A Secure Password-Authenticated Key Agreement Using Smart Cards

Kai Chain[1], Wen-Chung Kuo[2] and Jiin-Chiou Cheng[3]

[1]*Department of Computer and Information Science, R.O.C. Military Academy, Kaohsiung 83059, Taiwan, R.O.C.*
[2]*Department of Computer Science and Information Engineering, National Yunlin University of Science and Technology, Yunlin 64002, Taiwan, R.O.C.*
[3]*Department of Computer Science and Information Engineering, Southern Taiwan University of Science and Technology, Tainan 71005, Taiwan, R.O.C.*
chainkai@mail2000.com.tw, simonkuo@yuntech.edu.tw, chiou@mail.stut.edu.tw

Abstract

Smart card based password for authentication has become a common trend. Although smart card brings conveniences, it also increases the risk in the case of lost cards. In other words, when the smart card is possessed by an attacker, the attacker will possibly attempt to analyze the secret information within the smart card to deduce the authentication mechanism of the server and then forge user credentials or break the entire authentication system. In this paper, we analyze the lost smart card attack from Juang, et al.'s scheme [9] that proposes password authenticated key agreement and propose an improved robust and efficient user authentication and key agreement scheme using smart cards. In order to bolster the security of the entire system, we mitigated some of its weaknesses.

Keywords: *Key exchange, Elliptic curve cryptosystem, Smart card, Authentication*

1. Introduction

When a user wants to obtain server-related services, the user will use password authentication to verify identity to the server. Until now, many different authentication schemes have been proposed. In 2005, Fan et al. proposed a robust remote authentication scheme with smart cards [3]. They claimed that their proposed scheme can satisfy the following eight criteria:

1. Lower computational workload for smart cards.
2. Does not require the user passwords table.
3. Users can choose their own passwords freely.
4. Clock synchronization is not required and no delay-time limitations.
5. Thwarts replay attack.
6. Provides server authentication.
7. Offline dictionary attacks are ineffective.
8. Lost cards can be revoked without changing user identities.

The major contribution of Fan, *et al.*'s scheme (for short FCZ-scheme) [3] is providing a method for resisting offline dictionary attack so that the scheme is secure

even if the attackers acquire the information stored on the smart card. In 2008, Juang, *et al.*, (for short JCL-scheme) [9] point out the major drawbacks are loss of anonymity for the user and high computation and communication cost in Fan, *et al.*'s scheme. Furthermore, JCL-scheme does not provide a function for session key agreement and cannot prevent insider attack [5]. To improve upon these drawbacks, Juang, *et al.*, proposed a scheme that not only can provide identity protection but also keep lower communication and computation cost by using elliptic curve cryptosystems. They also proposed a solution for minimizing the risk of lost cards. In other words, in order to avoid information leakage when a card is lost, the card can be revoked. This approach seems viable on the surface, but actually has a design flaw. The use of a fixed server key allows an offline attack to be mounted against the server key when an attacker possesses the user card. Therefore, we propose to improve JCL-scheme and mitigate the exposure of the entire system when a smart card is compromised.

The paper is organized as follows: In Section 2, we review JCL-scheme [9] and analyze its weaknesses. In Section 3, we propose our scheme. In Section 4, the security analysis of our proposed scheme and comparison with JCL-scheme are discussed. Finally, in Section 5, we conclude the paper.

2. Review and Analysis of the JCL-scheme

In 2008, Juang, *et al.*, proposed a robust and efficient user authentication and key agreement scheme which not only satisfies all the benefits of Fan-Zhang scheme but can also provide identity protection and session key agreement. It can withstand insider attack and has low communication and computation requirements by utilizing elliptic curve cryptosystem. A review and analysis of the JCL-scheme is given in this section.

2.1. The JCL-scheme

The JCL-scheme [9] consists of five phases: parameter generation, registration, pre-computation, log-in, and the password-changing phase. Descriptions of these phases are given below.

