Wireless Pers Commun (2015) 84:2351–2362
DOI 10.1007/s11277-015-2708-4

An Improved Anonymous Multi-Server Authenticated Key Agreement Scheme Using Smart Cards and Biometrics

Hao Lin[1] · Fengtong Wen[1] · Chunxia Du[1]

Published online: 19 May 2015

Abstract Recently, Chuang et al. proposed a multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics. They claimed that their scheme can resist replay attacks, modification attack, off-line password guessing attack and insider attack. However, we demonstrated that their scheme is vulnerable to servers spoofing attack and cannot protect the user's anonymity and the session key, even if the adversary only knows the information transmitting in the public channel. Furthermore, their scheme cannot resist user impersonation attack if the smart cards is stolen. To overcome these problems, we proposed a robust anonymous multi-server authenticated key agreement scheme. We show that our proposed scheme can provide stronger security than previous protocols and protect the user anonymity.

Keywords Multi-server · Authentication · Biometrics · User anonymity

1 Introduction

Nowadays, an increasing number of people gets access to the service of the Internet in the public environment. The opening network poses a threat to the information safety and personal privacy. It is vital for us to establish a secure mechanism of information transmitting to achieve mutual authentication. The multi-server system is made up of three parts

✉ Fengtong Wen
wftwq@163.com

Hao Lin
linhao_ujn@163.com

Chunxia Du
15098823621@163.com

[1] School of Mathematical Science, University of Jinan, Jinan 250022, China

which contains users, servers and registration center. Users need to register to the registration center when they want to acquire the services provided by servers.

Lamport [1] first introduced password authentication with insecure communication in 1981. Following his job, researchers introduced a serious of authentication schemes [2–4] to enhance the security and efficiency. However, many schemes are based on single-server system which cannot be suitable for multi-server system. Then some researchers focused on the fields of multi-server architecture [5–11]. In 2009, Liao and Wang [5] proposed a dynamic ID based remote user authentication scheme. However, Hsiang and Shih [7] demonstrated that their scheme was vulnerable to insider attack, masquerade attack, poor reparability and cannot provide mutual authentication. Then they proposed a secure authentication scheme in the same year. Some scholars [12–14] work on planning more secure and efficient schemes after their achievement.

To resolve the security weaknesses in smart card based password authentication protocol, biometrics have been introduced as another authentication factor in designing authentication schemes. Recently, several biometrics-based remote user authentication schemes [15–17] have been proposed. However, it is unfortunate that most of the existing protocols have been broken shortly after they were proposed. In 2014, Chuang and Chen [18] proposed anonymous multi-server authenticated scheme using smart card and biometrics. They claimed that their scheme can resist many attacks and protect user anonymity. Unfortunately, we pointed out that their protocol cannot withstand servers spoofing attack and impersonation attack. In addition, their protocol cannot protect user anonymity and session key. In order to fix the flaws, we proposed an improved anonymous multi-server authenticated scheme using smart cards and biometrics. Compared with Chuang et al.'s protocol in [18], our protocol can provide stronger security and protect user's anonymity.

Paper Organization A brief review of Chuang et al.'s protocol is provided in Sect. 2. Then we demonstrated that their protocol is susceptible to certain attacks in Sect. 3. In Sect. 4, we propose an improved anonymous multi-server authenticated scheme. We analyze the security of our protocol in Sect. 5. Moreover, we also point out the performance of our protocol in Sect. 6. We will give a conclusion in the last section.

Table 1 Notations

Notation	Meaning
U_i	User
S_j	Authorized server
RC	The registration center
ID_i	The identity of U_i
SID_j	The identity of S_j
x	The secret number only known to RC
PW_i	The password of U_i
BIO_i	The biometrics information of U_i
N_i	A random number
AID_i	The anonymous identity of U_i
PSK	A secure pre-shared key shared among the S_j and RC
$h(\cdot)$	A one-way hash function
$\oplus$	The bitwise XOR operation
$\|\|$	The bitwise concatenation operation

2 Review of Chuang and Chen's Scheme

In this section, we review Chuang and Chen's anonymous multi-server authenticated scheme in [18]. Their scheme is made up of five phases: server registration phase, user registration phase, login phase, authentication phase and password change phase. We list the notations and meanings in Table 1. The login and authentication phase is illustrated in Fig. 1.

