Cryptanalysis and Improvement of an ECC-Based Password Authentication Scheme Using Smart Cards

Cheng-Chi Lee[1,2], Chun-Ta Li[3,⋆], Chi-Yao Weng[4], Jian-Jhong Jheng[3],
Xiao-Qian Zhang[3], and Yi-Rui Zhu[3]

[1] Department of Library and Information Science, Fu Jen Catholic University
510 Jhongjheng Road, New Taipei City 24205, Taiwan (R.O.C.)
cclee@mail.fju.edu.tw

[2] Department of Photonics and Communication Engineering, Asia University
500 Lioufeng Road, Taichung City 41354, Taiwan (R.O.C.)

[3] Department of Information Management, Tainan University of Technology
529 Zhongzheng Road, Tainan City 71002, Taiwan (R.O.C.)
th0040@mail.tut.edu.tw

[4] Department of Computer Science, National Tsing Hua University
101 Kuang-Fu Road, Hsinchu City 30013, TAIWAN (R.O.C.)
cyweng@is.cs.nthu.edu.tw

Abstract. Remote password authentication has been widely used in network systems and it aims to provide secure remote access control. In 2013, Li proposed a novel password authentication scheme based on elliptic curve cryptography and smart card [17]. However, we found that Li's authentication scheme has a serious security problem in that all registered users' sensitive passwords can be easily derived by the privileged-insider of remote server. Therefore, in this paper, we propose a slight modification on Li's scheme to prevent the shortcomings. Our improved scheme not only inherits the advantages of Li's password authentication scheme but also remedies the serious security weakness of not being able to withstand insider attack.

Keywords: Cryptanalysis, Elliptic curve cryptography (ECC), Password authentication, Insider attack, Smart card.

1 Introduction

With the rapid development of network technologies and Internet users, more and more remote services such as online transactions, online games and online electronic contents etc. are supported through Internet, which provides conveniences to Internet users. However, with the increase of network attacks such as message eavesdropping, participant masquerading and secret guessing, network security and information privacy become an important research issue in networking environments. Remote user authentication is crucial to prevent unregistered

[⋆] Corresponding author.

G. Wang et al. (Eds.): CSS 2013, LNCS 8300, pp. 338–348, 2013.

users from accessing remote service systems and password authentication with smart card is one of the most popular mechanisms to verify the validity of login user.

Recently, many smart card based password authentication schemes for remote login systems have been proposed [1–3, 5–16, 18–24, 26]. Particularly, in 2013, Islam and Biswas proposed an ECC-based password authentication and key agreement scheme using smart card [4] in remote login environments. Unfortunately, in the same year, Li [17] pointed out that Islam and Biswas's scheme cannot resist off-line password guessing attack, stolen-verifier attack and insider attack. Li also proposed a slightly modified version of Islam and Biswas's authentication scheme so as to remedy the identified deficiencies. However, in this paper, we find that both original and improved schemes [4, 17] are vulnerable to the insider attack. The spotted security flaw may allow a privileged-insider of remote server to derive the passwords of all login users registered with the remote server. In order to resist security flaw, we would like to propose an improved scheme that also inherits the advantages of Li's password authentication scheme and resistance to the insider attack.

The remainder of the paper is organized as follows. Section 2 is a brief review of Li's ECC-based password authentication scheme and a cryptanalysis of Li's scheme is given in Section 3. Our improved scheme against insider attack is proposed in Section 4. Security analysis of our improved scheme is presented in Section 5 and Section 6 concludes this paper.

2 A Review of Li's Password Authentication Scheme

In this section, we review Li's password-based remote authentication scheme [17] and Li's scheme is composed of five phases, registration, password authentication, password change, session key distribution and user eviction. For convenience of description, terminology and notations used in the paper are summarized as follows:

- (ID_A, pw_A): The identity and password of the client A.
- S: The remote server.
- d_s: The secret key, which is kept secret and only known by S.
- U_S: The public key of S, where $U_S = d_s \cdot G$
- U_A: Password-verifier of the client A, where $U_A = pw_a \cdot G$.
- G: Bases point of the elliptic curve group of order n such that $n \cdot G = O$, where n is a large prime number.
- K_x: Secret key computed either using $K = pw_A \cdot U_S = (K_x, K_y)$ or $K = d_s \cdot U_A = (K_x, K_y)$.
- $H(\cdot)$: A collision free one-way hash function.
- $E_{K_x}(.)/D_{K_x}(.)$: The symmetric encryption/decryption function with key K_x.
- r_i: A random number chosen by the entity i from $[1, n-1]$.
- $+/-$: Elliptic curve point addition/subtraction.