Parameter Generation Phase

The related parameters in this scheme are as follows:

(1) The server selects three numbers: a larger prime number P and two field elements (a, b). Where $a \in Z_P$ and $b \in Z_P$ must satisfy $4a^3 + 27b^2 (\text{mod} P) \neq 0$, and the elliptic curve equation is defined as: $E_P : y^2 = x^3 + ax + b$.

(2) The server generates a point G from order n, and satisfies $n \times G = O$.

(3) The server selects a random number x_s to be the private key, and computes the public key $P_S = (x_s \times G)$.

(4) The server publishes the parameters (P_S, P, E_P, G, n).

Registration Phase

The user will use the smart card to register and send identification information to the server. The server will then verify the user. Descriptions of these steps are as follows:

(1) The user will select a random number *b*, and $\{ID_i, h(PW_i \| b)\}$ will be passed to the server.

(2) After the server receives the message, it will calculate $b_i = E_s(h(PW_i||b) ||ID_i||CI_i|| h(ID_i||CI_i||h(PW_i||b)))$ and $V_i = h(ID_i, s, CI_i)$ where ID_i is the user's identity and CI_i is the card number. The server will store $\{ID_i, CI_i\}$ in the internal registry. Finally, (ID_i, CI_i, b_i, V_i) is returned to the user.

Pre-computation Phase

The smart card chooses a random number *r* and calculates $e = (r \times G)$ and $c = (r \times P_S) = r \times x \times G$. Then (*e*, *c*) stored in card's memory. In the log-in phase, (*e*, *c*) will also be used.

Log-in Phase

If the user *i* wants to log-in to the server, he will cooperatively perform the following steps:

Step 1: The smart card calculates $E_{V_i}(e)$ and sends $E_{V_i}(e)$ and b_i to the server. The server uses the secret key *s* to decrypt b_i, in other word, $D_s(b_i) = (ID_i \| CI_i \| h(PW_i \| b))$, and calculates $V_i = h(ID_i, s, CI_i)$ to $D_{V_i} = (E_{V_i}(e)) = e$. Then, the server will verify the following things:

- Is CI_i stored in the registration table?
- Is ID_i in the registration?

Step 2: If any of the above checks are false, the server revokes the agreement. If the above verifications are true, the server chooses a random number *u* and calculates $c = (r \times P_S) = r \times x \times G$ and $M_S = h(c \| u \| V_i)$. Then, the server sends (*c*, M_s) to the smart card.

Step 3: The smart card calculates and checks M_S. If $M_s = h(c \| u \| V_i)$, the smart card calculates $M_U = h(h(PW_i||b)||V_i||c||u)$ and a session key $S_k = h(V_i,c,u)$ and then sends M_U to the server.

Step 4: The server checks M_U. If $M_U = h(h(PW_i||b)||V_i||c||u)$, the server calculates a session key $S_k = h(V_i,c,u)$.

Password-Changing Phase

If the user *i* wants to change his password, the smart card can encrypt the password changing message using the session key that is produced in the log-in phase. To do so, the smart card selects a random number *b** and produces another new password PW_i^* and sends $E_{sk}(ID_i,h(PW_i*||b*))$ to the server. After the server receives the message, it recalculates $b_i^* = E_s(h(PWi*||b*)||ID_i||CI_i||h(ID_i||CI_i||h(PW_i*||b*)))$ and sends $E_{Sk}(b_i*)$ to the smart card. The smart card will decrypt b_i* using a session key and store it in its memory.

2.2. Security analysis of the JCL-scheme

Juang, *et al.*, proposed many insightful security analyses to their scheme. They also proposed a solution for the issue of lost cards and minimized system information disclosure by using card revocation. As mentioned before, the system may be compromised by extracting information from the smart card in order to falsify server authentication.

Specifically, in the case of known ID_i and CI_i (these messages are stored on the smart card), the attacker will attempt to solve $V_i = h(ID_i, s, CI_i)$. The attacker can seek out the secret server key *s* using offline attack. After the secret value *s* is known, the attacker can freely tamper with the internal value of b_i, compromising the security of the entire system.

