

Cryptanalysis and Improvement on Liu et al.'s User Authentication and Key Establishment Scheme

Bae-Ling Chen¹, Wen-Chung Kuo² and Lih-Chyau Wu¹,

¹ Graduate School of Engineering Science and Technology
National Yunlin University of Science and Technology
Douliu, Yunlin 640, Taiwan
{chenbl, wuulc}@yuntech.edu.tw

² Department of Computer Science and Information Engineering
National Formosa University
Huwei, Yunlin 632, Taiwan
simonkuo@nfu.edu.tw

Abstract. In 2007, Liu et al. proposed a password based remote user authentication and key establishment scheme using smart card for mobile environment. Though they said that their system is cost effectiveness, lower communication overhead, and protecting some attacks such as off-line and spoofing attacks, our research finds: (1) there are still flaws in their scheme such as lack of ownership check of the smart card and unconcern for the password change; (2) the scheme is vulnerable to the forgery attack. Those problems make their scheme fail in their design guidelines. We further propose an enhanced version of the scheme which does not only allow users to update their passwords securely, but also withstand forgery attacks.

Keywords: forgery attack, password security, mutual authentication, smart card, hash function.

1 Introduction

Remote mutual authentication is a mechanism to authenticate remote users and target server over insecure communication channel. It provides the legal users to use the resources of remote system. Among authentication scheme, password-based remote authentication schemes are most widely deploy. In 1981, Lamport [8] proposed the first one-time password remote authentication scheme. He used a password table to authenticate the validity of users over an open environment. Later, many researches [1-2, 4-5, 7, 9, 11-14] found that there are risk and cost in maintaining the password table which is shared between the remote system and the users in his scheme. Therefore, many password authentication schemes without password table were proposed [4-7, 9, 11-12]. Concurrently, several password authentication schemes using smart cards were proposed [1-2, 5-6, 9-10, 12-14]. For instance, Hwang and Li [5] proposed another remote user authentication scheme with smart cards and removed the risk and high cost in Lamport's scheme. Yang and Shieh

[14] proposed a user authentication scheme with smart cards based on RAS public key cryptosystem.

In 2007, Liu et al. proposed a password based remote user authentication and key establishment scheme [9] (LLZ-scheme for short) using smart card for mobile environment. Unfortunately, though they claimed their scheme has many merits such as cost effectiveness, lower communication overhead, and protecting some attacks such as off-line and spoofing attacks, we find that their scheme lacks to check the legality in ownership of the card; more than that, their scheme cannot withstand the forgery attacks; therefore, the scheme does not achieve their design guidelines. According to the defects of LLZ-scheme, we propose an improved version of the scheme to fix up their lacks.

The remainder of this paper is organized as follows. Section 2 lists the terms and notations used throughout the paper. We review LLZ-scheme and make an analysis of it in Section 3. In section 4, we propose a remote user authentication and key distribution scheme. We make security analysis of proposed scheme and comparison between LLZ-scheme and ours in section 5. Finally, we conclude our scheme in section 6.

2 The Notations

The terms and notations used throughout the paper are shown in Table 1.

Table 1. Terms and notations.

U_i	Identity of the i^{th} user
S_j	Identity of the j^{th} server
CM	Central manager
CA	Certificate authority
N, p, q	N is a modulus, p and q are two primitive numbers and $N = pq$, thus $\varphi(N) = \varphi(pq) = \varphi(p)\varphi(q) = (p-1)(q-1)$.
e, d	e and d are public and private keys, thus $ed = 1 \pmod{\varphi(N)}$.
$Cert_{CM}(N, e)$	Central manager's certificate issued from CA
g	g is a primitive element in \mathbf{Z}_N .
a	Public parameter and $a = g^d \pmod{N}$
H	One-way hash function
sk_j	Secret key of S_j
PW_i	Password of U_i
r_i	1 st random number for U_i generated by CM ; r_i is larger than 160 bits.
d_i	Secret parameter for U_i generated by CM
e_i	Public parameter for U_i generated by CM
p_i	Password-check parameter for U_i generated by CM
SC_i	U_i 's smart card
c_i	2 nd random number for U_i generated by SC_i ; c_i is larger than 160 bits.
t_i	Timestamp retrieved by SC_i .