2.1 Registration Phase

Step 1. $A \longrightarrow S$**:** ID_A, U_A
Client A computes $U_A = pw_A \cdot r_A \cdot G$ and sends it with ID_A to S through a secure channel, where r_A is a random number which is kept secret and only known by A.

Step 2. $S \longrightarrow A$**: SMART CARD**
S stores A's ID_A, U_A and a $status - bit$ in a write protected file as depicted in Table 1. Moreover, S writes $\{G, U_S, H(\cdot), E_K(.)/D_K(.)\}$ into a personal smart card and issues it to A through a secure channel.

Step 3. A writes r_A into the smart card. Finally, the smart card includes $\{r_A, G, U_S, H(\cdot), E_K(.)/D_K(.)\}$.

Table 1. The verifier table of S after finishing the registration phase [17]

Identity	Password-verifier	$Status - bit$
⋮	⋮	⋮
ID_A	$U_A = pw_A \cdot r_A \cdot G$	0/1
⋮	⋮	⋮

2.2 Password Authentication Phase

Client A inserts the personal smart card into the card reader and keys ID_A and pw_A. Then the smart card will perform the following operations:

Step 1. $A \longrightarrow S$**:** $ID_A, E_{K_x}(ID_A, R_A, W_A, U'_A)$
The smart card retrieves r_A, generates a random number r'_A, computes $R_A = r_A \cdot U_S = r_A \cdot d_S \cdot G$, $W_A = r_A \cdot r_A \cdot pw_A \cdot G$, $U'_A = pw_A \cdot r'_A \cdot G$ and $E_{K_x}(ID_A, R_A, W_A, U'_A)$ and sends $E_{K_x}(ID_A, R_A, W_A, U'_A)$ with ID_A to S, where the encryption key K_x is the x coordinate of $K = pw_A \cdot r_A \cdot U_S = pw_A \cdot r_A \cdot d_S \cdot G = (K_x, K_y)$.

Step 2. $S \longrightarrow A$**:** $(W_A + W_S), H(W_S, U'_A)$
Upon receiving the login message, S computes the decryption key $K = d_S \cdot U_A = pw_A \cdot r_A \cdot d_S \cdot G = (K_x, K_y)$ and decrypts $E_{K_x}(ID_A, R_A, W_A, U'_A)$ to reveal (ID_A, R_A, W_A, U'_A). S verifies the decrypted ID_A with received ID_A and $\hat{e}(R_A, U_A)$ with $\hat{e}(W_A, U_S)$. If they hold, S sends $\{(W_A + W_S), H(W_S, U'_A)\}$ to A, where r_S is a random number which is generated by S and $W_S = r_S \cdot U_S$. A bilinear pairing is used to assure the correctness of the scheme and is given below:

$$\begin{aligned}
\hat{e}(R_A, U_A) &= \hat{e}(r_A \cdot d_S \cdot G, r_A \cdot pw_A \cdot G) \\
&= \hat{e}(G, G)^{r_A \cdot r_A \cdot pw_A \cdot d_S} \\
&= \hat{e}(r_A \cdot r_A \cdot pw_A \cdot G, d_S \cdot G) \\
&= \hat{e}(W_A, U_S).
\end{aligned}$$

Step 3. $A \longrightarrow S$: $ID_A, H(W_A, W_S, U'_A)$
A retrieves W_S by subtracting W_A from $(W_A + W_S)$ and compares whether the hashed result of (W_S, U'_A) is equal to the received $H(W_S, U'_A)$. If it holds, A computes $H(W_A, W_S, U'_A)$ and sends it to S.

Step 4. $S \longrightarrow A$: **Access Granted/Denied**
S uses its own W_S and (W_A, U'_A) which is received from A in Step 1 to compute $H(W_A, W_S, U'_A)$ and verifies whether the hashed result of (W_A, W_S, U'_A) is equal to the received $H(W_A, W_S, U'_A)$. If so, S granted A's login request and replaced original $U_A = pw_A \cdot r_A \cdot G$ with new $U'_A = pw_A \cdot r'_A \cdot G$. Otherwise, S rejects A's login request. Finally, A's smart card replaces old r_A with new r'_A if all of the conditions are satisfied.