3. The Proposed Scheme

We improve on JCL-scheme and propose an enhanced password-authentication key agreement. This scheme not only maintains all the benefits of the JCL-scheme but also can enhance the security of the server when the smart card contents are disclosed. Our proposed scheme also consists of the same five phases: parameter generation, registration, pre-computation, log-in, and password-changing.

Parameter Generation Phase

In this phase, the proposed methods modeled after JCL-scheme.

(1) The server selects three numbers: a larger prime number *P* and two field elements (*a, b*). Where $a \in Z_P$ and $b \in Z_P$ must satisfy $4a^3 + 27b^2 (\mathrm{mod} P) \neq 0$, and the elliptic curve equation is defined as: $E_P : y^2 = x^3 + ax + b$.

(2) The server generates a point *G* from order *n*, and satisfies $n \times G = O$.

(3) The server selects a random number x_s to take the private key, and computes the public key $P_S = (x_s \times G)$.

(4) The server publishes the parameters (P_S, P, E_P, G, n).

Registration Phase

The user can use the smart card to send identification information for the server to authenticate. Descriptions of these steps (as depicted in Figure 1) are as follows:

Step 1: The smart card chooses a random number *b* and calculates Eq.(1).

$$T_1 = h(PW_i || b^{-1}). \quad (1)$$

Then the smart card sends $\{ID_i, h(PW_i || b), T_1\}$ to the server.

Step 2: The server chooses another random number S_2 and calculates Eqs.(2-4).

$$T_2 = T_1 \times S_2^{-1} \quad (2)$$

$$b_i = E_{S_1} (h(PW||b)||T_2 ||ID_i||CI_i|| h(ID_i||CI_i||h(PW_i||b))) \quad (3)$$

$$V_i = h(ID_i, T_1, CI_i) \quad (4)$$

Then, the server issues credentials to user *i* that contains parameters (ID_i, CI_i, b_i, V_i).

Step 3: The user receives (ID_i, CI_i, b_i, V_i) and then stores these parameters and *b* into the smart card.

Pre-computation Phase

The smart card chooses a random number r and calculates:

$$e = (r \times G) \tag{5}$$

$$c = (r \times P_S) = r \times x \times G \tag{6}$$

Then (e, c) is stored in card memory for use in the log-in phase.

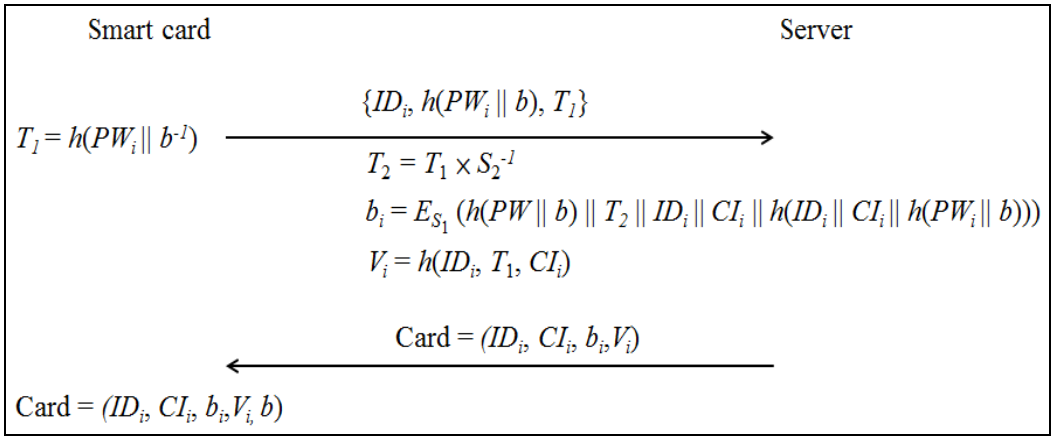

Figure 1. Registration and Pre-computation Phase of the Proposed Scheme

Log-in Phase

The user i wants to login to the server and must use his own smart card and password. Descriptions of these steps (as depicted in Figure 2) are as follows:

Step 1: After calculating $E_{V_i}(e)$, the smart card sends $E_{V_i}(e)$ and b_i to the server.

