Fast Remote User Authentication Scheme with Smart Card Based on Quadratic Residue

Tzong-Sun Wu[1], Han-Yu Lin[2], Ming-Lun Lee[1], Won-Yi Chen[3]
[1]Department of Computer Science and Engineering
National Taiwan Ocean University
Keelung, 202
Taiwan

[2]Department of Information Management
Chang Gung University
Tao-Yuan, 333
Taiwan

[3]Department of Informatics
Fo Guang University
I Lan, 262, Taiwan

Journal of Digital Information Management

ABSTRACT: *There are many kinds of online services in our daily life, such as e-mail and messenger. When we link to a remote host, we are required to login first, supplying a username and the corresponding password, and then obtain the resources or have services. Most commonly, we might use a smart card to assistant us to login a remote site. In 1981, Lamport proposed a remote authentication scheme for communication through insecure channels. Lu and Cao further proposed an efficient scheme based on quadratic residue in 2005. In the Lu-Cao scheme, it is unnecessary for the server to store the verification table. However, some obvious drawbacks are that users cannot freely choose their own passwords and their scheme is inefficient due to complex computations. In this paper, we proposed an improvement of the Lu-Cao scheme to make it more efficient and secure.*

Categories and Subject Descriptors
D.4.6 [Security and Protection]; Authentication: C.3 [Special-purpose and application based systems]; Smartcards

General Terms: Smart Cards, Remote Access, Quadratic residue, User authentication

Keywords: Password authentication, Quadratic residue, Verification table, Smart card

Received: 18 October 2010; Revised 29 November 2010; Accepted 5 December 2010

1. Introduction

In 1981, Lamport [3] proposed a remote authentication scheme for communication through insecure channels. It has to store a verification table for verifying the legitimacy of login users. Yet it might still cause problems if any intruder can modify the passwords stored in the verification table. In 2000, therefore, Hwang and Li [1] proposed a novel remote authentication scheme using smart cards [2, 11] based on ElGamal public key cryptosystem. The advantage is that the extra storage of smart cards helps the remote server to store data without the assistance of the verification table. Besides, a microprocessor embedded in the smart card can balance some computation load of the remote server, which contributes to the time saving. Also, a user just needs to compute the password with his own secret key to avoid accidentally revealing the password. However, to equip smart cards with cryptographic functions, the computation complexity and memory space are concerned the most.

Recently, Lu and Cao [6] proposed an improved scheme from the Hwang-Li scheme with quadratic residue [7]. It has better performance than the Hwang-Li scheme owing to the faster execution and more bandwidth savings. Notwithstanding the advantages above, it still has some defects. Although the remote server does not need to store the verification table, users cannot freely choose the password they want. Therefore, the server must check the password every time when a user logins. The computation is very complicated and leads to the poor efficiency in verification. In this paper, we amended the Lu-Cao scheme to propose an improved one which allows a user to freely select his own password, and is more efficient and secure.

2. The proposed scheme

Preserving the merit inherited in the Lu-Cao scheme that the server does not need to maintain a verification table, our proposed scheme also overcomes the inflexibility of choosing passwords at will. The proposed scheme is divided into four phases: initial, registration, login and authentication phases. Details of each phase are described as follows:

2.1 Initial phase

The remote server publishes the following parameters in this phase:

p, q: two large primes satisfying $p \equiv q \equiv 3 \bmod 4$;

n: a composite number, i.e., $n = pq$;

H: a one-way hash function mapping from $\{0, 1\}^*$ to Z_n^*.

Correspondence to:
Han-Yu Lin, Ph.D, Department of Information Management, Chang Gung University, Taiwan, Republic of China, hanyu.cs94g@nctu.edu.tw

2.2 Registration Phase

A new user U_i submits his PW_i to the system for registration. After receiving the request, the remote system performs the following procedures:

1. Compute S_i satisfying that $(H(PW_i))^2 = [H(ID_i) \oplus S_i] \bmod n$, where ID_i is generated by the remote system.
2. Send ID_i and a smart card containing (ID_i, S_i, H) to U_i.

2.3 Login Phase

When the user U_i wants to login the system, he inserts his smart card into a card reader and types his (ID_i, PW_i). The smart card then executes the following procedures:

1. Compute $k = H(H(PW_i), S_i, ID_i, T)$ where T is the login time-stamp with a fixed format.
2. Compute $w(S_i \oplus k)^2 \bmod n$ where "$\oplus$" denotes the XOR operation.
3. Send (ID_i, k, w, T) to the remote system.

2.4 Authentication Phase

On receiving the login request, the remote server records the receiving time T', and performs the following steps:

1. Validate the time difference of T and T'. If $\Delta T \le (T' - T)$, rejects the request; otherwise, proceed to the next step. Here, ΔT is the acceptable time interval of transmission delay.
2. Compute $S_i = (w^{1/2} \oplus k) \bmod n$.
3. Compute $H(PW_i) = (S_i \oplus H(ID_i))^{1/2} \bmod n$.
4. Check if $k = H(H(PW_i), S_i, ID_i, T)$ holds. If it does, the login request is granted; otherwise, terminate the procedure.