2.3 Password Change Phase

When the client A wants to update his/her current password pw_A to a new password pw'_A, A notifies S to replace current password-verifier $U_A = pw_A \cdot r_A \cdot G$ with new password-verifier $U'_A = pw'_A \cdot r'_A \cdot G$.

Step 1. $A \longrightarrow S$: $ID_A, E_{K_x}(ID_A, R_A, W_A, U'_A)$
Step 2. $S \longrightarrow A$: $W_A + W_S, H(W_S, U'_A)$
Step 3. $A \longrightarrow S$: $ID_A, H(W_A, W_S, U'_A), H(W_S + W_A + U'_A)$
Step 4. $S \longrightarrow A$: **Password Change Granted/Denied**

In Step 3, if the authentication token $H(W_A, W_S, U'_A)$ and $H(W_S + W_A + U'_A)$ are valid, A's smart card replaces r_A with r'_A and the password-verifier U_A has been changed with the new password-verifier U'_A.

2.4 Session Key Distribution Phase

Step 1. $A \longrightarrow S$: $ID_A, E_{K_x}(ID_A, R_A, W_A, U'_A)$
Step 2. $S \longrightarrow A$: $W_A + W_S, H(W_S, U'_A, SK)$
Step 3. $A \longrightarrow S$: $ID_A, H(W_A, W_S, U'_A, SK)$
Step 4. $S \longrightarrow A$: **Key distribution Granted/Denied**

In this phase, A and S compute the symmetric session key $SK = (r_A \cdot r_A \cdot pw_A) \cdot W_S = r_A \cdot r_A \cdot pw_A \cdot r_S \cdot d_S \cdot G = (r_S \cdot d_S) \cdot W_A = r_S \cdot d_S \cdot pw_A \cdot r_A \cdot r_A \cdot G$, where two random numbers r_A and r_S are chosen by A and S from $[1, n-1]$, respectively. After Step 1, 2, 3 and 4 are finished, S replaced U_A with U'_A and A's smart card replaces r_A with r'_A.

2.5 User Eviction Phase

In case of a client A is evicted by S, A cannot use (ID_A, U_A) to login S because S can delete (ID_A, U_A) from its verifier table and ID_A cannot be found in the verifier table in Step 2 of the password authentication phase.

3 Insider Attack on Li's Scheme

In this section, we show insider attack on Li's password authentication scheme. Let us consider the following scenarios. If a privileged-insider of S can find an opportunity to derive client A's real password pw_A, he/she may use A's password pw_A to impersonate A to login other servers.

After finishing the registration phase, the privileged-insider knows A's password-verifier $U_A = pw_A \cdot r_A \cdot G$. In addition, during the password authentication phase, client A sends a login request $\{ID_A, E_{K_x}(ID_A, R_A, W_A, U'_A)\}$ to S. Then the privileged-insider reveals (ID_A, R_A, W_A, U'_A) by using its secret key. Finally, the privileged-insider can derive client A's real password pw_A in off-line manner by using the following three steps:

Step 1. Select a guessed password pw^*_A.
Step 2. Compute $pw^*_A \cdot G$.
Step 3. Compare $\hat{e}(R_A, pw^*_A \cdot G)$ to $\hat{e}(U_S, U_A)$.

A match in Step 3 above indicates the correct guess of client A's password. The privileged-insider verifies the equation $\hat{e}(R_A, pw^*_A \cdot G)$ to $\hat{e}(U_S, U_A)$ holds or not as follows:

$$\begin{aligned}\hat{e}(R_A, pw^*_A \cdot G) &= \hat{e}(r_A \cdot U_S, pw^*_A \cdot G)\\ &= \hat{e}(r_A \cdot d_S \cdot G, pw^*_A \cdot G)\\ &= \hat{e}(d_S \cdot G, r_A \cdot pw^*_A \cdot G)\\ &= \hat{e}(U_S, U_A).\end{aligned}$$

As a result, the privileged-insider succeeds to guess the low-entropy password pw_A and Li's password authentication scheme is vulnerable to insider attack.

