Strong Remote User Authentication Scheme using Smart Cards

Rafael Martínez-Peláez[1], Francisco Rico-Novella[1], Cristina Satizábal[2], Jacek Pomykala[3]

[1]Department of Telematics Engineering
Techincal University of Catalonia, Barcelona, Spain
rafaelm@entel.upc.edu
f.rico@entel.upc.edu
[2]Departmento de Automática y Electrónica
Universidad Autónoma de Occidente, Cali, Colombia
icsatizabal@uao.edu.co
[3]Faculty of Mathematics, Informatics and Mechanics
Warsaw University, Poland
pomykala@mimuw.edu.pl

Abstract: Recently, Hu, Niu, and Yang proposed a remote user authentication scheme using smart cards. Although this scheme prevents several attacks, the scheme fails to establish a session key. Moreover, the scheme requires counters-synchronization and the server maintains a verification table. In order to remedy these drawbacks, this paper proposes a remote user authentication scheme which provides strong security with low communication- and computational-cost. In addition, the proposed scheme does not maintain a verification table, does not need synchronization, and establishes a session key. Furthermore, the security analysis and performance evaluation proved that the proposed scheme is feasible for smart cards.

1 Introduction

Remote user authentication is a key issue for electronic activities due to the absence of physical contact. Now a day, the most popular remote user authentication scheme is the classical username/password. Unfortunately, this scheme does not provide strong security to electronic activities because it requires one factor security to carry out the user's authentication process. Moreover, the server must maintain a verification table, making it vulnerable to insider attack.

An alternative solution to username/password scheme is remote user authentication scheme using smart cards. This schemes provide more security because the attacker must obtain the user's smart card and password.

In the literature, we can find many remote user authentication schemes using smart cards [CW91, HCL90, KC09, KC04, LLH02, LSH03, WL06]. In 2002, Chien et al. proposed a remote user authentication scheme using smart cards [CJT02]. The merits of Chien et al.'s scheme are: their scheme requires low computational- and communication-cost, and provide mutual authentication between the user and the server. Although, the scheme does not maintain a verification table, the server must maintain a table to store the users' *ID* to check its validity. However, Hsu demonstrated that Chien et al.'s scheme is vulnerable to parallel session attack [Hsu03]. Das et al. proposed a dynamic ID-based remote user authentication scheme using smart cards [DSG04]. They introduced the concept of dynamic ID-based which prevents that an attacker can know the user's *ID*. However, the scheme is susceptible to insider, masquerade, and server spoofing attacks [Wa09]. Hu et al. proposed a remote user authentication scheme in [HNY07]. Their scheme provides strong security with low computational- and communication-cost. However, Hu et al.'s scheme fails to establish a session key between the user and the server. Moreover, the server must maintain a verification table.

In this paper, we propose a secure remote user authentication scheme which achieves the essential requirements for remote user authentication schemes [LLH02, WL06] – single registration, low computational, no verification table, update password securely, mutual authentication, and key agreement –. In addition, the scheme can resist insider, leak of password, masquerade, replay, parallel, and server spoofing attacks, making it secure for electronic activities.

The rest of this paper is organized as follows. In Section 2, we give a brief review on Chien et al.'s scheme, Das et al.'s scheme, and Hu et al.'s scheme. In section 3, we describe our scheme. The security analysis of the proposed scheme is discussed and compared in Section 4. The performance analysis of our scheme is discussed and compared in Section 5. Finally, conclusions are given in Section 6.

2 Related Works

In this section, we describe briefly Chien et al.'s scheme, Das et al.'s scheme, and Hu et al.'s scheme. This section focuses on the registration phase, the login phase, and the verification phase. The notations used in this paper are described in Table 1.

U	The user
S	The server
ID	Unique identity of U
PW	Unique password of U
$h(\cdot)$	A one-way hash function
$SK(\cdot)$	A symmetric encryption function
x, y	Secret keys of S
$\oplus$	Exclusive-or operation
$\|\|$	String concatenation operation

Table 1: Notations

2.1 Chien et al.'s Scheme

In the registration phase, U_i must submit her PW_i and ID_i to S, through a secure channel. Then, S computes $A_i = h(ID_i \oplus x) \oplus PW_i$ and sends $(A_i, h(\cdot))$ to U_i, through a secure channel.

