Scientia Iranica D (2012) 19 (6), 1856–1860

Sharif University of Technology

Scientia Iranica

Transactions D: Computer Science & Engineering and Electrical Engineering

www.sciencedirect.com

Research note

Improvement of a security enhanced one-time two-factor authentication and key agreement scheme

Qi Xie

School of Information Science and Engineering, Hangzhou Normal University, Hangzhou 310036, PR China

Received 18 September 2011; revised 21 January 2012; accepted 29 February 2012

KEYWORDS

Authentication;
Key exchange;
Password;
Smart card;
One-time;
Two-factor.

Abstract In 2010, Hölbl et al. showed that Shieh et al.'s mutual authentication and key agreement scheme is vulnerable to the smart card lost attack, not achieving perfect forward secrecy, and proposed a security enhanced scheme to eliminate these weaknesses. In this paper, we show that Hölbl et al.'s security enhancement is still vulnerable to the smart card lost attacks. In addition, their scheme cannot resist impersonation attacks and parallel session attacks. Seeing that the existing mutual authentication schemes using smart cards are almost vulnerable to the smart card lost attacks, we further propose a new one-time two-factor mutual authentication and key agreement scheme to eliminate these weaknesses.

1. Introduction

With the rapid development of all sorts of network, more and more internet users are using various kinds of wireless devices to carry out different kinds of transactions, or obtaining many kinds of services provided by remote servers. However, the user and the server should pass through the mutual authentication in public networks before transactions. Password authentication is one of the simplest and most widely used strategies, because the user only needs to use the short password. On the other hand, because the smart card has many advantages in terms of cost, portability, capacity, efficiency and cryptographic properties, it can also be used to design network security protocols. Based on the merits of passwords and smart cards (two-factor), many two-factor mutual authentication and key agreement schemes have been proposed [1–5].

Generally speaking, two-factor authentication protocols should consist of registration, login and verification phases. The user and server share a secret, which is stored in the user smart card, by combining it with the user password in the registration phase. The user uses his password to extract the shared secret from his smart card, and the shared secret is used to authenticate the user and the server in the login and verification phases. However, since some secret information is stored in the smart cart, adversaries are considered to have the ability to get this information via logical or physical methods [6]. Thus, the adversary can use this information to mount off-line password guessing attacks, and get the user password easily, as the password is supposed to be easy-to-remember, and the password space is small. Therefore, the major challenges in designing two-factor mutual authentication and key agreement schemes are how to resist both off-line password guessing attacks and smart card lost attacks.

Considering the existing one-time two-factor mutual authentication schemes, many are insecure [7]. In 2002, Yeh et al. [8] and Chien et al. [9] proposed a one-time two-factor scheme, respectively. But Tsuji and Shimizu [10] and Ku et al. [11] showed that Yeh et al.'s scheme was vulnerable to stolen-verifier attacks, respectively. In 2005, Lee and Chen [12] proposed an improved scheme to eliminate the weakness of Yeh et al.'s scheme, but their scheme could not resist the smart card lost attacks. In 2004, Ku and Chen [13] showed that Chien et al.'s scheme is vulnerable to insider attacks. Hsu [14] and Lee et al. [15] also showed that Chien et al.'s scheme is vulnerable to parallel session attacks, and proposed improved schemes, respectively. Juang [16] also proposed an improvement of Chien et al.'s scheme. Later, Yoon et al. [17] and Yoon and Yoo [18] demonstrated that Ku et al. and Lee et al.'s improved schemes were still insecure. In 2007, Wang et al. [19] showed that both Ku et al. scheme and Yoon et al. scheme [17] were still vulnerable to password guessing attacks, forgery attacks and denial of service attacks, and proposed a further improvement

E-mail address: qixie68@yahoo.com.cn.

Peer review under responsibility of Sharif University of Technology.