4 The Improved Scheme

In this section, we propose some slight modifications to Li's password authentication scheme, such as registration phase, password authentication phase and password change phase of Li's scheme. The other part of Li's password authentication scheme, such as session key distribution phase and user eviction phase are the same as Li's scheme. Figure 1 shows the entire flowchart of our improved scheme.

4.1 Registration Phase

Step 1. $A \longrightarrow S$**:** ID_A, U_A

When a client A wants to access the remote server S, A must register to S. A computes $v_A = H(ID_A||pw_A||r_A)$ and $U_A = v_A \cdot r_A \cdot G$ and sends it with ID_A to S through a secure channel, where r_A is a random number which is kept secret and only known by A.

Client A		Server S
Registration Phase:		
Select ID_A, pw_A, r_A		
$v_A = H(ID_A \Vert pw_A \Vert r_A)$		
$U_A = v_A \cdot r_A \cdot G$		
	$\xrightarrow{\{ID_A, U_A\}}$	
		Store $\{G, U_S, H(\cdot), E_K(\cdot)/D_K(\cdot)\}$
	$\xleftarrow{\textbf{SMART CARD}}$	
Store r_A into **SMART CARD**		
Password Authentication Phase:		
Enter ID_A, pw_A		
Retrieve r_A from **SMART CARD**		
Generate r'_A		
$v_A = H(ID_A \Vert pw_A \Vert r_A)$		
$v'_A = H(ID_A \Vert pw_A \Vert r'_A)$		
$R_A = r_A \cdot U_S$		
$W_A = r_A \cdot r_A \cdot v_A \cdot G$		
$U'_A = v'_A \cdot r'_A \cdot G$		
$E_{K_x}(ID_A, R_A, W_A, U'_A)$		
	$\xrightarrow{\{ID_A, E_{K_x}(ID_A, R_A, W_A, U'_A)\}}$	
		Decrypt $E_{K_x}(ID_A, R_A, W_A, U'_A)$ Verify $\hat{e}(R_A, U_A) \overset{?}{=} \hat{e}(W_A, U_S)$ $W_S = r_S \cdot U_S$
	$\xleftarrow{\{(W_A + W_S), H(W_S, U'_A)\}}$	
Retrieve W_S from $(W_A + W_S)$ Verify $H(W_S, U'_A)$		
	$\xrightarrow{\{ID_A, H(W_A, W_S, U'_A)\}}$	
		Verify $H(W_A, W_S, U'_A)$ Replace U_A with U'_A
	$\xleftarrow{\textbf{Access Granted/Denied}}$	
Replace r_A with r'_A		
Password Change Phase:		
Enter $ID_A, pw_A, pw_A^{new}, r_A\text{”}$		
$v_A\text{”} = H(ID_A \Vert pw_A^{new} \Vert r_A\text{”})$		
$U_A\text{”} = v_A\text{”} \cdot r_A\text{”} \cdot G$		
	$\xrightarrow{\{ID_A, E_{K_x}(ID_A, R_A, W_A, U_A\text{”})\}}$	
	$\xleftarrow{\{(W_A + W_S), H(W_S, U_A\text{”})\}}$	
	$\xrightarrow{\{ID_A, H(W_A, W_S, U_A\text{”}), H(W_S + W_A + U_A\text{”})\}}$	
		Replace U_A with U_A”
	$\xleftarrow{\textbf{Password Change Granted/Denied}}$	
Replace r_A with r_A”		

Fig. 1. The improved scheme

Step 2. $S \longrightarrow A$**: SMART CARD**
After receiving A's registration request, S stores A's identity ID_A, password-verifier U_A and a $status - bit$ in a write protected file as depicted in Table 2. Finally, S writes $\{G, U_S, H(\cdot), E_K(.)/D_K(.)\}$ into a personal smart card and issues it to A through a secure channel.

Step 3. After receiving the smart card from S, A writes r_A into the smart card and the smart card includes $\{r_A, G, U_S,\ H(\cdot), E_K(.)/D_K(.)\}$. Note that the client A does not need to remember r_A after finishing the phase.

