_3^* and validates either $C_3^* \overset{?}{=} h(ID_i^{X_s} \bmod p\, Å\, T_S¢)$ or not. If it holds true, U_i believes that the responding party is authentic RS and mutual authentication between U_i and RS is completed, otherwise U_i terminates the connection.

2.4. Password Change Phase

Whenever U_i wants to change the old password PW_i to the new password PW_i^* , he/she has to imprint his/her fingerprint biometric F_i on the fingerprint device, then smart card compares it with the template stored on the smart card. If U_i passes the fingerprint verification, he/she then inputs old password PW_i and new password PW_i^* . The smart card will perform the following operations:

$$PW_i¢¢ = h(PW_i\, Å\, F_i) \bmod p$$

$$Y_i¢ = Y_i\, Å\, PW_i¢¢ = ID_i^{X_s} \bmod p$$

$$Y_i^* = Y_i¢\, Å\, h(PW_i^*\, Å\, F_i) \bmod p$$

Finally, replace the old Y_i with the new Y_i^* on the smart card.

3. Weaknesses of Khan-Zhang's Scheme

This section shows that Khan-Zhang's scheme [14] is vulnerable to the insider attack. Clock synchronization problem and DoS attack also exist in the scheme.

3.1. Suffering Insider Attack

If the password of a user can be derived by the server in the registration phase, it is called the insider attack [4, 10, 15]. In Khan-Zhang's scheme [14], users' passwords will be revealed to the remote system because they are directly transmitted to the system, so the server can get all the users' passwords in the registration phase. The insider of the server can use these passwords to access other servers with the same passwords [16]. In practice, users offer the same password to access several remote servers for their convenience. Thus, the insider of the remote system may try to use PW_i to impersonate U_i to login to the other remote systems that U_i has registered with outside this system. If the targeted outside remote system adopts the normal password authentication scheme, it is possible that the insider of the remote system could successfully impersonate U_i to login to it by using PW_i. Although it is also possible that all the insiders of the remote system can be trusted and that U_i does not use the same password to access several systems, the implementers and the users of the scheme should be aware of such a potential weakness.

3.2. Suffering Clock Synchronization Problem

The authentication scheme is used to carry out the timestamp verification in the authentication process, so the scheme results in a clock synchronization problem. If the system time of the remote server is faster $\mathrm{D}T$ than the user's system time, then $T_S - T_u < \mathrm{D}T$ is not satisfied, where T_S is the current timestamp of the remote server, T_u is the current timestamp of the device and $\mathrm{D}T$ denotes the expected valid time interval for transmission delay, so a valid request will be caused to reject.

3.3. Suffering DoS Attack

The remote server is also in existence of the DoS attack in Khan-Zhang's scheme [14]. If an attacker intercepts a request login message $C = \{ID_i, C_1, C_2, T_u\}$, then just select the appropriate time T_u^* or modify T_u large enough, and construct $(T_S - T_u^*) < \mathrm{D}T$ satisfied, send messages $C^* = \{ID_i, C_1, C_2, T_u^*\}$ to the remote server. The result is the attacker can pass through the remote server verification of the 1-2 step-by-step in the authentication phase, and make the remote server to busily compute and verify the step 3 of the authentication phase. So this will result in the DoS attack.

4. Proposed Improvement Scheme

In this section, we propose an enhancement to Khan-Zhang's scheme [14] that can withstand the security weaknesses described in previous sections. The proposed improvement scheme is also composed of four phases: registration, login, authentication, and password change. The scheme is illustrated in Figure 2. Now, we describe the four phases separately in our scheme as follows.

4.1. Registration Phase

Before the remote user logins to the remote system, the user needs to perform the following steps.

Table 2. ID **storage table**

User serial number	ID
1	ID_1
2	ID_2
…	…
i	ID_i
…	…

R1: First, the user U_i chooses his/her ID_i , password PW_i and a random number R . The registration center of the remote server searches user ID_i in the user ID storage table (see Table 2). If it exists, then return to require the user U_i to re-choose his/herID_i ; otherwise, user U_i interactively submits $\{ID_i, h(PW_i \oplus R)\}$ to the registration center in a secure channel, and inputs his/her personal fingerprint biometric F_i on the fingerprint device to the registration center in person.

R2: Next, the registration center computes PW_i' and Y_i as follows:

$$PW_i' = h(h(PW_i \oplus R) \oplus F_i) \bmod p$$

$$Y_i = (ID_i^{X_s} \bmod p \oplus PW_i')$$

R3: Lastly, the registration center stores $\{h(\cdot), p, Y_i, F_i, ID_i\}$ on the user's smart card and sends it to the user via a secure channel.