2.1 Server Registration Phase

Application servers need to submit information to registration center *RC* when they want to be authorized servers. After that, *RC* sends *PSK* to S_j via the Internet Key Exchange Protocol version 2. These authorized servers will use this key to realize authentication procedure.

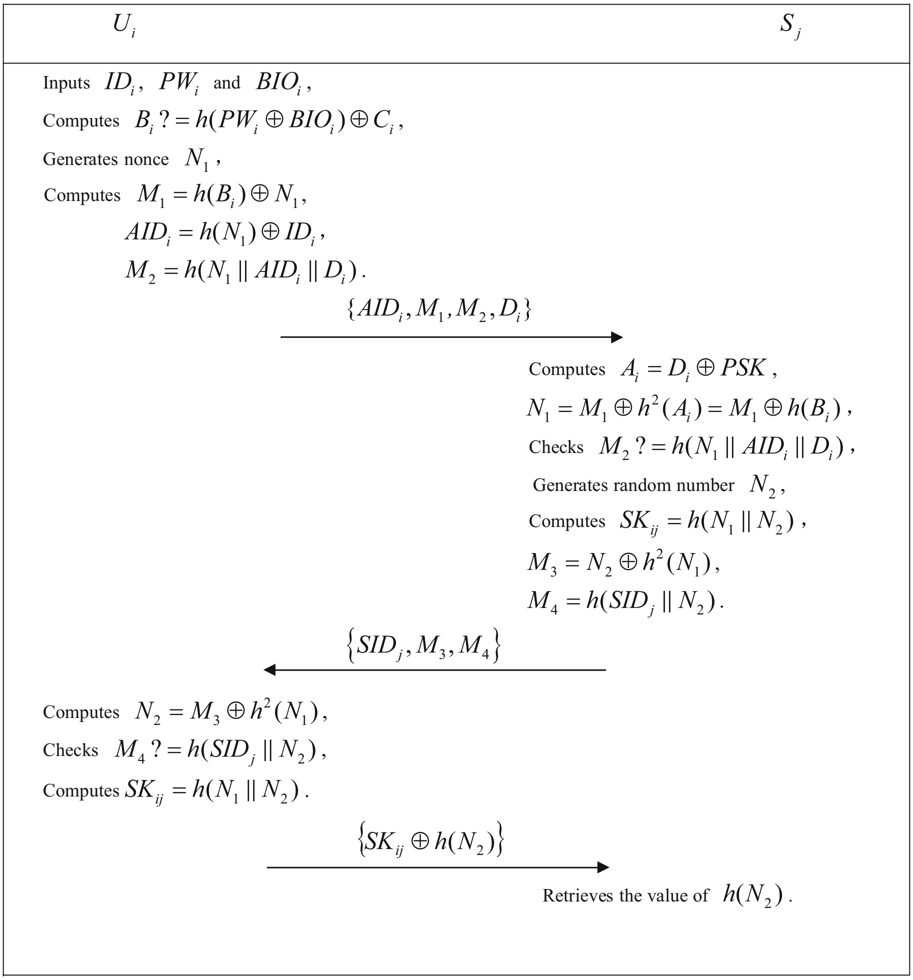

Fig. 1 Login and authentication phase

2.2 User Registration Phase

Step 1: U_i submits ID_i and $h(PW_i \oplus BIO_i)$ to RC through a secure channel.

Step 2: RC computes $A_i = h(ID_i \parallel x)$, $B_i = h^2(ID_i \parallel x) = h(A_i)$, $C_i = h(PW_i \oplus BIO_i) \oplus B_i$, $D_i = PSK \oplus A_i$. Then RC stores information $\{ID_i, B_i, C_i, D_i, h()\}$ in smart card and sends the smart card to U_i through a secure channel.

2.3 Login Phase

Step 1: U_i first inserts his smart card into the device and inputs ID_i and PW_i. Furthermore, he/she will enter his biometric information BIO_i into the sensor.

Step 2: The smart card verifies the ID_i and checks $B_i? = h(PW_i \oplus BIO_i) \oplus C_i$. Then the smart card generates a random number N_1 and computes $M_1 = h(B_i) \oplus N_1$, $AID_i = h(N_1) \oplus ID_i$, $M_2 = h(N_1 \parallel AID_i \parallel D_i)$.