v_1, v_2	Two temporal variables; $v_1 = g^{c_i e} \bmod N$, $v_2 = d_i^{PW_i} g^{c_i t_i} \bmod N$
n_i	Random nonce generated by SC_i
req_{ji}	Authentication request message to S_j submitted by SC_i $req_{ji} = \{U_i, t_i, e_i, v_1, v_2, E_{sk_j}(n_i), H(U_i t_i e_i v_1 v_2 n_i)\}$
$E_{sk_j}(M)$	Symmetrical ciphertext encrypted with key sk_j for plaintext M
t_j	Timestamp retrieved by S_j
Δt	Reasonable maximum transmission delay for SC_i
ck	Communication session key
$resp_{ji}$	Authentication response message for SC_i issued by S_j $resp_{ji} = \{E_{sk_j}((n_i + 1), ck)\}$
$D_{sk_j}(C)$	Plaintext decrypted with key sk_j for symmetrical ciphertext C

3 Review of LLZ-Scheme

We sketch LLZ-scheme in the first sub-session. The flaws of LLZ-scheme are shown in the second.

3.1 LLZ-Scheme

LLZ-scheme is designed under following guidelines:

- (1) the scheme should be suitable for the wireless environment;
- (2) the scheme should be secure enough to withstand the secret data reveal attacks;
- (3) the scheme should provide every user a convenience way to select or change his/her identity and password without the involvement of servers.

There are three roles in LLZ-scheme. They are: the remote user U , the server S , and the central manager CM . The scheme could be divided into five phases: initialization, registration, login, authentication, and password change.

Initialization Phase. This phase is the beginning of LLZ-scheme. In this phase, CM obtains its certificate, setups system parameters, and distributes secret information to every server S . It comprises the following steps:

- i. CM obtains its RSA certificate $Cert_{CM}(N, e)$ from a CA .
- ii. CM computes j th server S_j 's secret key $sk_j = H(a || S_j) \bmod N$.
- iii. CM distributes $\{Cert_{CM}(N, e), sk_j, H\}$ to each server S_j for $j = 1 \dots n$ via a secure channel.

Registration Phase. This phase is invoked when a new user registers him/herself to CM . It performs following steps:

- i. A new user chooses U_i and PW_i be his/her identity and password, and sends them to CM via a secure channel.
- ii. After received the messages, CM verifies the qualification of the user and

- drops the request if the user is not eligible. CM picks a random number r_i , computes $d_i = g^{r_i d} \bmod N$ and $e_i = g^{r_i PW_i} \bmod N$, where d_i and e_i are the secret and the public parameters of U_i .
- iii. CM stores $\{U_i, N, e, g, a, d_i, e_i, H\}$ on a smart card SC_i and issues SC_i to U_i securely.

Login Phase. After received a valid smart card, a user U_i wants to login a server S_j . It comprises the following steps:

- i. U_i inserts SC_i into his/her card reader and types-in the target server identity S_j and his/her password PW_i . After the input completed, SC_i picks a random number c_i larger than 160 bits and retrieves current time sequence t_i as a timestamp. SC_i then computes $v_1 = g^{c_i e} \bmod N$ and $v_2 = d_i^{PW_i} g^{c_i t_i} \bmod N$.
- ii. SC_i computes S_j 's secret key $sk_j = H(a \parallel S_j) \bmod N$ and an authentication request message $req_{ji} = \{U_i, t_i, e_i, v_1, v_2, E_{sk_j}(n_i), H(U_i \parallel t_i \parallel e_i \parallel v_1 \parallel v_2 \parallel n_i)\}$, where n_i is a random nonce generated by SC_i and E_{sk_j} is a symmetric encryption with key sk_j .