R4: After receiving the smart card, U_i enters R into his/her smart card.

4.2. Login Phase

Whenever the user wants to logon to the remote server, he/she must perform the following steps.

L1: First, U_i inserts his/her smart card into the card reader and inputs the personal fingerprint biometric F_i on the fingerprint device to verify the user's fingerprint biometrics.

L2: If U_i does not pass the fingerprint verification, then remote user authentication scheme is terminated. On the contrary, If U_i passes the fingerprint verification, then the smart card generates a random number r using the minutiae extracted from the fingerprint template and U_i enters PW_i to perform the following operations in L3.

L3: After receiving U_i 's password, the card reader generates a fresh random nonce ("Nonce" means "used only once." [17, 3]) N_1 , then the smart card will compute the following messages:

$$PW_i'' = h(h(PW_i \oplus R) \oplus F_i) \bmod p$$

$$Y_i' = Y_i \oplus PW_i''$$

$$C_1 = (ID_i)^r \bmod p$$

$$M = h(Y_i' \oplus N_1) \bmod p$$

$$C_2 = (Y_i')^r M \bmod p$$

L4: Finally, U_i sends the login message $C = \{ID_i, C_1, C_2, N_1\}$ to RS for the authentication process.

4.3. Authentication Phase

To discuss conveniences, we have given the following definition of fresh tag in the authentication phase.

Definition 1. For any message which is sent by users, if it is the first time arising message, then it is fresh and acceptable; otherwise it is not fresh, and then the system rejects any service.

After receiving the request login message from the user, RS will perform the following steps to authenticate that the user is legal or not.

A1: First, RS sets up a counter and a timestamp for the ID_i, which is used to calculate the frequency of the ID_i. RS checks the session state table (see Table 3) to see whether the ID_i is in the session state or not. If so, the login request is rejected; otherwise RS checks further user ID storage table to see if it has been in existence of the ID_i. If it does not exist, RS rejects the request of the user; otherwise RS checks the frequency value of the user ID_i or the fresh tag of messages $\{ID_i, C_{1i}, C_{2i}, N_{1i}\}$, if the value is more than the experience of a certain threshold or the fresh tag of messages $\{ID_i, C_{1i}, C_{2i}, N_{1i}\}$ is not fresh, then that is illegal users try to login RS or illegal to attacks on RS, so RS deletes or quarantines review of the ID_i; otherwise performs step 2. In short, RS checks the validity of ID_i. If ID_i is invalid, it rejects the login request.