Production and hosting by Elsevier

1026-3098 © 2012 Sharif University of Technology. Production and hosting by Elsevier B.V. Open access under CC BY-NC-ND license.
doi:10.1016/j.scient.2012.02.029

scheme. But, Chung et al. [20] showed that Wang et al.'s scheme was vulnerable to impersonation attacks and smart card lost attacks, which are not easily reparable and do not provide perfect forward secrecy. In 2011, Chen et al. [21] showed that Wang et al.'s improved scheme is vulnerable to the impersonation attack and parallel session attack, and proposed an improvement scheme. However, their improved scheme is still insecure, e.g. vulnerable to the smart card lost attacks. Shieh and Wang [22] pointed out the weakness of Juang's scheme [16] and proposed an improved scheme. In 2010, Hölbl et al. [23] showed that Shieh et al.'s scheme is vulnerable to the smart card lost attack and does not achieve perfect forward secrecy. Further, they proposed a security enhanced scheme. In 2009, Xu et al. [24] proposed a new one-time authentication scheme with security proof, but Song [25] showed that their scheme was vulnerable to impersonation attacks, and proposed an improved scheme. Unfortunately, Song's scheme cannot resist the smart card lost attack, and does not provide the perfect forward secrecy.

In this paper, we show that Hölbl et al.'s security enhancement is still vulnerable to the smart card lost attack, and their scheme cannot resist impersonation attacks and parallel session attacks. Then, we further propose a new one-time two-factor mutual authentication and key agreement scheme to eliminate their weaknesses.

The rest of the paper is organized as follows. In Section 2, we review Hölbl et al.'s scheme. In Section 3, some attacks against their scheme are described, and in Sections 4 and 5 we propose a new scheme and analyze its security. After that, the security and performance comparisons are presented in Section 6, and we conclude the paper in Section 7.

2. Review of Hölbl et al.'s scheme

The notations used throughout this paper are summarized as follows:

- p, q: two large prime numbers, such as $q|p-1$;
- g: a generator with order q of the group $GF(p)$;
- U: the user;
- ID: U's identity;
- PW: U's password;
- S: the server;
- $(X, Y = g^X \bmod p)$: S's private–public key pair;
- $h()$: a secure cryptographic hash function;
- $\|$: string concatenation operation;
- $\oplus$: bitwise XOR operation.

Hölbl et al.'s scheme consists of two phases: registration, and login and key agreement phases.

2.1. Registration phase

U and S carry out the following steps during the user registration phase.

Step 1: U chooses his identity ID, password PW and a random number b, and computes $h(b \oplus PW)$. He submits his identity ID and $h(b \oplus PW)$ to S through a secure channel.

Step 2: S computes $R = h(ID \oplus X) \oplus h(b \oplus PW)$, and sends a smart card to U via a secure channel, where the smart card contains R and $h()$.

Step 3: U enters b to the smart card.

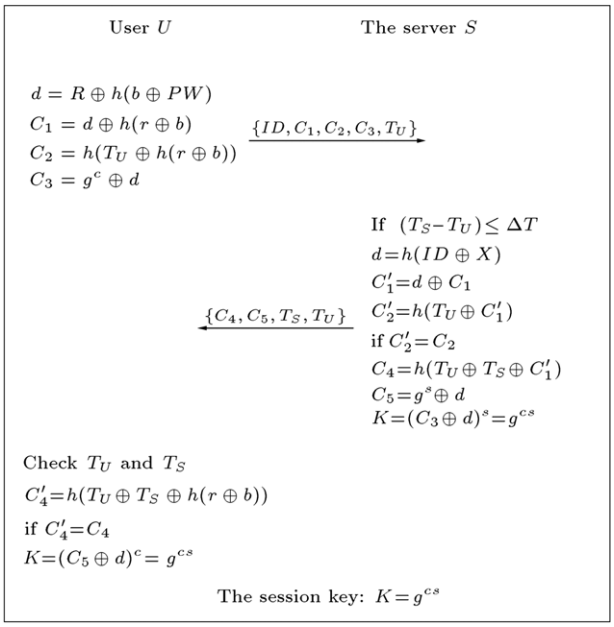

Figure 1: Hölbl et al.'s authentication scheme.