2.4 Authentication Phase

Step 1: The smart card transmits an authentication information $\{AID_i, M_1, M_2, D_i\}$ to an authorized server S_j.

Step 2: The server uses a secure pre-shared key PSK to compute $A_i = D_i \oplus PSK$, $N_1 = M_1 \oplus h^2(A_i)$ and verifies $M_2? = h(N_1 \| AID_i \| D_i)$. If not, the server rejects the request and terminates conversation. Otherwise, the server generates a arbitrary number N_2 and computes the session key $SK_{ij} = h(N_1 \| N_2)$ and the information $M_3 = N_2 \oplus h^2(N_1)$ and $M_4 = h(SID_j \| N_2)$.

Step 3: The server submits the information $\{SID_j, M_3, M_4\}$ to the smart card.

Step 4: After receiving the information, the smart card calculates $N_2 = M_3 \oplus h^2(N_1)$ and verifies whether M_4 is equal to $h(SID_j \| N_2)$. If they are equal, the legitimacy of server is verified by the smart card. Then the smart card will calculate the session key $SK_{ij} = h(N_1 \| N_2)$.

Step 5: The smart card submits the information $SK_{ij} \oplus h(N_2)$ to the server.

Step 6: The server retrieves $h(N_2)$ by using SK_{ij} and verifies the value of $h(N_2)$.

2.5 Password Change Phase

When the user wants to update his password, he/she first inserts his smart card into the device and enters his/her ID_i and PW_i. Then he/she keys BIO_i at the sensor. The smart card will calculate $B_i? = h(PW_i \oplus BIO_i) \oplus C_i$ and check the validity of ID_i. If it does hold, the user can input new password PW_i^*. Otherwise, the smart card rejects request. Then the smart card computes $C_i^* = C_i \oplus h(PW_i \oplus BIO_i) \oplus (PW_i^* \oplus BIO_i)$ and replaces C_i by C_i^*.

3 Security Analysis

The contents of this section are twofold. Firstly, the adversary can obtain the identity of the user U_i only using the information $\{AID_i, M_1, M_2, D_i, SID_i, M_3, M_4, SK_{ij} \oplus h(N_2)\}$

transmitting in the public channel. This also renders that the scheme is insecure against servers spoofing attacks and the session key SK_{ij} will be compromised. Secondly, the adversary can proceed the user impersonation attack if the smart card is compromised [19, 20].

3.1 User Anonymity

Step 1: The attacker selects an identity ID_i^* of U_i and calculates $h(N_1)^* = AID_i \oplus ID_i^*$, $N_2^* = M_3 \oplus h(h(N_1)^*)$, $M_4^* = h(SID_j \| N_2^*)$.

Step 2: The attacker verifies $M_4^* = ?M_4$, if it is true, the adversary gets the right identity of U_i. Otherwise, the adversary repeats the above steps until the correct ID_i is found.

Thus, by launching the above off-line guessing attack, the adversary can successfully recover the user's identity and spoof the server afterwards.

3.2 Servers Spoofing Attack

Step 1: After surmising ID_i of U_i, the adversary computes $h(N_1) = AID_i \oplus ID_i^*$.

Step 2: The adversary selects a random number N_2^* and computes $M_3^* = N_2^* \oplus h^2(N_1)$, $M_4^* = h(SID_j \| N_2^*)$. Then the attacker sends the message $\{SID_j, M_3^*, M_4^*\}$ to the user.

Step 3: On receiving the message $\{SID_j, M_3^*, M_4^*\}$ from S_j, The user computes $N_2' = M_3^* \oplus h^2(N_1) = N_2^*$, $M_4' = h(SID_j \| N_2')$ and verifies $M_4'? = M_4^*$. It is easy to see that $M_4'? = M_4^*$, the adversary is verified by the user.

3.3 Compromise of the Session Key

Step 1: Just like the Sect. 3.2, the attacker can get the value $h(N_1)$.

Step 2: The attacker computes the value $N_2 = M_3 \oplus h^2(N_1)$.

Step 3: Finally, the attacker computes $SK_{ij} = SK_{ij} \oplus h(N_2) \oplus h(N_2)$.