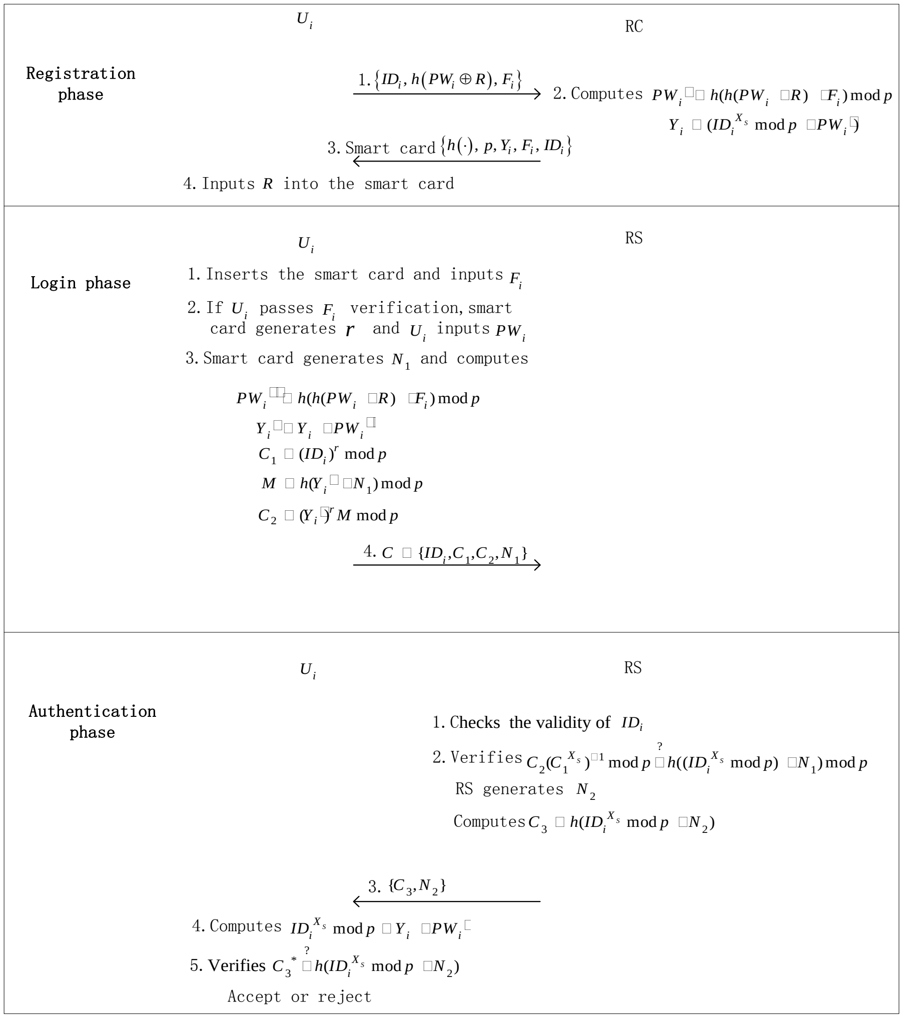
U_i
RC
Registration phase
1. $\{ID_i, h(PW_i \oplus R), F_i\}$
2. Computes $PW_i^{\square} \square h(h(PW_i \square R) \square F_i) \bmod p$
$Y_i \square (ID_i^{X_s} \bmod p \square PW_i^{\square})$
3. Smart card $\{h(\cdot), p, Y_i, F_i, ID_i\}$
4. Inputs R into the smart card
U_i
RS
Login phase
1. Inserts the smart card and inputs F_i
2. If U_i passes F_i verification, smart card generates r and U_i inputs PW_i
3. Smart card generates N_1 and computes
$PW_i^{\square\square} \square h(h(PW_i \square R) \square F_i) \bmod p$
$Y_i^{\square} \square Y_i \square PW_i^{\square}$
$C_1 \square (ID_i)^r \bmod p$
$M \square h(Y_i^{\square} \square N_1) \bmod p$
$C_2 \square (Y_i^{\square})^r M \bmod p$
4. $C \square \{ID_i, C_1, C_2, N_1\}$
U_i
RS
Authentication phase
1. Checks the validity of ID_i
2. Verifies $C_2(C_1^{X_s})^{\square 1} \bmod p \overset{?}{\square} h((ID_i^{X_s} \bmod p) \square N_1) \bmod p$
RS generates N_2
Computes $C_3 \square h(ID_i^{X_s} \bmod p \square N_2)$
3. $\{C_3, N_2\}$
4. Computes $ID_i^{X_s} \bmod p \square Y_i \square PW_i^{\square}$
5. Verifies $C_3^* \overset{?}{\square} h(ID_i^{X_s} \bmod p \square N_2)$
Accept or reject

Figure 2. Our Improvement Scheme

A2: If A1 holds, RS then verifies whether $C_2(C_1^{X_s})^{-1} \bmod p \overset{?}{=} h((ID_i^{X_s} \bmod p) \oplus N_1) \bmod p$ or not. If it does not hold true, then the login request is rejected; otherwise, RS accepts the login request. And RS generates a fresh random nonce ("Nonce" means "used only once." [17, 3]) N_2 and computes the following message:

$$C_3 = h(ID_i^{X_s} \bmod p \oplus N_2)$$

A3. Then, RS sends the message $\{C_3, N_2\}$ to U_i .

A4. After receiving RS's message, U_i then computes the following message:

$$ID_i^{X_s} \bmod p = Y_i \oplus PW_i'$$

Where Y_i is stored in U_i 's smart card and PW_i' is the password of the user.

A5. Finally, U_i computes C_3^* and validates either $C_3^* \overset{?}{=} h(ID_i^{X_s} \bmod p \oplus N_2)$ or not. If it holds true, U_i believes that the responding party is authentic RS and mutual authentication between U_i and RS is completed, otherwise U_i terminates the connection.