2.2. Login and key agreement phase

When U is about to logon to the remote server S, U completes the following operations:

Step 1: U inserts his smart card into the card reader of a terminal, and keys ID and PW.

Step 2: The smart card generates a random number, r, a large random integer, $c < p - 1$, and computes $d = R \oplus h(b \oplus PW)$, $C_1 = d \oplus h(r \oplus b)$, $C_2 = h(T_U \oplus h(r \oplus b))$, $C_3 = g^c \oplus d$, where T_U is the current timestamp. Then, the smart card sends $\{ID, C_1, C_2, C_3, T_U\}$ to S.

Upon receiving the message $\{ID, C_1, C_2, C_3, T_U\}$, S completes the following operations:

Step 3: S checks if $(T_S - T_U) \leq \Delta T$, where T_S is S's current timestamp and ΔT is the expected valid time interval for transmission. If not, S rejects U's login request. Otherwise, S computes $d = h(ID \oplus X)$, $C_1' = d \oplus C_1$, $C_2' = h(T_U \oplus C_1')$, and checks if $C_2' = C_2$. If not, S rejects U's login request. Otherwise, U is authenticated by S.

Step 4: S chooses a large random integer, $s < p - 1$, and computes $C_4 = h(T_U \oplus T_S \oplus C_1')$, $C_5 = g^s \oplus d$, the session key, $K = (C_3 \oplus d)^s = g^{cs}$, and returns $\{C_4, C_5, T_S, T_U\}$ to U.

After receiving the message, $\{C_4, C_5, T_S, T_U\}$, the smart card checks the validities of T_S and T_U. If not, it is terminated. Otherwise, the smart card computes $C_4' = h(T_U \oplus T_S \oplus h(r \oplus b))$, and compares it with the received C_4. If they are equal, S is authenticated by U. Then, the smart card computes the session key, $K = (C_5 \oplus d)^c = g^{cs}$.

Thus, S and U share the session key, K, for subsequent private communications. Figure 1 illustrates Hölbl et al.'s authentication protocol.

3. Cryptanalysis of Hölbl et al.'s scheme

Hölbl et al. claimed that their scheme is robust and secure against impersonation, parallel session and smart card lost attacks. However, in this section, we show that Hölbl et al.'s scheme does not provide these security requirements. The details are as follows.

3.1. Parallel session attack

In Hölbl et al.'s scheme, when U and S perform the login and key agreement phase, an adversary can get $\{ID, C_1, C_2, C_3, T_U\}$ and $\{C_4, C_5, T_S, T_U\}$ from the public network. Then, the adversary can impersonate the user, U, and send $\{ID, \overline{C_1}, \overline{C_2}, C_3, \overline{T_U}\}$ to S as a new session where $\overline{C_1} = C_1 \oplus T_U = d \oplus h(r \oplus b) \oplus T_U$, $\overline{C_2} = C_4 = h(T_U \oplus T_S \oplus h(r \oplus b))$, $\overline{T_U} = T_S$.

After receiving $\{ID, \overline{C_1}, \overline{C_2}, C_3, \overline{T_U}\}$, S can check the validity of $\overline{T_U}$, and can compute:

$$C_1' = d \oplus \overline{C_1} = d \oplus d \oplus h(r \oplus b) \oplus T_U = h(r \oplus b) \oplus T_U,$$
$$C_2' = h(\overline{T_U} \oplus C_1') = h(T_S \oplus h(r \oplus b) \oplus T_U).$$

Because $C_2' = \overline{C_2}$, the adversary can pass through the authentication of S.

The reason why Hölbl et al.'s scheme suffers from this attack is that S uses bitwise XOR operations among $h(r \oplus b)$, T_S and T_U to compute $C_4 = h(T_U \oplus T_S \oplus h(r \oplus b))$. In order to overcome this weakness, a possible countermeasure is that S replace C_4 with $C_4 = h(T_U \parallel T_S \parallel h(r \oplus b))$.