It can be seen that the proposed scheme overcomes the disadvantage of the Lu-Cao scheme that users can not choose their passwords at will. Some security considerations and the performance evaluation will be demonstrated in next section.

3. Security Analyses and Performance Evaluation

In this section, we analyze the security of our proposed scheme and evaluate its efficiency. For facilitating the reader with quantitative results of the proposed scheme, we define the following notations:

a: the size of password space;

b: the output size of hash function H;

c: the number of all legitimate users. (from published ID list)

3.1 Security Analyses

We discuss the following situations and some well-known attacks against the proposed scheme.

3.1.1 Losing the Smart Card

Each S_i stored in a smart card is different. Even if the smart card is lost, any legitimate user can not use the smart card along with his own ID and password to login the remote system for that the S_i will not pass the verification of the server. For an illegal user U_A attempting to login with a randomly picked card, the probability of being successfully authenticated is $1/a \times 1/c$. Even though he can track the ID within the card, the success probability is only $1/a$.

3.1.2 Impersonation Attack/ Forgery Attack [10]

When a legitimate user finishes the registration phase, a secret S_i will be stored in the user's smart card. Upon receiving a login request, the server will use the submitted information to compute S_j which is then compared with S_i stored in the user's smart card. If $S_i \neq S_j$, the remote server rejects the login request. Hence, it is impossible for an adversary to impersonate a legitimate user without obtaining the secret S_i. The success probability of this attack is also $1/a$.

3.1.3 Eavesdropping Attack

The security of our proposed scheme is based on the factorization problem. Without the knowledge of two large primes p and q, any intruder can not obtain useful information from intercepted data. Without p and q, the adversary cannot derive S_i from $w = (S_i \oplus k)^2 \bmod n$ with his intercepted k and w. Therefore, the situation is identical to the above ones.

3.1.4 Denial of Service Attack, DoS Attack [5]

To launch a denial of service attack, an adversary has to login the remote system first. However, according to the analysis of eavesdropping attacks, any intruder cannot obtain useful information to help him login the remote system, let alone launch denial of service attacks, unless he can make enough registrations or obtain sufficient (ID, PW)'s within cards. The success probability of the latter is $1/a$. And the former will be more costly.

3.1.5 Replay Attack [5]

We apply the concept of timestamps to our proposed scheme, so that the smart card will record every login time when a user enters his PW and ID. Similarly, the remote server also generates a timestamp upon receiving the login request sent from the smart card. If the time interval between the two timestamps is within reasonable period, the login request is granted. Any replay attack that can not update the intercepted data with a fresh timestamp will be detected by the remote server. In other words, the adversary cannot use a selected timestamp T' to forge k without obtaining PW and S_i. The probability of successfully launching this attack is only $1/a$.

3.1.6 Password Guessing Attack [9]

Each time a legitimate user enters his PW, the smart card XORs the current timestamp with inputted data to get a various remote login request. Either online or offline attacks, any attacker cannot guess a valid password by gathering transmitted data. Unless the adversary possesses S_i, he cannot launch this attack. Besides, S_i can only be derived from either w and k with the knowledge of p and q, or hashed value k. Hence, the probability of successful guessing is $1/a + 1/b$.

3.1.7 Man-in-the-Middle Attack

It is infeasible to successfully launch a man-in-the-middle attack, since the proposed scheme employs the concept of timestamps. Further, by XORing the timestamp, the login request is different each time, which prevents an attacker from getting useful information. Therefore, the situation is the same as that in attacks 3.1.3 and 3.1.4.

3.1.8 The Stolen-Verifier Attack [5, 10]

In the proposed scheme, the remote server does not need the assistance of a verification table to authenticate a user login request. That is, all the required data is computed from the information offered by the user. Hence, it is impossible for an attacker to mount the stolen-verifier attack on it. All he can do is to try any possible PW to login with the success probability of $1/a$.

3.2 Capability Analyses

We summarize the functionality and repellence of our proposed scheme and other schemes in Table 1.

	Wu *et al.*'s scheme [12]	Lee-Chiu scheme [4]	Hwang-Li scheme [1]	Lu-Cao scheme [6]	Proposed scheme
Without verification table	×	O	O	O	O
Choose password at will	O	O	×	×	O
Change password at will	O	O	×	×	O
Resist DoS attack (Denial of Service)	×	O	O	O	O
Resist replay attack	×	O	O	O	O
Resist guessing attack	O	O	O	O	O
Resist man-in-the-middle attack	×	O	O	O	O

Table 1. Capability comparisons of the proposed and other schemes

As shown in Table 1, only Wu *et al.*'s scheme needs to store a verification table. In their scheme, the remote server has to record the login times which might result in login failure if the recorded times is accidentally increased by illegal attempted logins such as replay attacks and so on. Therefore, Wu *et al.*'s scheme can not resist DoS attack, replay attack and men-in-the-middle attack. A common weakness of the Hwang-Li and the Lu-Cao schemes is that they can not enable users to choose and change their passwords at will.