Table 3. ID Session State Table

ID which is applying Conversation	Message received $\{ID_i, C_1, C_2, N_1\}$	RS time T
ID_1	$\{ID_1, C_{11}, C_{21}, N_{11}\}$	T_1
ID_3	$\{ID_3, C_{13}, C_{23}, N_{13}\}$	T_3
ID_5	$\{ID_5, C_{15}, C_{25}, N_{15}\}$	T_5
…	…	…
ID_i	$\{ID_i, C_{1i}, C_{2i}, N_{1i}\}$	T_i
…	…	…

4.4. Password Change Phase

Whenever U_i wants to change the old password PW_i to the new password PW_i^* , he/she has to imprint his/her fingerprint biometric F_i on the fingerprint device, then smart card compares it with the template stored on the smart card. If U_i passes the fingerprint verification, he/she then inputs old password PW_i and new password PW_i^* . The smart card will perform the following operations:

$$PW_i'' = h(h(PW_i \oplus R) \oplus F_i) \bmod p$$

$$Y_i' = Y_i \oplus PW_i'' = ID_i^{X_s} \bmod p$$

$$Y_i^* = Y_i' \oplus h(PW_i^* \oplus F_i) \bmod p$$

Finally, replace the old Y_i with the new Y_i^* on the smart card.

5. Security Analysis

The security of the improved scheme is still based on the security of one-way hash function and the difficulty of computing the discrete logarithm. In the following, we will discuss security of the improved scheme.

5.1. Preventing Insider Attack

The insider attack is when the user's password is obtained by the server in the registration phase [4, 10, 15]. Therefore, the user must conceal his/her password from the server to prevent the insider attack. In our scheme, the user will choose a random number R and generate $h(PW_i \text{ Å } R)$. Then he/she sends $h(PW_i \text{ Å } R)$ to the server for registration. The server cannot know the correct password PW_i since the entropy of R is very large.

5.2. Preventing Replay Attack

The proposed scheme can withstand message replay attack for the authentication system without synchronization clocks by using random nonce in place of timestamps. An attacker pretending to be a user may attempt to login to the server by sending messages ever transmitted by a legal user. Our scheme uses the nonce-based method to withstand the replay attack. Nonce variables N_1 and N_2 are generated independently, and both values will be different in each session. This ensures that authentication messages exposed in an unsecured channel are distinct among all sessions of authentication. Thus, an attacker has no opportunity to successfully replay used messages. Two nonce values used in our schemes can prevent replay attacks to either side of the authentication system.

5.3. Preventing DoS Attack

As the remote server sets up the conversation state table in the authentication process, so can effectively prevent the DoS attack by testing the frequency value of ID_i and fresh tag of messages $\{ID_i, C_{1i}, C_{2i}, N_{1i}\}$.

5.4. Preventing Guessing Attack

It is impossible for an attacker to compute the user password PW_i from the intercepted messages $\{ID_i, C_1, C_2, N_1\}$ and $\{C_3, N_2\}$, which include no information about the password. It is also extremely hard for an attacker to derive the remote server secret key X_s from the eavesdropped messages $\{ID_i, C_1, C_2, N_1\}$ and $\{C_3, N_2\}$, because of the property of the collision free one-way hash function and the difficulty of computing the discrete logarithm.

5.5. Preventing Server Spoofing Attack

An attacker may try to masquerade as a server such that users send confidential information to the spoofing server. In our improved scheme, a user will first authenticate the server in the registration phase. Thus, to successfully masquerade as the server, an attacker must provide the mutual authentication messages $\{C_3, N_2\}$ correctly. Since C_3 is computed by $C_3 = h(ID_i^{X_s} \bmod p \text{ Å } N_2)$, the attacker cannot generate C_3 without knowing the secret key X_S of the server. Thus, our improved scheme can also successfully resist the server spoofing attack.

5.6. Preventing Forgery Attack

A valid user's login message comprises ID_i, C_1, C_2 and N_1, where $C_1 = (ID_i)^r \bmod p$ and $C_2 = (Y_i\text{¢})^r M \bmod p$. An attacker cannot make a valid C_1 and C_2 without the information of the server's secret key X_S and the user's password PW_i and the random number r. Note that the random number r is generated by using the coordinate of imprint fingerprint minutiae. This method can generate a one-time random number because the picture of matched minutiae is always different [11, 12].

Table 4. The security property comparison between our scheme and Khan-Zhang's scheme

	Ours	Khan-Zhang
Insider attack	No	Yes
Replay attack	No	No
DoS attack	No	Yes
Guessing attack	No	No
Server spoofing attack	No	No
Forgery attack	No	No
Mutual authentication	Yes	Yes
No clock synchronization	Yes	No

5.7. Achieving Mutual Authentication

The improved scheme can also achieve mutual authentication: RS can authenticate the user U_i in step A2 of the authentication phase because only the valid RS can compute and verify $C_2(C_1^{X_s})^{-1} \bmod p \overset{?}{=} h((ID_i^{X_s} \bmod p) \text{ Å } N_1) \bmod p$. User U_i can also authenticate RS because only the legitimate remote user U_i can compute $ID_i^{X_s} \bmod p = Y_i \text{ Å } PW_i\text{¢}$ and $C_3^* = h(ID_i^{X_s} \bmod p \text{ Å } N_2)$. Therefore, the improved scheme can achieve mutual authentication.