3.2. Impersonation attack

In Hölbl et al.'s scheme, if an adversary gets $\{ID, C_1, C_2, C_3, T_U\}$ then, he can impersonate the user, U, to pass through the authentication of S. The details are as follows:

The adversary computes $T = T_U \oplus \overline{T_U}$, $\overline{C_1} = C_1 \oplus T = d \oplus h(r \oplus b) \oplus T$ where $\overline{T_U}$ is the current timestamp. Then, he sends $\{ID, \overline{C_1}, C_2, C_3, \overline{T_U}\}$ to S.

Upon receiving $\{ID, \overline{C_1}, C_2, C_3, \overline{T_U}\}$, S checks the validity of $\overline{T_U}$, and computes:

$$d = h(ID \oplus X),$$

$C_1' = d \oplus \overline{C_1} = d \oplus d \oplus h(r \oplus b) \oplus T = h(r \oplus b) \oplus T$, $C_2' = h(\overline{T_U} \oplus C_1') = h(\overline{T_U} \oplus h(r \oplus b) \oplus T) = h(\overline{T_U} \oplus h(r \oplus b) \oplus T_U \oplus \overline{T_U}) = h(h(r \oplus b) \oplus T_U)$ and checks whether $C_2 = h(T_U \oplus h(r \oplus b)) = C_2'$ or not. Obviously, S may accept the adversary's login request.

The reason why Hölbl et al.'s scheme suffers from this attack is that the smart card uses a bitwise XOR operation between $h(r \oplus b)$ and T_U to compute $C_2 = h(T_U \oplus h(r \oplus b))$. In order to overcome this weakness, a possible countermeasure is for the smart card to replace C_2 with $C_2 = h(T_U \parallel h(r \oplus b))$.

3.3. Smart card lost attack and off-line password guessing attack

If an adversary obtains R and b, which are stored in U's smart card, where $R = d \oplus h(b \oplus PW)$, and gets U's login message, $\{ID, C_1, C_2, C_3, T_U\}$, an adversary can launch the off-line password guessing attack.

Because $C_1 = d \oplus h(r \oplus b)$ and $C_2 = h(T_U \oplus h(r \oplus b))$, an adversary can compute $d' = R \oplus h(b \oplus PW')$, $C_1' = C_1 \oplus d'$, and verify whether $C_2 = h(T_U \oplus C_1')$ holds or not, where PW' is a guessed password. If it holds, the guessed password is U's password, otherwise the adversary tries another password, and so on. Finally, the adversary can get the correct password, as the password is short and human memorizable.

The reason why Hölbl et al.'s scheme suffers from this attack is that the user's smart card contains R and b, and U's login message includes the authenticator, C_2, which only depends on a password, thus the adversary can launch an off-line password guessing attack. In order to overcome this weakness, a possible countermeasure is that authenticator C_2 should depend on both the password and a random nonce.

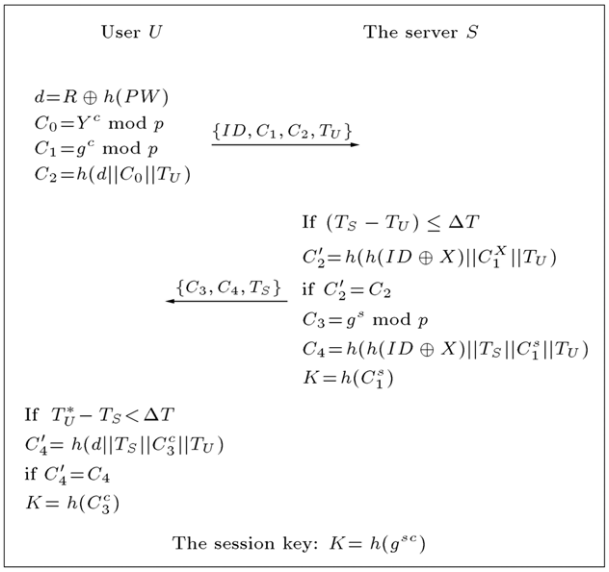

Figure 2: The proposed authentication scheme.