Authentication Phase. Upon receiving the request req_{ji} from U_i (infect, SC_i) to S_j , S_j and U_i performs following steps:

- i. S_j retrieves current time sequence t_j and checks that $t_j - t_i \leq \Delta t$ holds, otherwise S_j drops req_{ji} .
- ii. S_j recovers $n_i = D_{sk_j}(E_{sk_j}(n_i))$, verifies the identical of $H(U_i \parallel t_i \parallel e_i \parallel v_1 \parallel v_2 \parallel n_i)$, and S_j drops the request if anyone of those tests fail.
- iii. S_j computes and checks $(v_2^e)/(v_1^{t_i} e_i) \bmod N$, and reject the login of U_i if the result is not equal to 1.
- iv. S_j chooses a session key ck and generates the response message $resp_{ji} = \{E_{sk_j}((n_i + 1), ck)\}$, and sends $resp_{ji}$ to U_i .
- v. After received $resp_{ji}$, U_i retrieves $\{(n_i + 1), ck\}$ from $D_{sk_j}(E_{sk_j}((n_i + 1), ck))$. U_i confirms n_i and saves the session key ck for the subsequent communication.

Password Change Phase. As one of Liu et al.'s design guidelines, any user could select or change his/her password freely. Assume U_i wants to change his/her password PW_i to $PW_{i_{new}}$, it comprises the following steps:

- i. U_i makes sure that his/her smart card SC_i is inserted into the card reader properly and types-in the new password $PW_{i_{new}}$.
- ii. After the input completed, SC_i calculates the new public parameter $e_{i_{new}} = d_i^{e^{PW_{i_{new}}}} \bmod N = g^{r_i PW_{i_{new}}} \bmod N$, and overwrites e_i saved on SC_i with $e_{i_{new}}$.

3.2 The Ownership Checking and Forgery Attack on LLZ-Scheme

Although Liu et al. claimed that their scheme can achieve their three requires, our research finds that their scheme lacks to check the ownership legality of the smart card; beside, their scheme cannot withstand the forgery attack; therefore it does not achieve their design guidelines. We discuss in the following.

The Ownership Checking. For security, every time after a smart card SC_i is inserted into a card reader and before it works, SC_i should check the owner's legality by verifying the password. As the same point of view, SC_i should do the check before the password change to prevent mindless action and get better security. Without above password checks, there is no fundamental security. Note that, in our design, every time after the card is inserted into a card reader or before a user wants to change the password on the card, SC_i asks the user to input password. If the user completes the verification, SC_i allows subsequent process; otherwise SC_i keeps asking the password or halt after certain errors.

Since this ownership check only runs when the user inserts the smart card into the card reader or when the user wants to change the password, the computation overhead of this verification is negligible.

The Forgery Attack. In this section, we show an attacker U_k , a malicious user, collects the communication traffic between a user U_i (infect, the smart card SC_i) and a server S_j in LLZ-scheme. Then the attacker could retrieve the authentication request message req_{ji} from the traffic, manipulate the req_{ji} , masquerade as the user U_i , pass the reply test, and then retrieve the session key ck .

Assume the attacker U_k retrieves $req_{ji} = \{U_i, t_i, e_i, v_1, v_2, E_{sk_j}(n_i), H(U_i || t_i || e_i || v_1 || v_2 || n_i)\}$ from the communication traffic between the user U_i and the server S_j .

- i. U_k guesses a forge password PW_k and computes following values by his/her smart card SC_k ;

$$\begin{aligned} e_k &= e_i^{PW_k} = d_i^{ePW_iPW_k} = g^{r_iPW_iPW_k} \pmod N, \\ v_{1k} &= v_1^{PW_k} = g^{c_iPW_k} \pmod N, \\ v_{2k} &= v_2^{PW_k} = d_i^{PW_iPW_k} g^{c_{t_i}PW_k} \pmod N, \text{ and} \\ sk_j &= H(a || S_j) \pmod N. \end{aligned}$$

Then SC_k computes a forge authentication request message req_{jik} as following:

$$req_{jik} = \{U_i, t_k, e_k, v_{1k}, v_{2k}, E_{sk_j}(n_k), H(U_i || t_k || e_k || v_{1k} || v_{2k} || n_k)\};$$

U_k transmits the forge request req_{jik} to S_j .

- ii. Upon receiving the request req_{jik} from U_k , S_j performs the authentication phase as usual. We note that $(v_2^e)/(v_1^t e_i) = 1 \pmod N$ will be the critical testimony proving the eligibility of the attacker U_k by following:

$$(v_{2k}^e)/(v_{1k}^t e_k) = (g^{r_i dPW_iPW_k} g^{c_{t_i}PW_k})^e / (g^{c_iPW_k} g^{r_iPW_iPW_k}) = 1 \pmod N.$$

- iii. Since S_j treats U_k as U_i , as a result of the authentication phase, the attacker U_k retrieves ck from $resp_{jik}$ successfully.