Table 2. The verifier table of S after finishing the registration phase in our improved scheme

Identity	Password-verifier	$Status - bit$
$\vdots$	$\vdots$	$\vdots$
ID_A	$U_A = v_A \cdot r_A \cdot G$	0/1
$\vdots$	$\vdots$	$\vdots$

4.2 Password Authentication Phase

When a client A wants to access the server S, the client A inserts his/her personal smart card into card reader and keys the identity ID_A and the password pw_A. The following steps are performed during the password authentication phase.

Step 1. $A \longrightarrow S$**:** $ID_A, E_{K_x}(ID_A, R_A, W_A, U'_A)$
The smart card retrieves r_A, generates a new random number r'_A, computes $v_A = H(ID_A||pw_A||r_A)$, $v'_A = H(ID_A||pw_A||r'_A)$, $R_A = r_A \cdot U_S = r_A \cdot d_S \cdot G$, $W_A = r_A \cdot r_A \cdot v_A \cdot G$, $U'_A = v'_A \cdot r'_A \cdot G$ and $E_{K_x}(ID_A, R_A, W_A, U'_A)$ and sends $E_{K_x}(ID_A, R_A, W_A, U'_A)$ with ID_A to S, where the encryption key K_x is the x coordinate of $K = v_A \cdot r_A \cdot U_S = H(ID_A||pw_A||r_A) \cdot r_A \cdot d_S \cdot G = (K_x, K_y)$.

Step 2. $S \longrightarrow A$**:** $(W_A + W_S), H(W_S, U'_A)$
Upon receiving the login message, S computes the decryption key K_x by computing $K = d_S \cdot U_A = v_A \cdot r_A \cdot d_S \cdot G = (K_x, K_y)$ and decrypts $E_{K_x}(ID_A, R_A, W_A, U'_A)$ to reveal (ID_A, R_A, W_A, U'_A). S verifies the decrypted ID_A with received ID_A and $\hat{e}(R_A, U_A)$ with $\hat{e}(W_A, U_S)$. If they hold, S sends $\{(W_A + W_S), H(W_S, U'_A)\}$ to A, where r_S is a random number which is generated by S and $W_S = r_S \cdot U_S$. We can proof that the verification $\hat{e}(R_A, U_A) = \hat{e}(W_A, U_S)$ is correct and the remote server can confirm that A is a legal client. A bilinear pairing is used to assure the correctness of the scheme and is given below:

$$\begin{aligned}\hat{e}(R_A, U_A) &= \hat{e}(r_A \cdot d_S \cdot G, r_A \cdot v_A \cdot G)\\ &= \hat{e}(G, G)^{r_A \cdot r_A \cdot v_A \cdot d_S}\\ &= \hat{e}(r_A \cdot r_A \cdot v_A \cdot G, d_S \cdot G)\\ &= \hat{e}(W_A, U_S).\end{aligned}$$

Step 3. $A \longrightarrow S$**:** $ID_A, H(W_A, W_S, U'_A)$

A retrieves W_S by subtracting W_A from $(W_A + W_S)$ and compares whether the hashed result of (W_S, U'_A) is equal to the received $H(W_S, U'_A)$. If it holds, A computes $H(W_A, W_S, U'_A)$ and sends it to S.

Step 4. $S \longrightarrow A$**: Access Granted/Denied**

S uses its own W_S and (W_A, U'_A) which is received from A in Step 1 to compute $H(W_A, W_S, U'_A)$ and verifies whether the hashed result of (W_A, W_S, U'_A) is equal to the received $H(W_A, W_S, U'_A)$. If so, S granted A's login request and replaced original $U_A = v_A \cdot r_A \cdot G$ with new $U'_A = v'_A \cdot r'_A \cdot G$. Otherwise, S rejects A's login request. Finally, A's smart card replaces old r_A with new r'_A if all of the conditions are satisfied.

After finishing the password authentication phase, the verifier table of S is updated and the content of the verifier table is shown in Table 3.

Table 3. The verifier table of S after finishing the password authentication phase in our improved scheme

Identity	Password-verifier	$Status - bit$
⋮	⋮	⋮
ID_A	$U'_A = v'_A \cdot r'_A \cdot G$	0/1
⋮	⋮	⋮

4.3 Password Change Phase

When the client A wants to change his/her current password pw_A to a new password pw_A^{new}, A generates a new random number r_A'' and notifies S to replace current password-verifier $U_A = v_A \cdot r_A \cdot G$ with new password-verifier $U_A'' = v_A'' \cdot r_A'' \cdot G$, where $v_A'' = H(ID_A || pw_A^{new} || r_A'')$.