4. The proposed scheme

In this section, we will propose a new scheme with strong security. The basic ideas of our constructions are as follows:

(1) In order to resist the smart card lost attack and off-line password guessing attack, each authenticator in the login and key agreement phase should depend on both the password and a random nonce, and the Diffie–Hellman value can be regarded as a random nonce.
(2) To resist parallel session and impersonation attacks, the login and response messages should contain a string concatenation operation.

The details of the proposed scheme are as follows.

4.1. Registration phase

U and S carry out the following steps during the user registration phase.

Step 1: U chooses a password, PW, and his identity ID, then, submits ID to S.

Step 2: S computes $d = h(ID \oplus X)$, and sends a smart card to U via a secure channel, where the smart card contains ID, d and $h()$.

Step 3: U computes $R = d \oplus h(PW)$, and replaces d with R. That is, the smart card contains ID, $h()$ and $R = h(ID \oplus X) \oplus h(PW)$.

4.2. Login and key agreement phase

When U is about to logon to the remote server, S, U completes the following operations (Figure 2 illustrates our protocol).

Step 1: U inserts his smart card into the card reader of a terminal, and keys PW.

Step 2: The smart card generates a random number, $c < p - 1$, and computes:

$$d = R \oplus h(PW) = h(ID \oplus X), \qquad C_0 = Y^c \bmod p,$$
$$C_1 = g^c \bmod p, \qquad C_2 = h(d \parallel C_0 \parallel T_U),$$

where T_U is the current timestamp. Then, the smart card sends $\{ID, C_1, C_2, T_U\}$ to S.

Upon receiving the message, $\{ID, C_1, C_2, T_U\}$, S completes the following operations:

Step 3: S checks if $(T_S - T_U) \leq \Delta T$, where T_S is S's current timestamp and ΔT is the expected valid time interval for transmission. If not, S rejects U's login request. Otherwise, S computes $C_2' = h(h(ID \oplus X) \parallel C_1^X \parallel T_U)$, and checks if $C_2' = C_2$. If not, S rejects U's login request. Otherwise, U is authenticated by S.

Step 4: S chooses a large random integer, $s < p - 1$, computes $C_3 = g^s \bmod p$, ${C_1}^s$, $C_4 = h(h(ID \oplus X) \parallel T_S \parallel {C_1}^s \parallel T_U)$, and the session key, $K = h({C_1}^s)$, and returns $\{C_3, C_4, T_S\}$ to U.

After receiving the message, $\{C_3, C_4, T_S\}$, the smart card checks the validity of T_S by ${T_U}^* - T_S < \Delta T$, where ${T_U}^*$ is the current time. If not, it is terminated. Otherwise, the smart card computes $C_4' = h(d \parallel T_S \parallel C_3^c \parallel T_U)$, and compares it with the received C_4. If they are equal, S is authenticated by U. Then, the smart card computes the session key, $K = h({C_3}^c)$.

Thus, S and U share the session key, K, for subsequent private communications.

5. Security analysis

In this section, we will show that the proposed scheme is secure.

5.1. Offline password guessing attacks and smart card lost attacks

If an adversary can get all transmitted messages, $\{ID, C_1, C_2, T_U\}$ and $\{C_3, C_4, T_S\}$, between U and S, and $R = h(ID \oplus X) \oplus h(PW)$, which is stored in U's smart card, then, he launches an off-line password guessing attack as follows:

The adversary may compute $d' = R \oplus h(PW')$ where PW' is a trial password, but he cannot compute $C_2' = h(d' \parallel C_0 \parallel T_U)$ and cannot verify if $C_2' = C_2$ holds or not, since he is unable to compute the Diffie–Hellman value, $C_0 = Y^c = g^{cX}$, from C_1 and Y, due to the intractability of the Computational Diffie–Hellman (CDH) problem. The same reason why the adversary is unable to compute ${C_1}^s$ and cannot verify whether his trial password, PW', is correct or not.