3.3 Performance Evaluation

The key point in improving the efficiency of authentication processes is to reduce computation loads of smart cards. Thus, we have made the computation load of smart cards as light as possible in our proposed scheme. In table 2, we compare the proposed scheme with some previous works in terms of the computation complexity of each phase and each phase is further divided into two parts: User U_i and Server S. Note that the computation cost of smart cards in Login Phase is regarded as that of U_i. Some used notations are defined below:

- T_m: the time for performing a modular multiplication computation.
- T_i: the time for performing a modular inverse computation.
- T_e: the time for performing a modular exponentiation computation.
- T_h: the time for performing a one-way hash function.
- T_r: the time for performing a modular square root computation.

It can be seen from Table 2 that the computational efficiency of our proposed scheme in either Login or Authentication phases is not the best among all works. Nevertheless, consider total computational efficiency of the two phases, one can find that our proposed scheme outperforms others. Note that the Registration Phase is only performed for the first time. Among mentioned several computation operations, the modular exponentiation operation is of main concern for whole computation loads since it is far heavier than those of others. Further, square root computation needs to perform exponentiation operation. Consider computation efforts of the user side, the proposed scheme is better than others except the Lu-Cao scheme. However, their scheme can not provide complete functionality. Thus, we conclude that the proposed scheme has a better performance in computation efficiency.

	Registration Phase		Login Phase		Authentication Phase	
	U_i	S	U_i	S	U_i	S
Wu *et al.*'s scheme [12]	T_m	$2T_m + 3T_h + 2T_r$	T_r	0	0	$3T_m + 2T_r + 3T_h$
Lee-Chiu scheme [4]	0	$T_m + T_e + 2T_h$	$2T_m + T_e + 2T_h$	0	0	$T_m + T_i + 2T_h$
Hwang-Li scheme [1]	0	T_e	$T_m + 3T_e + T_h$	0	0	$T_m + T_i + 2T_e + T_h$
Lu-Cao scheme [6]	0	$3T_m + 3T_e + 2T_h + T_r$	$T_m + T_h$	0	0	$4T_m + T_r + T_i + 3T_e + 3T_h$
Proposed scheme	0	$T_m + 2T_h$	$T_m + 2T_h$	0	0	$3T_h + 2T_r$

Table 2. Efficiency comparisons of the proposed and other schemes

4. Conclusions

Since Lamport proposed a remote authentication scheme for communication through insecure channels in 1981, many researchers have addressed improved schemes to obtain better security and efficiency. In 2005, Lu and Cao proposed an enhanced version based on quadratic residue. Although the Lu-Cao scheme has the merit that the remote server does not need to store a verification table, users can not freely choose their passwords and their scheme is inefficiency due to complicated computations. In this paper, we amended the weaknesses of the Lu-Cao scheme and proposed an improvement which still preserves the advantage of their scheme. The proposed scheme is more efficient and user-friendly than the Lu-Cao scheme, because computation loads of both the smart card and the whole system are rather low. Users can freely choose their own passwords at will and the server does not need to store the verification table.

References

[1] Hwang, M.S., Li, L.H (2000). A new remote user authentication scheme using smart cards, *IEEE Transactions on Consumer Electronics* 46 (1) 28-30.

[2] Konstantinos, M., Keith, M. (2003). An overview of the GlobalPlatform smart card specification, Information Security Technical Report 8 (1) 17-29.

[3] Lamport, L. (1981). Password authentication with insecure communications, *Communication of ACM* 24 (11) 770-772

[4] Lee, N.Y., Chiu, Y.C (2005). Improved remote authentication scheme with smart card, *Computer Standards & Interfaces* 27 (2) 177-180.

[5] Lin, C.L., Sun, H.M., Hwang, T. (2001). Attacks and solution on strong-password authentication, *IEICE Transactions on Communications* E84-B (9) 2622-2627.

[6] Lu, R., Cao, Z. (2005). Efficient remote user authentication scheme using smart card, *Computer Networks* 49 (4) 535-540

[7] Manders, K.L., Adleman, L.M. (1978). NP-complete decision problems for binary quadratics, *Journal of Computer and System Sciences* 16 (2) 168-184.

[8] Purdy, G.B. (1974). High security log-in procedure, Communications of the ACM 17 (8) 442-445.

[9] Shai, H., Hugo, K (1999). Public key cryptography and password protocols, *ACM Transaction on Information and System Security* 2 (3) 230-268.

[10] Tsuji, T., Shimizu, A (2004). One-time password authentication protocol against theft attacks, *IEICE Transactions on Communications* E87-B (3) 523-529.

[11] Wolfgang, R (2003). Overview about attacks on smart cards, Information Security Technical Report 8 (1) 67-84.

[12] Wu, Y.C., Lin, H.F., Chen, C.Y. (2006). A fair and dynamic password authentication system, *In*: The 16th Information Security Conference 2006, Taichung, Taiwan, June 8-9, p. 222-228.