Step 1. $A \longrightarrow S$**:** $ID_A, E_{K_x}(ID_A, R_A, W_A, U_A'')$
Step 2. $S \longrightarrow A$**:** $W_A + W_S, H(W_S, U_A'')$
Step 3. $A \longrightarrow S$**:** $ID_A, H(W_A, W_S, U_A''), H(W_S + W_A + U_A'')$
Step 4. $S \longrightarrow A$**: Password Change Granted/Denied**

In Step 3, if the authentication token $H(W_A, W_S, U_A'')$ and $H(W_S + W_A + U_A'')$ are valid, A's smart card replaces r_A with r_A'' and the password-verifier U_A has been changed with the new password-verifier U_A''. After finishing the password change phase, the verifier table of S is updated and the content of the verifier table is shown in Table 4.

5 Security Analysis of the Proposed Scheme

The improved authentication scheme benefits from the protection of registration user to prevent the sensitive password for a privileged-insider to steal and guess

Table 4. The verifier table of S after finishing the password change phase in our improved scheme

Identity	Password-verifier	$Status - bit$
$\vdots$	$\vdots$	$\vdots$
ID_A	$U_A" = v_A" \cdot r_A" \cdot G$	0/1
$\vdots$	$\vdots$	$\vdots$

the real password stored in the verifier table or in the exchange of authentication messages. In the following propositions, we give an in-depth analysis of the proposed scheme in terms of security properties.

Since we add only two one-way hashing operations to replace $U_A = pw_A \cdot r_A \cdot G$ with $U_A = v_A \cdot r_A \cdot G$ during registration phase, where $v_A = H(ID_A||pw_A||r_A)$. Therefore, using the above mentioned attack in Section 3, a privileged-insider of S who does not know client A's random number r_A tries to derive client A's real password pw_A as follows:

$$\begin{aligned} v_A^* &= H(ID_A||pw_A^*||r_A^*) \\ U_A^* &= v_A^* \cdot r_A^* \cdot G, \end{aligned}$$

where pw_A^* is a guessed password and r_A^* is a guessed random number. Note that the random number r_A does not reveal to the privileged-insider of S and the bit length of $|r_A|$ is large enough. As a result, due to the intractability under the protection of $H(\cdot)$, a privileged-insider of S is unable to derive client A's pw_A without knowing r_A and client A's password pw_A will not be revealed by the privileged-insider.

On the other hand, if SHA-256 [25] is used in the proposed scheme, the privileged-insider of S may attempt to derive $v_A = H(ID_A||pw_A||r_A)$ and r_A due to the bit-length of v_A and r_A are 256 bits and 160 bits, respectively. Therefore, the probability to guess correct v_A and r_A at the same time are $\frac{1}{2^{256+160}}$. In addition, U_A must guess a correct password pw_A at the same time and the probability to guess a correct n characters pw_A approximated to $\frac{1}{2^{6n}}$. Therefore, it is computationally infeasible for the privileged-insider of S to guess correct $H(ID_A||pw_A||r_A)$, r_A and pw_A at the same time because the probability approximated to $\frac{1}{2^{6n+256+160}}$ and the privileged-insider of S will not be able to perform this attack.

6 Conclusions

In this paper, we have shown that a recently proposed ECC-based password authentication scheme in remote networking environment is insecure against insider attack and should not be implemented in real applications. To remedy the security problem, we have proposed security improvements which not only repair the weak features of Li's authentication scheme but also inherit the merits of Li's scheme.

Acknowledgements. This research was partially supported by the National Science Council, Taiwan, R.O.C., under contract no.: NSC 101-2221-E-165-002 and NSC 102-2221-E-030-003.