5.2. Forgery attack or impersonation attack

An adversary can get $R = h(ID \oplus X) \oplus h(PW)$, which is stored in U's smart card, and launch forgery or impersonation attacks. He chooses a random number, $c < p - 1$, and computes

$$d' = R \oplus h(PW') = h(ID \oplus X), \qquad C_0 = Y^c \bmod p,$$
$$C_1 = g^c \bmod p, \qquad C_2 = h(d' \parallel C_0 \parallel T_U)$$

and sends $\{ID, C_1, C_2, T_U\}$ to S, where PW' is a trial password. However, the adversary's login messages cannot pass the verification process of S due to $d' = R \oplus h(PW') \neq h(ID \oplus X)$.

5.3. Parallel session attack

In the proposed scheme, an adversary wants to launch a parallel session attack. However, it is infeasible, because two authenticators, $C_2 = h(d \parallel C_0 \parallel T_U)$ and $C_4 = h(h(ID \oplus X) \parallel T_S \parallel {C_1}^s \parallel T_U)$, contain different Diffie–Hellman values, $C_0 = g^{cX}$ and ${C_1}^s = g^{sc}$. The adversary's login request as a new session run will not be accepted by S because his login request cannot pass the verification process of S.

5.4. Perfect forward secrecy

In the improved scheme, the session key is $K = h(g^{sc})$, where c and s are nonces chosen by U and S, respectively. Even if an adversary gets U's password, PW, and S's secret key, X, he is not able to compute g^{sc}, that is he is not able to compute the session key of any previously established session, due to the intractability of the CDH problem.

Table 1: Security properties comparison.

	P1	P2	P3	P4	P5	P6	P7	P8
Wang et al. [19]	*x*	*x*	*x*	*y*	*y*	*x*	*y*	*y*
Chen et al. [21]	*y*	*y*	*x*	*y*	*x*	*x*	*x*	*y*
Hölbl et al. [23]	*x*	*x*	*x*	*y*	*y*	*y*	*y*	*y*
Xu et al. [24]	*y*	*x*	*y*	*y*	*y*	*y*	*y*	*y*
Song [25]	*y*	*y*	*x*	*y*	*y*	*x*	*y*	*y*
The proposed scheme	*y*	*y*	*y*	*y*	*y*	*y*	*y*	*y*

P1: parallel session attack;
P2: forgery attack or impersonation attack;
P3: smart card lost attack;
P4: password guessing attack;
P5: known-key attack;
P6: perfect forward secrecy;
P7: Denning–Sacco attack;
P8: replay attack;
y: scheme can resist such attack;
x: scheme cannot resist such attack.

5.5. Replay attack

Suppose that an adversary impersonates U and replays U's $\{ID, C_1, C_2, T_U\}$ to S, his login request will be rejected due to the invalid timestamp, T_U. If he revises T_U to the valid T_U', his login request still cannot pass the verification process of S, since he cannot generate the valid C_2 corresponding to the new T_U'.

For the same reason, if an adversary impersonates S and replays S's $\{C_3, C_4, T_S\}$, he still cannot succeed.

5.6. Denning–Sacco attack

Even if an adversary gets the session key, $K = h(g^{sc})$, he cannot get g^{sc} due to the intractability under the assumption of a one-way collision resistant cryptographic hash function. Therefore, the adversary cannot get U's password, PW, S's secret key, X and $d = h(ID \oplus X)$.

5.7. Known-key security

Even if an adversary gets U's password, PW, and S's secret key, X, he still cannot compute the session key due to the intractability of the CDH problem. Due to the randomness and independence of the generations of c and s in all sessions, the session key, $K = h(g^{sc})$, of each session is independent of that of any other sessions. Therefore, an adversary is unable to compute the previous and future session keys from one session key.