3.4 User Impersonation Attack

If the adversary obtains the information $\{ID_i, B_i, C_i, D_i, h()\}$ stored in the smart card, he/she can proceed the user impersonation attack as follows:

Step 1: The adversary selects an arbitrary number N_1' and computes $M_1' = h(B_i) \oplus N_1'$, $AID_i' = h(N_1') \oplus ID_i$, $M_2' = h(N_1' \| AID_i' \| D_i)$. Then the adversary transmits the message $\{AID_i', M_1', M_2', D_i\}$ to S_j.

Step 2: S_j computes $N_1^* = M_1' \oplus h^2(A_i) = M_1' \oplus h(B_i) = N_1'$ and verifies $M_2^* = h(N_1^* \| AID_i' \| D_i)? = M_2'$.

It is easy to see that the adversary can be verified by the server.

4 Our Improved Scheme

Our scheme is made up of five basic phases: initialization phase, registration phase, login phase, authentication phase and password change phase. Similarly we list the notations and meanings in Table 2. The detailed steps of these phases are described as follows and the login and authentication phase is further illustrated in Fig. 2.

4.1 Initialization Phase

RC computes $r_j = h(SID_j \| x)$ and sends it to the corresponding server S_j, where x is the master secret key of RC. The RC chooses an elliptic curve $E_p(a,b)$, P is a generator with large prime number order, such that the discrete logarithm problem in the cyclic subgroup $<P>$ is hard to be solved.

4.2 Registration Phase

If U_i wants to be a legal user and acquires services from authorized servers, he/she must register to the registration center.

Step 1: U_i submits his/her identity ID_i, $h(PW_i \| BIO_i)$ and $h(ID_i \| BIO_i)$ to RC through a secure communication channel.

Step 2: After receiving the information, RC computes $A_i = h(ID_i \| x)$, $B_{ij} = h(A_i \| r_j)$, $C_{ij} = E_{h(ID_i \| BIO_i)}[B_{ij}]$, $D_i = h(A_i \| h(PW_i \| BIO_i) \| ID_i)$.

Step 3: The registration center RC stores $\{(C_{i1}, C_{i2}, \cdots, C_{ik}), P, A_i, D_i\}$ into the smart card and sends it to the user.

Table 2 Notations

Notation	Meaning
U_i	User
S_j	Authorized server
RC	The registration center
ID_i	The identity of U_i
SID_j	The identity of S_j
x	The master secret key of RC
PW_i	The password of U_i
BIO_i	The biometrics information of U_i
r_j	A secure pre-shared key shared among the S_j and RC
m, n	A random number selected by U_i and S_j
SK_{ij}	Session key shared among the U_i and the S_j
$E_s(\cdot)$	Symmetric key encryption under the key s
$D_s(\cdot)$	Symmetric key decryption under the key s
$h(\cdot)$	A one-way hash function
$\oplus$	The bitwise XOR operation
$\|$	The bitwise concatenation operation

U_i		S_j
Inputs ID_i, PW_i and BIO_i, Computes $D_i^* = h(A_i \| h(PW_i \| BIO_i) \| ID_i)$, Checks $D_i^* ?= D_i$, Generates nonce m, timestamp T_i, Computes $M = m \cdot P$, $D_{h(ID_i \| BIO_i)}[C_{ij}] \rightarrow B_{ij}$, $F_i = A_i \oplus h(M \| T_i \| SID_j)$, $G_i = E_{B_{ij}}[h(PW_i \| BIO_i), M, T_i]$.		
	$\{A_i, F_i, G_i\}$ →	
		Computes $B_{ij} = h(A_i \| r_j)$, $D_{B_{ij}}[G_i] \rightarrow \{h(PW_i \| BIO_i), M, T_i\}$, Generates current timestamp T_j, Checks $\|T_j - T_i\| < \Delta T$, Computes $F_i^* = A_i \oplus h(M \| T_i \| SID_j)$, Checks $F_i^* ?= F_i$, Generates random number n, Computes $J_i = h(SID_j \oplus h(PW_i \| BIO_i))$, $N = n \cdot P$, $K_i = E_{B_{ij}}[J_i, N, SID_j]$ $SK_{ij} = n \cdot M$.
	← $\{K_i\}$	
Computes $D_{B_{ij}}[K_i] \rightarrow \{J_i, N, SID_j\}$, Computes $J_i^* = h(SID_j \oplus h(PW_i \| BIO_i))$, Checks $J_i^* ?= J_i$, Computes $SK_{ij} = m \cdot N$, $L_i = h(SK_{ij} \| h(PW_i \| BIO_i))$.		
	$\{L_i\}$ →	
		Computes $L_i^* = h(SK_{ij} \| h(PW_i \| BIO_i))$, Checks $L_i^* ?= L_i$.