Step 2: The server decrypts b_i using the secret key S_1 and obtains $D_{S_1} = (T_2||ID_i||CI_i||h(PW_i||b)) = b_i$, and calculates Eq.(7) and Eq.(8), respectively.

$$T_1 = T_2 \times S_2 \tag{7}$$

$$V_i = h(ID_i, T_1, CI_i) \tag{8}$$

Then, the server will verify the following:

- Is CI_i stored in the registration table?
- Is ID_i in the registration?

If any of the above verifications are false, the server revokes the agreement. If the above verifications are true, the server chooses a random number u and calculates:

$$c = (e \times x) = (r \times x \times G) \tag{9}$$

$$M_S = h(c \| u \| V_i) \tag{10}$$

Then, the server sends (c, M_S) to the smart card.

Step 3: The smart card calculates and checks M_S. If *Ms* is true, the smart card calculates:

$$M_U = h(h(PW_i||b)|| T_1||c||u) \quad (11)$$

$$S_k = h(V_i,c,u) \quad (12)$$

And then the smart card sends M_U to the server.

Step 4: The server checks M_U. If M_U is true, the server calculates a session key $S_k = h(V_i,c,u)$ and accepts the log-in request.

Password-Changing Phase

When user *i* wants to change his password, the smart card can encrypt the password changing message using the session key that is produced in the log-in phase. Then, the smart card selects a random number *b** and produces another new password PW_i^* and sends E_{S_k} (ID_i,$h(PW_i*||b*)$, T_1) to the server. After the server receives the message, it recalculates $b_i^* = E_{S_1}(h(PWi*||b*)||T_2||ID_i||CI_i||h(ID_i||CI_i||h(PW_i*||b*)))$ and sends E_{S_k} (b_i*) to the smart card. The smart card will decrypt b_i* using a session key and store b_i^* and *b** in its memory (as depicted in Figure 3).