6. Security and performance comparisons

Security properties and performance cost comparisons between our scheme and the other five related schemes in [19,21,23–25] are given in Tables 1 and 2.

Let T_e, T_h, T_d and T_a be the time for performing a modular exponentiation, a one-way hash function, a symmetric encryption/decryption and an exclusive OR operation, respectively. We ignore modular addition and string concatenation operations that are negligible compared to others. In the user registration of our scheme, it requires two hash and two exclusive OR operations. In the login and key agreement phase, it requires two hash, seven exclusive OR and six modular exponentiation operations.

Table 2: Performance comparison.

	Cost of login and key agreement phase
Wang et al. [19]	$10T_a + 8T_h$
Chen et al. [21]	$8T_a + 8T_h$
Hölbl et al. [23]	$17T_a + 7T_h + 4T_e$
Xu et al. [24]	$9T_h + 6T_e$
Song [25]	$3T_a + 8T_h + 2T_d + T_e$
The proposed scheme	$2T_a + 7T_h + 6T_e$

In performance comparison, we mainly focus on computations of the login and key agreement phase, since it is the main body of an authentication scheme, and the registration phase only performs one time before authentication. Both Wang et al.'s [19] and Chen et al.'s [21] schemes are more efficient, because they are completely based on hash and exclusive OR operations, but do not provide perfect forward secrecy. In order to achieve perfect forward secrecy, the designers always use the Diffie–Hellman key exchange technique to establish session keys. The proposed scheme needs a few additional modular exponentiation operations compared to others, but is more secure.

7. Conclusion

We have shown that Hölbl et al.'s security enhancement is still vulnerable to the smart card lost attacks, and cannot resist impersonation and parallel session attacks. To eliminate these weaknesses, we propose a secure one-time two-factor mutual authentication and key agreement scheme, which keeps the merits of Hölbl et al.'s scheme.

Acknowledgments

The author would like to thank the anonymous referees for their constructive comments. This research was supported by the National Natural Science Foundation of China (No. 61070153), Natural Science Foundation of Zhejiang province (No. LZ12F02005), and Opening Fund of Top Key Discipline of Computer Software and Theory in Zhejiang Provincial Colleges at Zhejiang Normal University (No. ZSDZZZZXK35).