References

1. Chang, C.C., Lee, C.Y.: A smart card-based authentication scheme uing user identify cryptography. International Journal of Network Security 15(2), 139–147 (2013)
2. Das, A.K.: Improving identity-based random key establishment scheme for large-scale hierarchical wireless sensor networks. International Journal of Network Security 14(1), 1–21 (2012)
3. He, D., Zhao, W., Wu, S.: Security analysis of a dynamic ID-based authentication scheme for multi-server environment using smart cards. International Journal of Network Security 15(5), 350–356 (2013)
4. Islam, S.H., Biswas, G.P.: Design of improved password authentication and update scheme based on elliptic curve cryptography. Mathematical and Computer Modelling 57(11-12), 2703–2717 (2013)
5. Kar, J.: ID-based deniable authentication protocol based on Diffie-Hellman problem on elliptic curve. International Journal of Network Security 15(5), 357–364 (2013)
6. Kim, S.K., Chung, M.G.: More secure remote user authentication scheme. Computer Communications 32(6), 1018–1021 (2009)
7. Lamport, L.: Password authentication with insecure communication. Communications of the ACM 24(11), 770–772 (1981)
8. Lee, C.C., Chen, C.L., Wu, C.Y., Huang, S.Y.: An extended chaotic maps-based key agreement protocol with user anonymity. Nonlinear Dynamics 69(1-2), 79–87 (2012)
9. Lee, C.C., Hsu, C.W.: A secure biometric-based remote user authentication with key agreement protocol using extended chaotic maps. Nonlinear Dynamics 71(1-2), 201–211 (2013)
10. Lee, C.C., Li, C.T., Hsu, C.W.: A three-party password-based authenticated key exchange protocol with user anonymity using extended chaotic maps. Nonlinear Dynamics 73(1-2), 125–132 (2013)
11. Lee, C.C., Chen, C.T., Li, C.T., Wu, P.H.: A practical RFID authentication mechanism for digital television. Telecommunication Systems (article in press, 2013)
12. Li, C.T., Hwang, M.S.: An efficient biometrics-based remote user authentication scheme using smart cards. Journal of Network and Computer Applications 33(1), 1–5 (2010)
13. Li, C.T., Hwang, M.S.: An online biometrics-based secret sharing scheme for multiparty cryptosystem using smart cards. International Journal of Innovative Computing, Information and Control 6(5), 2181–2188 (2010)
14. Li, C.T.: Secure smart card based password authentication scheme with user anonymity. Information Technology and Control 40(2), 157–162 (2011)
15. Li, C.T., Lee, C.C.: A robust remote user authentication scheme using smart card. Information Technology and Control 40(3), 236–245 (2011)
16. Li, C.T., Lee, C.C.: A novel user authentication and privacy preserving scheme with smart cards for wireless communications. Mathematical and Computer Modelling 55(1-2), 35–44 (2012)

17. Li, C.T.: A new password authentication and user anonymity scheme Based on elliptic curve cryptography and smart card. IET Information Security 7(1), 3–10 (2013)
18. Li, C.T., Lee, C.C., Weng, C.Y., Fan, C.I.: An extended multi-server-based user authentication and key agreement scheme with user anonymity. KSII Transactions on Internet and Information Systems 7(1), 119–131 (2013)
19. Li, C.T., Weng, C.Y., Lee, C.C.: An advanced temporal credential-based security scheme with mutual authentication and key agreement for wireless sensor networks. Sensors 13(8), 9589–9603 (2013)
20. Li, C.T., Lee, C.C., Weng, C.Y.: An extended chaotic maps based user authentication and privacy preserving scheme against DoS attacks in pervasive and ubiquitous computing environments. Nonlinear Dynamics (article in press, 2013)
21. Liao, I.E., Lee, C.C., Hwang, M.S.: A password authentication scheme over insecure networks. Journal of Computer and System Sciences 72(4), 727–740 (2006)
22. Naveed, M., Habib, W., Masud, U., Ullah, U., Ahmad, G.: Reliable and low cost RFID based authentication system for large scale deployment. International Journal of Network Security 14(3), 173–179 (2012)
23. Kumar, M.: A new secure remote user authentication scheme with smart cards. International Journal of Network Security 11(2), 88–93 (2010)
24. Ramasamy, R., Muniyandi, A.P.: An efficient password authentication scheme for smart card. International Journal of Network Security 14(3), 180–186 (2012)
25. National Institute of Standards and Technology, US department of commerce, secure hash standard. US Federal Information Processing Standard Publication, 180–182 (2002)
26. Yang, L., Ma, J.F., Jiang, Q.: Mutual authentication scheme with smart cards and password under trusted computing. International Journal of Network Security 14(3), 156–163 (2012)