In the login phase, U_i keys her ID_i^* and PW_i^*. Then, her smart card computes $B_i = A_i \oplus PW_i$ and $CID_i = h(B_i \oplus T)$, where T is the current date and time of U_i's device. Finally, the smart card sends the login request message (ID_i, T, CID_i) to S.

In the verification phase, upon receiving the login request message (ID_i, T, CID_i), S checks the validity of ID_i and verifies the time interval between T and T^*, where T^* is the arrival time of the message. If the verification process is correct, S computes $B_i^* = h(ID_i \oplus x)$ and $CID_i^* = h(B_i^* \oplus T)$. Then, S checks whether or not CID_i^* is equal to CID_i. If the verification fails, S rejects the request; otherwise, S computes $C_j = h(B_i^* \oplus T_2)$, where T_2 is the current date and time of S. Finally, S sends (C_j, T_2) to U_i. Upon receiving the acknowledgement message (C_j, T_2), U_i checks the validity of T_2 and verifies whether or not C_j is correct. If C_j is correct, the identity of S is assured; otherwise, U_i disconnects the connection.

2.2 Das et al.'s Scheme

In the registration phase, U_i chooses freely her PW_i and sends it to S, through a secure channel. Then, S computes $A_i = h(PW_i) \oplus h(x)$. Finally, S sends $(A_i, y, h(\cdot))$ to U_i, through a secure channel.

In the login phase, U_i keys her PW_i^*. Then, her smart card computes $CID_i = h(PW_i) \oplus h(A_i \oplus y \oplus T)$, where T is the current date and time of U_i's device, $B_i = h(CID_i \oplus h(PW_i))$, and $C_i = h(T \oplus A_i \oplus B_i \oplus y)$. Finally, the smart card sends the login request message (A_i, CID_i, C_i, T) to S.

In the verification phase, upon receiving the login request message (A_i, CID_i, C_i, T), S verifies the validity of the time delay between T and T^*, where T^* is the arrival time of the message. If the verification process is correct, S computes $h(PW_i) = CID_i \oplus h(A_i \oplus y \oplus T)$, $B_i^* = h(CID_i \oplus h(PW_i))$, and $C_i^* = h(T \oplus A_i \oplus B_i^* \oplus y)$. Then, S checks whether or not C_i^* is equal to C_i. If it holds, S accepts the login request; otherwise, S rejects the login request.

2.3 Hu et al.'s Scheme

In the registration phase, U_i chooses freely her ID_i, PW_i, random number b. Then, her smart card computes $h(b \oplus PW_i)$. Finally, her smart card sends $(ID_i, h(b \oplus PW_i))$ to S, through a secure channel. Then, S creates an entry for U_i in the account database and stores U_i, e, T_i, and O_i, where e is a random number for each new U_i, T_i is the timestamp when U_i is registered by S, and O_i is the U_i's counter. Then, S computes $A_i = h(ID_i \| T_i)$ and $B_i = h(A_i \oplus x) \oplus h(b \oplus PW_i)$. Finally, S sends $(B_i, ID_i, e, h(\cdot), O_i)$ to U_i, through a secure channel. Upon receiving the message $(B_i, ID_i, e, h(\cdot), O_i)$, U_i computes $C_i = h(PW_i)$, and stores b and C_i in her smart card. Note that U_i's smart card contains b, C_i, B_i, ID_i, e, O_i, and $h(\cdot)$.

In the login phase, U_i keys her ID_i^* and PW_i^*. Then, her smart card computes $C_i^* = h(PW_i)$ and checks whether or not C_i^* and C_i are equal. If the verification process is correct, the legality of U_i is assured, and the smart card computes $D_i = B_i \oplus h(b \oplus PW_i)$, $E_i = h(e \| O_i)$, and $CID_i = h(D_i \oplus E_i \oplus e)$. Finally, U_i sends (ID_i, CID_i, E_i) to S.

In the verification phase, upon receiving the login request message (ID_i, CID_i, E_i), S verifies the validity of ID_i. If it is correct, S computes $E_i^* = h(e \| O_i)$, $A_i^* = h(ID_i \| T_i)$, $h(A_i^* \oplus x)$ and $CID_i^* = h(h(A_i^* \oplus x) \oplus E_i^* \oplus e)$. Then, S checks whether or not CID_i^* and CID_i are equal. If the verification process is correct, S accepts U_i's login request, and computes and stores $O_i = O_i + 1$. Finally, S computes and sends $F_j = h(h(A_i \oplus x) \oplus (e \| O_i))$ to U_i. Upon receiving the acknowledgement message (F_i), U_i computes $O_i = O_i + 1$ and $F_i^* = h(D_i \oplus (e \| O_i))$. Then, U_i checks whether or not F_i^* and F_i are equal. If the verification process is correct, U_i successfully authenticates S and stores $O_i = O_i + 1$.