Fig. 2 Login and authentication phase

4.3 Login Phase

Step 1: U_i first inserts his smart card into the device and inputs his/her identity ID_i, password PW_i, and scans his/her biological information BIO_i at the sensor.

Step 2: The smart card calculates $D_i^* = h(A_i \| h(PW_i \| BIO_i) \| ID_i)$ and checks whether D_i^* is equal to D_i. If they are equal, proceeds to Step 3; otherwise, terminates this procedure.

Step 3: The smart card generates a random number m, timestamp T_i and calculates $M = m \cdot P$, $D_{h(ID_i \| BIO_i)}[C_{ij}] \longrightarrow B_{ij}$, $F_i = A_i \oplus h(M \| T_i \| SID_j)$, $G_i = E_{B_{ij}}[h(PW_i \| BIO_i), M, T_i]$. Then the smart card transmits the login request message $\{A_i, F_i, G_i\}$ to the authorized server S_j.

4.4 Authentication Phase

Step 1: Upon receiving the information from the smart card, S_j calculates $B_{ij} = h(A_i \| r_j)$ and decrypts G_i by using B_{ij} to obtain the information $\{h(PW_i \| BIO_i), M, T_i\}$.

Step 2: S_j generates the current time T_j and checks the validity of timestamp T_j by computing $|T_j - T_i| < \triangle T$. If it holds, S_j computes $F_i^* = A_i \oplus h(M \| T_i \| SID_j)$. and checks $F_i^*? = F_i$. If they are equal, the identity of U_i is verified by S_j. Otherwise, S_j terminates this session.

Step 3: S_j generates a random number n and calculates $J_i = h(SID_j \oplus h(PW_i \| BIO_i))$, $N = n \cdot P$, $K_i = E_{B_{ij}}[J_i, N, SID_j]$, the session key $SK_{ij} = n \cdot M$ and submits the reply message $\{K_i\}$ to the smart card.

Step 4: On receiving the reply message from S_j, the smart card can obtain the message $\{J_i, N, SID_j\}$ by using B_{ij} to decrypt K_i. Then the smart card calculates $J_i^* = h(SID_j \oplus h(PW_i \| BIO_i))$ and checks whether J_i^* is equal to J_i. If yes, the legality of S_j is verified by the smart card. Otherwise, terminates the session.

Step 5: The smart card computes the session key $SK_{ij} = m \cdot N$, $L_i = h(SK_{ij} \| h(PW_i \| BIO_i))$ and sends validation messages $\{L_i\}$ to S_j.

Step 6: After receiving the message $\{L_i\}$ from the smart card, S_j computes $L_i^* = h(SK_{ij} \| h(PW_i \| BIO_i))$ and checks whether L_i^* is the same as L_i. If yes, the mutual authentication is finished.

4.5 Password Change Phase

Step 1: The user U_i can update his/her password without the help of authorized server S_j. The user U_i inserts his smart card into the device and keys his identity ID_i^*, password PW_i^* and biological information BIO_i^* in the smart card.

Step 2: The smart card computes $D_i^* = h(A_i \| h(PW_i^* \| BIO_i^*) \| ID_i^*)$ and compares the value of D_i^* with the stored value of D_i. If they are the same, U_i selects a new password PW_i^{new}.

Step 3: The smart card calculates $D_i^{new} = h(A_i \| h(PW_i^{new} \| BIO_i)) \| ID_i)$ and stores D_i^{new} into the smart card to replace D_i.

5 Security Analysis

5.1 User Anonymity

The identity ID_i of users are protected by using the secret key x. The adversary cannot obtain ID_i by equation $A_i = h(ID_i \| x)$ even if he/she extracts the value A_i from the user's smart card.

5.2 Off-Line Password Guessing Attack

If the adversary extracts the secret information ID_i stored in the smart card of the user, he/she wants to derive the password from the equation $D_i^{new} = h(A_i \| h(PW_i \| BIO_i)) \| ID_i)$. However, the password is protected by the secret value of BIO_i, ID_i, and ID_i is protected by the secret key x only known by RC, the attacker can not guess the correct password without knowing the secret key x and biometrics information BIO_i.