4 The Proposed Scheme

In the following, we propose a modification scheme. This modification is more secure and efficient than LLZ-scheme. In our scheme, there are also five phases: initialization, registration, login, authentication, and password change. The most different parts in our scheme are completed by the smart card. Fig. 1 shows our scheme, and we describe the detail as following.

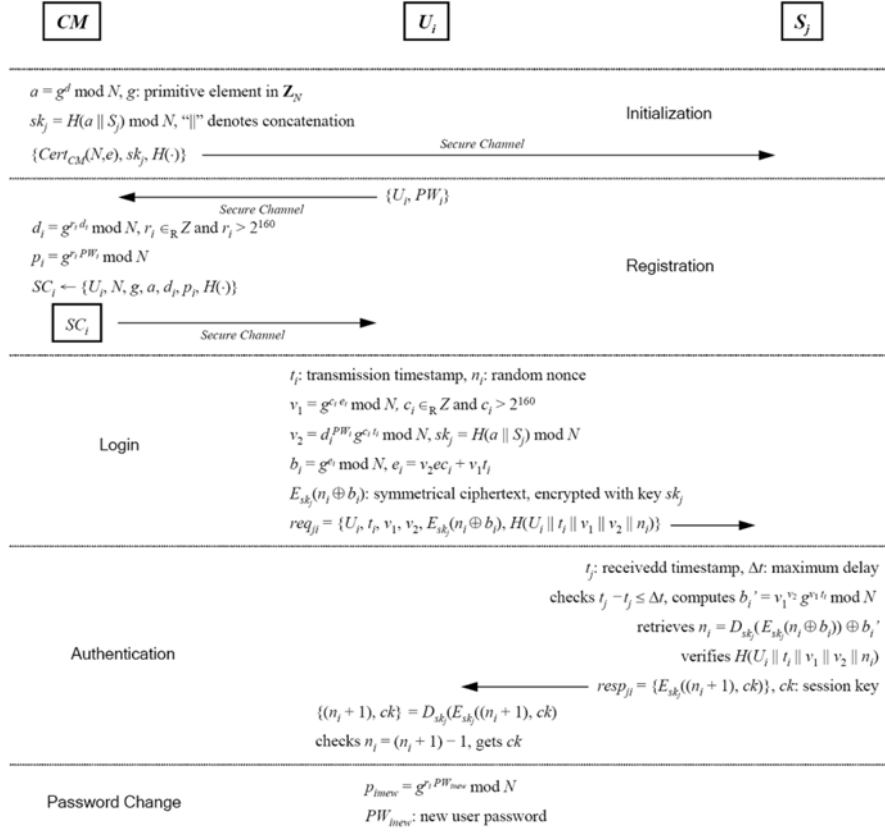


Fig. 1. Our improved user authentication and key establishment scheme.

4.1 Initialization Phase

This phase is the same as LLZ-scheme.

4.2 Registration Phase

This phase is invoked when a new user registers him/herself to CM . It performs following steps:

- Step 1. A new user chooses U_i and PW_i be his/her identity and password, and sends them to CM via a secure channel.
- Step 2. After received the messages, CM verifies the qualification of the user and drops the request if the user is not eligible. CM picks a random number r_i , computes $d_i = g^{r_i} \bmod N$ and $p_i = g^{r_i PW_i} \bmod N$. CM stores $\{U_i, N, e, g, a, d_i, p_i, H\}$ on a smart card SC_i and issues SC_i to U_i securely.