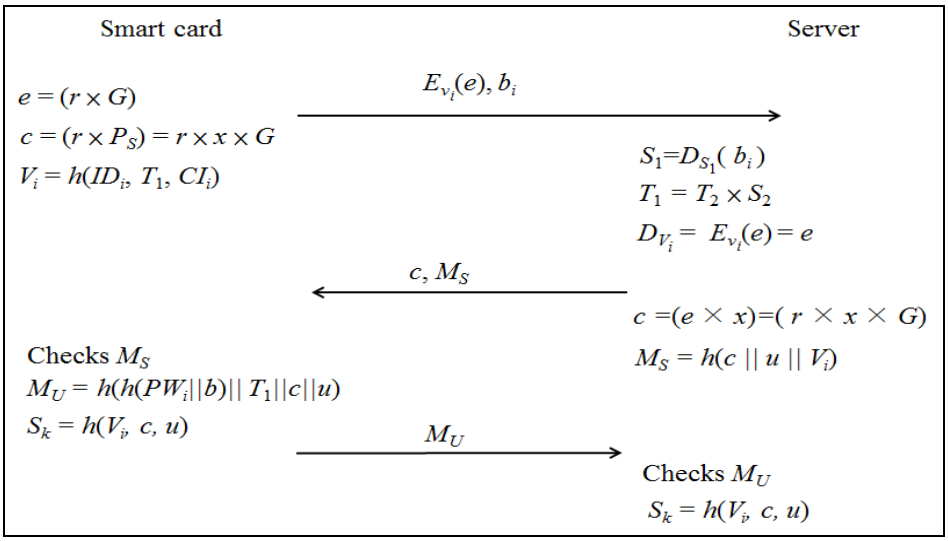

Figure 2. Log-in Phase of the Proposed Scheme

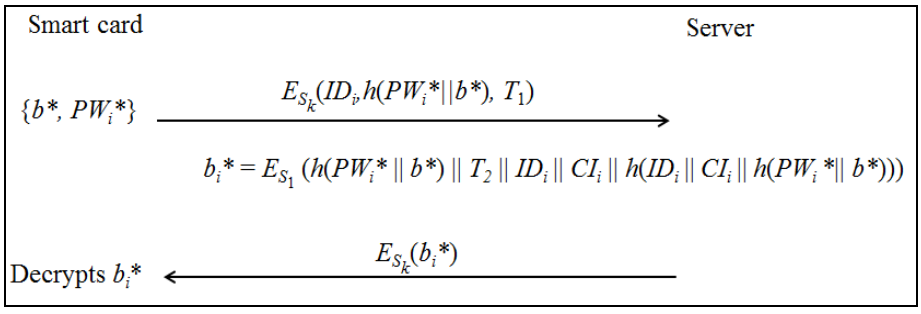

Figure 3. Password Changing Phase of the Proposed Scheme

4. Security Analysis and Comparison

In this section, we will analyze the security of our proposed scheme and make some comparisons with related schemes.

4.1. Security Analysis

In this paper, our proposed scheme provides the same benefits as JCL-scheme [9] but also improves upon their scheme. Though the approaches are similar, disclosing the information on a smart card is catastrophic to JCL-scheme leads to total compromise. We discuss two different aspects of our approach:

- Lost smart card

Assume the attacker accesses the smart card and wants ascertain internal value b_i. Value b_i cannot be decrypted without possessing the secret server key S_1. In the case of known ID_i and CI_i, if the attacker tries to calculate $V_i = h(ID_i, T_1, CI_i)$, the value T_1 is required. In order to obtain T_1, the attacker needs to know the user password PW_i in $h(Pw_i||b^{-1})$. Disclosure of the information on the smartcard still requires additional information in order to be of any value.

- Mutual authentication

In the log-in phase of our proposed scheme, the server sends M_s to the smart card. After receiving M_s, the smart card verifies it is true or false. The server can check if $h(PW_i||b)$ in M_U is equal to $h(PW_i||b)$ in b_i. If it is not, the server sends a wrong password message back to the user.

- Preventing the Replay Attack

What is replay attack? That is when an attacker tries to imitate the user to log in to the server by resending the messages transmitted between the user and the server. In our proposed scheme, we use random numbers to prevent this kind of attack. The smart card chooses a random number r and calculates $e = (r \times G)$ and $c = (r \times P_S) = r \times x \times G$ in the pre-computation phase and then sends it to the server in the log-in phase. The second random number u is chosen by the server.

- Security of secret keys

In our proposed scheme, we use two secret keys (S_1, S_2). The server decrypts b_i using the secret key S_1, and calculates T_1 from secret key S_2 and T_2. In our scheme, assuming the attacker holds the user's card and uses offline attack to obtain the server key, it will not result in increased risk to the entire system. For revocation, we use Juang et al.'s mechanism to revoke the card to ensure the privacy of the user.

4.2. Comparison

The following table compares the properties of the proposed scheme and previous schemes [3, 4, 5, 6, 9, 10]:

C1: low communication and computation cost
C2: no password table
C3: users can choose the passwords
C4: no time-synchronization problem
C5: mutual authentication
C6: revoking a lost card without changing the user's identity

C7: identity protection
C8: session key agreement
C9: preventing offline dictionary attack against the smart card information

Table 1. Properties of the Proposed Scheme versus Previous Schemes

	Hwang &Li scheme	Fan et al scheme	Juang scheme	Sun scheme	Chien et al scheme	Juang et al scheme	Our Scheme
C1	X	O	O	O	O	O	O
C2	O	O	O	O	O	O	O
C3	X	X	O	X	O	O	O
C4	X	X	O	X	X	O	O
C5	X	O	O	X	O	O	O
C6	X	X	X	X	X	O	O
C7	X	X	X	X	X	O	O
C8	X	O	O	X	X	O	O
C9	X	X	X	X	X	X	O

5. Conclusion

In this paper, we review JCL-scheme [9] and discuss the major drawbacks of their scheme. Then we proposed an improvement scheme that not only maintains all the benefits of the JCL-scheme but also enhances the security of the server when the server key is disclosed. In our scheme, even if the attacker holds the user's card, and mounts an offline attack to obtain the server key, it will not result in risk to the entire system. We use Juang, *et al.*'s mechanism to revoke cards and ensure the privacy of the user. Possession of a smart card does not allow knowledge of the second secret key in the server, so the attacker cannot break the security of the system.