3 Proposed Scheme

In this section, we describe a new remote user authentication scheme. The scheme consists of four phases – registration, login, verification, and password change –.

3.1 Registration Phase

In this phase, U_i must carry out the registration process with S. The process is as follows: she keys her identity ID_i and password PW_i, and her smart card computes and submits $(h(ID_i \| PW_i), ID_i)$ to S, through a secure channel. Then, S performs the following operations:

1. Computes $A_i = h(h(ID_i) \oplus x)$.
2. Computes $B_i = A_i \oplus h(ID_i \| PW_i) \oplus h(ID_i)$.
3. Computes $C_i = h(A_i)$.
4. Computes $D_i = h(h(ID_i \| PW_i) \oplus y)$.
5. Sends (B_i, C_i, D_i) to U_i through a secure channel.

Note that U_i's smart card stores (B_i, C_i, D_i, h(·)).

3.2 Login Phase

In this phase, U_i sends a login request message to S when she wants to access the resources of S. U_i keys her ID_i^* and PW_i^* and her smart card performs the following operations:

1. Computes $A_i^* = B_i \oplus h(ID_i \| PW_i) \oplus h(ID_i)$.
2. Computes $C_i^* = h(A_i^*)$.

Then, the smart card checks whether or not C_i^* and C_i are equal. If the verification process is correct, the legality of U_i is assured. Then, the smart card randomly chooses a nonce N_i and performs the following operations:

1. Computes $E_i = A_i \oplus N_i$.
2. Computes $CID_i = h(ID_i \| PW_i) \oplus N_i$.
3. Computes $F_i = h(A_i \| D_i \| N_i)$.
4. Sends the login request message ($h(ID_i)$, E_i, CID_i, F_i) to S.

3.3 Verification Phase

Upon receiving the login request message ($h(ID_i)$, E_i, CID_i, F_i), S performs the following operations:

1. Computes $A_i^* = h(h(ID_i) \oplus x)$.
2. Computes $N_i^* = A_i^* \oplus E_i$.
3. Computes $h(ID_i \| PW_i)^* = CID_i \oplus N_i^*$.
4. Computes $D_i^* = h(h(ID_i \| PW_i)^* \oplus y)$.
5. Computes $F_i^* = h(A_i^* \| D_i^* \| N_i^*)$.

Then, S checks whether or not F_i^* is equal to F_i. If it does not hold, S rejects the login request. In the other case, S chooses a nonce N_j and performs the following operations:

1. Computes $SK_{ij} = h(A_i \| D_i \| N_i \| N_j)$.
2. Computes $G_j = SK_{ij}(N_i \oplus N_j)$.
3. Sends (N_j, G_j) to U_i.

After receiving the acknowledgement message (N_j, G_j), U_i performs the following operations:

1. Computes $SK_{ij}^* = h(A_i \| D_i \| N_i \| N_j)$.
2. Computes $G_j^* = SK_{ij}^*(N_i \oplus N_j)$.
3. Computes $(N_i \oplus N_j)^*$.

Then, U_i checks whether or not $(N_i \oplus N_j)^*$ is correct. If it is correct, the identity of S can be assured. After finishing verification phase, U_i and S have a session key SK_{ij}.

3.4 Password Change Phase

When U_i wants to change her password PW_i, she keys her ID_i^* and PW_i^*, and she requests to change the password to new one PW_{inew}. Then, the smart cart computes $A_i^* = B_i \oplus h(ID_i \| PW_i) \oplus h(ID_i)$ and $C_i^* = h(A_i^*)$, and checks whether or not C_i^* and C_i are equal. If the verification process is correct, U_i can key a new password PW_{inew}. The smart card computes $B_{inew} = A_i \oplus h(ID_i \| PW_{inew}) \oplus h(ID_i)$. The new value B_{inew} replaces the old B_i, making possible the password change.

4 Security Analysis

In this section, we describe the security of the proposed scheme. We assume that an attacker can extract the secret values stored in the smart card [Koc99, MDS02]. The comparison of our scheme and others schemes is summarized in Table 2. It demonstrates that our scheme can resistance several attacks to provide strong security.