4.3 Login Phase

After received a valid smart card, a user U_i wants to log-in a server S_j . It comprises the following steps:

- Step 1. U_i inserts SC_i into his/her card reader and types-in the target server identity S_j and his/her password PW_i' . After the input completed, SC_i retrieves p_i and checks PW_i' by verifying $p_i \stackrel{?}{=} d_i^{ePW_i'} \pmod N$. If the equation holds, SC_i allows subsequent process; otherwise SC_i keeps asking the password or halt after certain errors.
- Step 2. If U_i passes through the ownership checking, SC_i picks a random number c_i and retrieves current time sequence t_i as a timestamp. SC_i then computes $v_1 = g^{c_i e} \pmod N$, $v_2 = d_i^{PW_i'} g^{c_i t_i} \pmod N$, and $b_i = g^{e_i} \pmod N$, where b_i is a blind factor and $e_i = v_2 e c_i + v_1 t_i$. Note that there are many time servers on Internet, such as, the Network Time Protocol (NTP) server. The user and server could randomly access the time server to synchronize the system clocks.
- Step 3. SC_i computes S_j 's secret key $sk_j = H(a || S_j) \pmod N$ and an authentication request message $req_{ji} = \{U_i, t_i, v_1, v_2, E_{sk_j}(n_i \oplus b_i), H(U_i || t_i || v_1 || v_2 || n_i)\}$, where n_i is a random nonce generated by SC_i and E_{sk_j} is a symmetric encryption with key sk_j .

4.4 Authentication Phase

Upon receiving the request req_{ji} from U_i (SC_i , in fact), S_j and U_i perform following steps:

- Step 1. S_j retrieves current time sequence t_j and checks that $t_j - t_i \leq \Delta t$ holds, otherwise S_j drops req_{ji} .
- Step 2. S_j calculates $b_i' = v_1^{v_2} g^{v_1 t_i} \pmod N$, recovers $n_i = D_{sk_j}(E_{sk_j}(n_i \oplus b_i)) \oplus b_i'$, and verifies the identical of $H(U_i || t_i || v_1 || v_2 || n_i)$. S_j drops the request if anyone of these tests fails.
- Step 3. S_j chooses a session key ck , generates the response message $resp_{ji} = \{E_{sk_j}((n_i + 1), ck)\}$, and sends $resp_{ji}$ to U_i .
- Step 4. After received $resp_{ji}$, U_i retrieves $\{(n_i + 1), ck\}$ from $D_{sk_j}(E_{sk_j}((n_i + 1), ck))$. If U_i confirms n_i , U_i believe the session key ck could be trusted. U_i saves ck for the subsequent communication.

4.5 Password Change Phase

As we mention before, there should be an ownership checking before the password change. Except the ownership checking, this phase is the same as LLZ-scheme.

5 Analysis and Comparison

To prove the result of our proposed scheme, we analyze the security of our paper and compare the overhead of LLZ-scheme with ours.

5.1 Security Analysis

We have been defined the attack models in section 3.2. In this section, we will describe the security analysis of our scheme against them.

Provide The Fundamental Password Security. Our scheme checks the ownership between the user and his/her smart card when a smart card is inserted into a card reader or the password saving on card changed every time. That is the fundamental password security.

Withstand The Forgery Attack. The authentication request message of our scheme is $req_{ji} = \{U_i, t_i, v_1, v_2, E_{sk_j}(n_i \oplus b_i), H(U_i || t_i || v_1 || v_2 || n_i)\}$. Since there is a timestamp t_i included within the request message, the attacker cannot forgery this message in time. Furthermore, the attacker must need to recover $b_i' = v_1^{v_2} g^{v_1 t_i} \text{ mod } N$ before he/she obtains $n_i = D_{sk_j}(E_{sk_j}(n_i \oplus b_i)) \oplus b_i'$. Therefore there is no the forgery problem in our system.

Reduce The Traffic of Communications. After we remove the e_i term from authentication request message in our scheme, it reduces the communication overhead, and it comes with another benefit: a foe could not collect any information of it.

Table 2. Function comparisons between LLZ-scheme and ours.

	LLZ-Scheme	Ours
Provide fundamental password security	No	Yes
Provide secure password change	No	Yes
Withstand the man-in-the-middle attack	No	Yes
Withstand the forgery attack	No	Yes

5.2 Performance Comparison

We compare the difference between LLZ-scheme and our scheme in Table 3. Although it does not save great numbers of costs more than LLZ-scheme, our scheme provides fundamental password security, withstands the forgery attack, and reduces the communication overhead.

We note that our system does not take $E_{sk_j}(n_i)$ but $E_{sk_j}(n_i \oplus b_i)$ instead in the authentication request message. It is a light overhead method to protect n_i to ensure

that only the target server S_j could retrieve the token, n_i . We also recall that we remove e_i from the message; the $e_i = v_2ec_i + v_1t_i$ is calculated dynamically, going along with the timestamp t_i . It makes our scheme efficient and secures more than LLZ-scheme.