The security properties of Khan-Zhang's scheme and of the improved scheme are summarized in Table 4. In contrast with Khan-Zhang's scheme, the proposed scheme is more secure.

6. Conclusion

In this paper, we demonstrate Khan-Zhang's scheme is vulnerable to the insider attack, the denial-of-server attack and the existing clock synchronization problem. To remedy these pitfalls, we present an improvement scheme. The improved scheme can also safely achieve mutual authentication between the users and the remote system. Moreover, the improved scheme has the important merits as follows: (1) it can prevent the insider attack; (2) it can effectively prevent the denial-of-service attack by testing the frequency value of ID_i and fresh tag of the login messages $\{ID_i, C_{1i}, C_{2i}, N_{1i}\}$; (3) it can overcome the existing clock synchronization and transmission delay problem. And the security analysis shows that the improved scheme not only inherits the merits of their scheme but also enhances the security of their scheme. Meantime the improved scheme does not add additional computation cost to the smart card.

Acknowledgements

The research in this paper is in part supported by the National High Technology Research and Development Program of China under Grant No.2007AA01Z423, National Natural Science Foundation under Grant No. 60703113 and No. 60803028, Sichuan Committee of Economics under Grant No. 2008CD00053

References

[1] L. Lamport, “Password authentication with insecure communication”, Communications of the ACM, 24(11) 1981, pp. 770-772.

[2] A. Shimizu, T. Horioka, and H. Inagaki, “A password authentication method for contents communication on the Internet”, IEICE Transactions on Communications, E81-B(8), 1998, pp. 1666-1673.

[3] C. Fan, Y. Chan, and Z. Zhang, “Robust remote authentication scheme with smart cards”, Computers & Security, 24, 2005, pp. 619–628.

[4] W. Juang, “Efficient password authenticated key agreement using smart card”, Computer & Security, 23, 2004, pp. 167-173.

[5] W. Ku and S. Chen, “Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards”, IEEE Transactions on Consumer Electronics, 50(1), 2004, pp. 204-207.

[6] C. Lee, L. Li, and M. Hwang, “A remote user authentication scheme using hash functions”, ACM Operating Systems Review, 36(4), 2002, pp. 23-29.

[7] M. Peyravian and N. Zunic, “Methods for protecting password transmission”, Computers & Security, 19(5), 2000, pp. 466-469.

[8] W. Ku, “A hash-based strong-password authentication scheme without using smart cards”, ACM Operating Systems Review, 38(1), ,2004, pp. 29-34.

[9] W. Ku, C. Chen, and H. Lee, “Weaknesses of Lee-Li-Hwang’s hash-based password authentication scheme”, ACM Operating Systems Review, 37(4), 2003, pp. 9-25.

[10] H. Wen, T. Lee, and T. Hwang, “Provably secure three-party password-based authenticated key exchange protocol using Weil pairing”, IEE Proceedings of Communications, 152(2), 2005, pp. 138-143.

[11] J.K. Lee, S.R. Ryu, and K.Y. Yoo, “Fingerprint-based remote user authentication scheme using smart cards”, Electronics Letters, 38(12), 2002, pp. 554-555.

[12] C.H. Lin and Y.Y. Lai, “A flexible biometrics remote user authentication scheme”, Computer Standards & Interfaces, 27(1), 2004, pp. 19-23.

[13] W.C. Ku, S.T. Chang, and M.H. Chiang, “Further cryptanalysis of fingerprint-based remote user authentication scheme using smart cards”, Electronics Letters, 41(5), 2005, 240-241.

[14] M.K. Khan and J.S. Zhang, “Improving the security of ‘a flexible biometrics remote user authentication scheme’”, Computer Standards & Interfaces, 29, 2007, pp. 82-85.

[15] W.S. Juang, S.T. Chen, and H.T. Liaw, “Robust and Efficient Password-Authenticated Key Agreement Using Smart Cards”, IEEE Transactions on Industrial Electronics, 55(6), 2008, pp. 2551-2556.

[16] W. Kuand and S. Chen, “Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards”, IEEE Transactions Consumer Electronics, 50(1), 2004, pp. 204-207.

[17] R.M. Needham and M.D. Schroeder, “Using encryption for authentication in large networks of computer”, Communication of the ACM, 21(12), 1978, pp. 993-998.