5.3 Impersonation Attack

If an attacker attempts to impersonate a legal user to login the server, he/she should forge a valid login request $\{A_i, F_i, G_i\}$ and a corresponding reply message L_i. Even if the adversary knows the information stored in the smart card, he/she can not figure out the valid login message G_i to pass the authentication. Because the attacker does not know the secret message B_{ij} which is protected by the user's secret information ID_i, BIO_i.

5.4 Servers Spoofing Attack

In order to masquerade as the legal server, the attacker should forge a valid reply message according to the user's login request. In our proposed scheme, an attacker cannot forge a valid responding message K_i, because he/she doesn't know the encryption key B_{ij} which is protected by the secret key r_j of the server S_j.

We can conclude that our proposed scheme provides mutual authentication based on the Sects. 3 and 4.

5.5 Replay Attack

In order to protect our scheme from replay attack, we use the timestamp T_i and add a random number into the message. If an attacker wants to replay the previous login message $\{A_i, F_i, G_i\}$ to impersonate a legal user, the server would reject the login request by checking the validity of the timestamp T_i and the message L_i. Because an attacker cannot forge a valid message L_i without knowing the correct random number m.

5.6 Forward Secrecy

The meaning of forward secrecy is that the previous established session key should be safety even if the master secret key is compromised. In our scheme, the session key $SK_{ij} = n \cdot M = m \cdot N$ is computed with the contribution of m and n. The attacker can not compute previous session key due to the intractability of the computation Diffie–Hellman problem.

Table 3 Security comparision

Feature	Li et al.'s [21]	Chuang et al.'s [18]	Our
Servers spoofing attack	No	No	Yes
User anonymity	No	No	Yes
Replay attack	No	Yes	Yes
Impersonation attack	No	No	Yes
Internal attack	No	Yes	Yes

Table 4 Efficiency comparison

Feature	Li et al.'s [21]	Chuang et al.'s [18]	Our
Computation cost of the U_i	$4t_a$	$8t_a$	$5t_a + 2t_b + 3t_c$
Computation cost of the S_j	$3t_a$	$7t_a$	$3t_a + 2t_b + 2t_c$

6 Performance Comparison

In this section, we will compare our proposed authentication and key agreement protocol with two previous related schemes due to Li et al. [21] and Chuang et al. [18]. In Table 3, We evaluate the security of the schemes, while we compare the efficiency in terms of computation in Table 4. We list the implications of notations as follows: T_a: the time complexity of the hash computation; T_b: The time complexity of modular multiplication; T_c: the time complexity of encrypting and decrypting.

From Table 3, it is easy to see that Li et al.'s scheme is vulnerable to many attacks. Their scheme cannot satisfy any kind of the five criterions. The security of Chuang et al.'s scheme in [18] is better than the prior scheme. Nonetheless, Their scheme only satisfies two of five criterions. Our scheme can achieve all the criterions from Table 3. Therefore, our scheme provide stronger security.

From Table 4, we can conclude that Li et al. only used few hash computation to design their authentication protocol. Although their protocol is simple, it is hard for them to ensure security. The time complexity of the hash computation in our protocol is less than Chuang et al.'s protocol, but we employ extra modular multiplication and encryption and decryption. In spite of the slightly higher computation cost than those of Chuang et al.'s scheme in [18] and Lee et al.'s protocol in [21], our scheme achieves stronger security than their solutions, as is shown in Table 3.

7 Conclusion

In this article, we reanalyzed Chuang et al.'s scheme based on multi-server architecture using smart cards and biometrics and claimed that their scheme was vulnerable to servers spoofing attack, even if the attacker didn't know the secret information stored in the user's smart card. Moreover, the attacker can further obtain the user's ID_i and the session key SK_{ij}. If the attacker extracts the information stored in the smart card, he/she can also proceed impersonation attack. Then, we proposed an modified scheme to defuse these attacks and pointed out that our scheme can provide stronger security.

Acknowledgments The authors are grateful to the editor and anonymous reviewers for their valuable suggestions. This work is supported by Natural Science Foundation of Shandong Province (No. ZR2013FM009).