Table 3. Computation overheads between LLZ-scheme and ours.

	LLZ-Scheme	Ours
Initialization	$E \times n, H \times n$	$E \times n, H \times n$
Registration	$E \times 3$	$E \times 3$
Login	$E \times 4, M \times 1,$ $H \times 2, Enc \times 1$	$E \times 4, M \times 4,$ $H \times 2, Enc \times 1$
Authentication	$E \times 2, M \times 2, H \times 1$ $Enc \times 1, Dec \times 2$	$E \times 3, M \times 1, H \times 1$ $Enc \times 1, Dec \times 2$
Password Change	$E \times 2$	$E \times 2$

* Where: E : exponential, M : multiplication, H : hashing, Enc : encryption, Dec : decryption, n : numbers of servers

5.3 The Designed Guidelines of LLZ-Scheme

Our proposed scheme could meet three designed guidelines of LLZ-scheme.

The scheme should be suitable for the wireless environment. As the above discussions, our scheme is more secure than LLZ-Scheme and fit for the wireless environment.

The scheme should be secure enough to withstand the secret data reveal attacks. Our scheme fixes up the lacks of LLZ-Scheme to withstand the secret data reveal attacks.

The scheme should provide every user a convenience way to select or change his/her identity and password without the involvement of servers. We have showed that our scheme does not only hold the fundamental password security but also provide a convenience method to users to maintain their password freely.

6 Conclusion

We show that LLZ-scheme lacks for fundamental password security, loses on user authentication, and fails in their design guidelines. Furthermore, we proposed an improved scheme which does not only enable users to update their passwords securely, but also withstand the forgery attack.

References

1. H.-Y. Chien, J.-K. Jan and Y.-M. Tseng: A modified remote log in authentication scheme based on geometric approach. *Journal of System and Software*, vol. 55, pp. 287–290 (2001)
2. C.-C. Chang and T.-C. Wu: Remote password authentication with smart cards. *IEEE Proceedings*, vol. 138, pp. 165–168 (1991)
3. National Institute of Standards and Technology (NIST): Digital signature standard. Federal Information Processing Standards Publication 186 (1994)
4. G. Horng: Password authentication without using password table. *Inform Process*, pp. 247–250 (1995)
5. M. S. Hwang, and L. H. Li: A new remote user authentication scheme using smart cards. *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp 28-30 (2000)
6. M.-S. Hwang, C.-C. Lee, Y.-L. Tang: A simple remote user authentication scheme. *Mathematical and Computer Modelling*, vol. 36, no. 1, pp. 103-107 (2002)
7. M. S. Hwang: A remote password authentication scheme based on the digital signature method. *International Journal of Computer Mathematics* 70, pp. 657–666 (1998)
8. L. Lamport: Password authentication with insecure communication. *Communication of ACM*, vol. 24, pp. 770-772 (1981)
9. J. Liu, J. Liao, X. Zhu: A password-based authentication and key establishment scheme for mobile environment. *21st International Conference on Advanced Information Networking and Applications Workshops, AINAW'07*, vol. 2, pp. 99-104 (2007)
10. H.-M. Sun: An efficient remote user authentication scheme using smart cards. *IEEE Transactions on Consumer Electronics*, vol. 46, no. 4, pp. 958-961 (2000)
11. T.-C. Wu: Remote login authentication scheme based on a geometric approach. *Computer Communications*, vol. 18, no. 12, pp. 959–963 (1995)
12. C.-T. Wang, C.-C. Chang, and C.-H. Lin: Using IC cards to remotely login passwords without verification tables. *18th International Conference on Advanced Information Networking and Applications, AINA 2004*, vol. 1, Fukuoka, Japan (2004)
13. E. J. Yoon, E. K. Ryu, and K. Y. Yoo: Efficient remote user authentication scheme based on generalized ElGamal signature scheme. *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 568-570 (2004)
14. W. H. Yang, S. P. Shieh: Password authentication schemes with smart cards. *Computers & Security*, vol. 18, no. 8, pp. 727–733 (1999)