Acknowledgements

This work was supported by NSC 101-2221-E-224-100.

References

[1] A. Jurisic and A. Menezes, "Elliptic Curves and Cryptography", **(1997)**, pp. 1–13.
[2] D. Nguyen, S. Oh and B. You, "A framework for Internet-based interaction of humans, robots, and responsive environments using agent technology", IEEE Trans. Ind. Electron., vol. 52, **(2005)**, pp. 1521–1529.
[3] Fan, Y. Chan and Z. Zhang, "Robust remote authentication scheme with smart cards", Computer Security, vol. 24, **(2005)**, pp. 619–628.
[4] H. Chien, J. Jan and Y. Tseng, "An efficient and practical solution to remote authentication: Smart card", Computer Security, vol. 21, **(2002)**, pp. 372–375.
[5] H. Sun, "An efficient remote use authentication scheme using smart cards", IEEE Trans. Consum. Electron., vol. 46, **(2000)**, pp. 958–961.
[6] H. Hwang and L. Li, "A new remote user authentication scheme using smart cards", IEEE Trans. Consum. Electron., vol. 46, **(2000)**, pp. 28-30.

[7] K. Saeed and M. Nammous, "A speech-and-speaker identification system: Feature extraction, description, and classification of speech-signal Image", IEEE Trans. Ind. Electron., vol. 54, **(2007)**, pp. 887–897.
[8] N. Koblitz, A. Menezes and S. Vanstone, "The state of elliptic curve cryptography", Designs, Codes Cryptography, vol. 19, **(2000)**, pp. 173–193.
[9] W. S. Juang, S. T. Chen and H. T. Liaw, "Robust and Efficient Password-Authenticated Key Agreement Using Smart Cards", IEEE Transactions on Industrial Electronics, vol. 55, **(2008)**, pp. 2551-2556.
[10] W. Juang, "Efficient password authenticated key agreement using smart cards", Computer Security, vol. 23, **(2004)**, pp. 167–173.
[11] W. Ku and S. Chen, "Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards", IEEE Trans. Consum. Electron., vol. 50, **(2004)**, pp. 204–207.
[12] W. Yang and S. Shieh, "Password authentication schemes with smart cards", Computer Security, vol. 18, **(1999)**, pp. 727–733.

Authors

Kai Chain

He was born in Kaohsiung, Taiwan, on March 13, 1975. He received the M.S. degree in Electrical Engineering from National Taiwan University in 2001-2003. He is a lecturer in the Department of Computer and Information Science at the Republic of China Military Academy. He is currently pursuing his Ph.D. degree in Cryptography from the Institute of Computer Science and Communication Engineering at National Cheng Kung University under Profs. Chi-Sung Laih and Jar-Ferr Yang. His research interests include Network and Information Security, with a concentration on applied Cryptography.

Wen-Chung Kuo

He received the B.S. degree in Electrical Engineering from National Cheng Kung University and M.S. degree in Electrical Engineering from National Sun Yat-Sen University in 1990 and 1992, respectively. Then, He received the Ph.D. degree from National Cheng Kung University in 1996. Now, he is an associate professor in the Department of Computer Science and Information Engineering at National Yunlin University of Science & Technology. His research interests include steganography, cryptography, network security and signal processing.

Jiin-Chiou Cheng

He was born in 1961. He received his M.S. degree in Communication Engineering from National Chiao Tung University, Taiwan, R.O.C. in 1985 and Ph.D. degree in Electrical Engineering form National Cheng Kung University in 2009. He is an associate professor in the Department of Computer Science and Information Engineering at Southern Taiwan University from 1990. He is engaged in the research of application of Elliptic Curve Cryptography. His research interests also include network security and Stegography.