References

1. Lamport, L. (1981). Password authentication with insecure communication. *Communication of ACM*, *24*(11), 770–772.
2. Sun, H. M. (2000). An efficient remote user authentication scheme using smart cards. *IEEE Transactions on Consumer Electronics*, *46*(4), 958–961.
3. Awashti, A. K., & Sunder, L. (2004). An enhanced remote user authentication scheme using smart cards. *IEEE Transactions on on Consumer Electronics*, *50*(2), 583–586.
4. Khan, M. K. (2009). Fingerprint biometric-based self and deniable authentication schemes for the electronic world. *IETE Technical Review*, *26*(3), 191–195.
5. Liao, Y. P., & Wang, S. S. (2009). A secure dynamic ID based remote user authentication scheme for multi-server environment. *Computer Standards and Interfaces*, *31*(1), 24–29.
6. Li, X., Ma, J., Wang, W. D., Xiong, Y. P., & Zhang, J. S. (2013). A novel smart card and dynamic ID based remote user authentication scheme for multi-server environments. *Mathematical and Computer Modelling*, *58*(1–2), 85–95.
7. Hsiang, H. C., & Shih, W. K. (2009). Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment. *Computer Standards and Interfaces*, *31*(6), 1118–1123.
8. Lee, C. C., Lin, T. H., & Chang, R. X. (2011). A secure dynamic ID based remote user authentication scheme for multi-server environment using smart cards. *Expert Systems with Applications*, *38*(11), 13863–13870.
9. Guo, D. L., & Wen, F. T. (2014). Analysis and improvement of a robust smart card based-authentication scheme for multi-server architecture. *Wireless Personal Communications*, *78*(1), 475–490.
10. Wen, F. T., & Li, X. L. (2011). An improved dynamic ID-based remote user authentication with key agreement scheme. *Computers and Electrical Engineering*, *38*(2), 381–387.
11. Wen, F. T., Susilo, W., & Yang, G. M. (2013). A robust smart card based anonymous user authentication protocol for wireless communications. *Security and Communication Networks*, *7*(6), 987–993.
12. Sood, S. K., Sarje, A. K., & Singh, K. (2011). A secure dynamic identity based authentication protocol for multi-server architecture. *Journal of Network and Computer Applications*, *34*(2), 609–618.
13. Li, X., Xiong, Y. P., Ma, J., & Wang, W. D. (2012). An efficient and security dynamic identity based authentication protocol for multi-server architecture using smart cards. *Journal of Network and Computer Applications*, *35*(2), 763–769.
14. Xue, K. P., Hong, P. L., & Ma, C. S. (2014). A lightweight dynamic pseudonym identity based authentication and key agreement protocol without verification tables for multi-server architecture. *Journal of Computer and System Sciences*, *80*(1), 195–206.
15. Khan, M. K., & Zhang, J. (2007). Improving the security of a flexible biometrics remote user authentication scheme. *Computer Standards and Interfaces*, *29*(1), 82–85.
16. Kim, H. S., Lee, J. K., & Yoo, K. Y. (2003). ID-based password authentication scheme using smart cards and fingerprints. *ACM SIGOPS Operating Systems Review*, *37*(4), 32–41.
17. 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.
18. Chuang, M. C., & Chen, M. C. (2014). An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics. *Expert Systems with Applications*, *41*(4), 1411–1418.
19. Kocher, P., Jaffe, J., & Jun, B. (1999). Differential power analysis. *Advances in Cryptology—CRYPTO'99*, *1666*(16), 388–397.
20. Messerges, T. S., Dabbish, E. A., & Sloan, R. H. (2002). Examining smart-card security under the threat of power analysis attacks. *IEEE Trans on Computers*, *51*(5), 541–552.
21. Li, C. T., & Hwang, M. S. (2010). An efficient biometrics-based remote user authentication scheme using smart cards. *Journal of Network and Computer Applications*, *33*(1), 1–5.

Hao Lin received his bachelor degree at University of Jinan. Now he is a master student at school of mathematical sciences, university of Jinan. His main research topics are cryptography and information security.

Fengtong Wen received his Ph.D. degree at Beijing University of Posts and Telecommunications. Now he is a professor of University of Jinan. His main research topics are cryptography and information security. He has published more than 20 research paper.

Chunxia Du received his bachelor degree at college of Heze. Now she is a master student at school of mathematical sciences, university of Jinan. His main research topic is cryptography.