References

[1] Chang, C.C. and Wu, T.C. "Remote password authentication with smart cards", *IEE Proceedings-E Computers and Digital Techniques*, 138(3), pp. 165–168 (1991).
[2] Hwang, M.S. and Li, L.H. "A new remote user authentication scheme using smart cards", *IEEE Transactions on Consumer Electronics*, 46(1), pp. 28–30 (2000).
[3] Sun, H.M. "An efficient remote user authentication scheme using smart cards", *IEEE Transactions on Consumer Electronics*, 46(4), pp. 958–961 (2000).
[4] Awasthi, A.K. and Lal, S. "A remote user authentication scheme using smart cards with forward secrecy", *IEEE Transactions on Consumer Electronics*, 49(4), pp. 1246–1248 (2003).
[5] Yeh, K.H., Sub, C.H., Loa, N.W., Li, Y. and Hung, Y.X. "Two robust remote user authentication protocols using smart cards", *The Journal of Systems and Software*, 83(12), pp. 2556–2565 (2010).
[6] Witteman, M. "Advances in smartcard security", *Information Security Bulletin*, 7(2002), pp. 11–22 (2002).
[7] Nose, P. "Security weaknesses of authenticated key agreement protocols", *Information Processing Letters*, 111(14), pp. 687–696 (2011).
[8] Yeh, T.C., Shen, H.Y. and Hwang, J.J. "A secure one-time password authentication scheme using smart cards", *IEICE Transaction on Communication*, E85-B(11), pp. 2515–2518 (2002).
[9] Chien, H.Y., Jan, J.K. and Tseng, Y.M. "An efficient and practical solution to remote authentication smart card", *Computers & Security*, 21(4), pp. 372–375 (2002).
[10] Tsuji, T. and Shimizu, A. "One-time password authentication protocol against theft attacks", *IEICE Transactions on Communications*, E87-B(3), pp. 523–529 (2004).
[11] Ku, W.C., Tsai, H.C. and Tsaur, M.J. "Stolen-verifier attack on an efficient smartcard based one-time password authentication scheme", *IEICE Transactions on Communications*, E87-B(8), pp. 2374–2376 (2004).
[12] Lee, N.Y. and Chen, J.C. "Improvement of one-time password authentication scheme using smart card", *IEIC E Transaction on Communications*, E88-B(9), pp. 3765–3769 (2005).
[13] Ku, W.C. and Chen, S.M. "Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards", *IEEE Transactions on Consumer Electronics*, 50(1), pp. 204–207 (2004).
[14] Hsu, C.L. "Security of Chien et al.'s remote user authentication scheme using smart cards", *Computer Standards & Interfaces*, 26(3), pp. 167–169 (2004).
[15] Lee, S., Kim, H. and Yoo, K. "Improvement of Chien et al.'s remote user authentication scheme using smart cards", *Computer Standards & Interfaces*, 27(2), pp. 181–183 (2005).
[16] Juang, W.S. "Efficient password authenticated key agreement using smart cards", *Computers & Security*, 23(4), pp. 167–173 (2004).
[17] Yoon, E.J., Ryu, E.K. and Yoo, K.Y. "Further improvement of an efficient password based remote user authentication scheme using smart cards", *IEEE Transactions on Consumer Electronics*, 50(2), pp. 612–614 (2004).
[18] Yoon, E. and Yoo, K. "More efficient and secure remote user authentication scheme using smart cards", *Proceedings of 11th International Conference on Parallel and Distributed System*, Fukuoka, Japan, pp. 73–77 (2005).
[19] Wang, X.M., Zhang, W.F., Zhang, J.S. and Khan, M.K. "Cryptanalysis and improvement on two efficient remote user authentication scheme using smart cards", *Computer Standards & Interfaces*, 29(5), pp. 507–512 (2007).
[20] Chung, H.R., Ku, W.C. and Tsaur, M.J. "Weaknesses and improvement of Wang et al.'s remote user password authentication scheme for resource-limited environments", *Computer Standards & Interfaces*, 31(4), pp. 863–868 (2009).
[21] Chen, T.H., Hsiang, H.C. and Shih, W.K. "Security enhancement on an improvement on two remote user authentication schemes using smart cards", *Future Generation Computer Systems*, 27(4), pp. 377–380 (2011).
[22] Shieh, W.G. and Wang, F.M. "Efficient remote mutual authentication and key agreement", *Computers & Security*, 25(1), pp. 72–77 (2006).
[23] Hölbl, M., Welzer, T. and Brumen, B. "Attacks and improvement of an efficient remote mutual authentication and key agreement scheme", *Cryptologia*, 34(1), pp. 52–59 (2010).
[24] Xu, J., Zhu, W. and Feng, D. "An improved smart card based password authentication scheme with provable security", *Computer Standards & Interfaces*, 31(4), pp. 723–728 (2009).
[25] Song, R. "Advanced smart card based password authentication protocol", *Computer Standards & Interfaces*, 32(5–6), pp. 321–325 (2010).

Qi Xie received his Ph.D. degree in Applied Mathematics from Zhejiang University, Hangzhou, PR China, in 2005. He was a postdoctoral researcher in the College of Computer Science at Zhejiang University from 2007 to 2010, and an academic visitor at the School of Computer Science at Birmingham University, UK, from Sept. 2009–Sept. 2010. Currently, he is Professor at the School of Information Science and Engineering at Hangzhou Normal University, PR China. His research interests include digital signatures, authentication and key exchange protocols, etc.