4.1 Insider Attack

If an attacker obtains B_i, C_i, and D_i from U_i's smart card, she cannot extract sensitive information, like PW_i, ID_i, A_i, x, or y from B_i, C_i, and D_i, because it is computationally infeasible to invert the one-way hash function $h(\cdot)$. Moreover, she cannot extract A_i from B_i without the knowledge of PW_i and ID_i. Furthermore, the server does not maintain any verification table. In addition, if the attacker is a legal user U_i, she cannot obtain x and y from her smart card.

4.2 Leak of Password Attack

In the proposed scheme, if the attacker obtains the U_i's smart card, she cannot extract U_i's password PW_i using B_i, C_i, D_i or other combination of them.

4.3 Masquerade Attack

If the attacker intercepts a user's login request message ($h(ID_i)$, E_i, CID_i, F_i), she cannot masquerade as a legal user U_i because she cannot breach A_i, $h(ID_i \| PW_i)$, and D_i from the intercepted login request message. The attacker could not forge a login request message to pass server's authentication, because she does not know a valid A_i, $h(ID_i \| PW_i)$, and D_i. Although, the attacker is a legal user U_i, she cannot compute A_i without the knowledge of x.

Let us suppose that the attacker knows B_i, C_i, and D_i, and previous U_i's login request message (h(ID_i), E_i, CID_i, F_i), she cannot forge a valid login request message because she cannot extract h(ID_i || PW_i) from CID_i without the knowledge of N_i. Although, the attacker knows h(ID_i) she cannot extract A_i from B_i without a valid h(ID_i || PW_i). Thus, the scheme can resist the masquerade attack.

4.4 Parallel Session Attack

If the attacker intercepts the acknowledgement message (N_j, G_j), she cannot re-use G_j to create a valid login request message (h(ID_i), E_i, CID_i, F_i), because the acknowledgement message does not contain information to construct a valid login request message.

4.5 Replay Attack

Our scheme uses nonce instead of timestamp to withstand replay attacks. Suppose that the attacker has intercepted a previous login request message (h(ID_i), E_i, CID_i, F_i) from U_i, the attacker can replay the same message to S. Upon receiving the login response message (N_j, G_i), the attacker cannot compute the session key SK_{ij} to establish a secure channel with S without knowing A_i, D_i, and N_i. Thus, the scheme can prevent replay attack.

4.6 Server Spoofing Attack

An attacker cannot masquerade as a legal server S because she cannot compute A_i and D_i without knowing x and y. Moreover, she cannot extract N_i and h(ID_i || PW_i) without the knowledge of A_i. Furthermore, she cannot compute a correct session key SK_{ij} without a valid D_i. Thus, the scheme can prevent server spoofing attack.

Resistance to	Chien et al.	Das et al.	Hu et al.	Ours
Insider attack	Yes	No	Yes	Yes
Leak of password	Yes	Yes	Yes	Yes
Masquerade attack	Yes	No	Yes	Yes
Parallel session attack	No	-	Yes	Yes
Replay attack	Yes	Yes	Yes	Yes
Server spoofing attack	Yes	No	Yes	Yes

Table 2: Security comparison between our scheme and other schemes

5 Performance Analysis

In this section, we summarize some performance issues of our scheme. We compare our scheme with related schemes in terms of computational cost and storage capacity.

Due to the limited computational power of smart cards, the remote user authentication scheme must take computational cost evaluation into consideration. In order to carry out the computational cost evaluation, we use the following notations: T_h and T_{sym} are defined as the execution times for one-way hash functions and symmetric operations, respectively. Because exclusive-or operation requires very low execution time, it is usually neglected considering its computational cost. The time complexity associated with the different operations can be expressed as exclusive-or << T_h < T_{sym}.

The computational cost is defined as the total time of various operations executed in each step. According to the above definition, the computation cost in the registration phase is $5T_h$ time. The customer requires T_h time while the server requires $4T_h$ time. In the login and verification phases, the user requires $5T_h + T_{sym}$ time and the server requires $4T_h + T_{sym}$ time.

In addition, we evaluate and compare our scheme in terms of storage capacity. We assume that the output size of a one-way hash function, random numbers and secret keys are 160-bit length, timestamps and counters are 40-bit length, and identity is 32-bit length; so the memory needed in the user's smart card is 480(3*160) bits and the server requires 320(2*160) bits to store the two secret keys x and y. Table 3 shows the performance comparison of our scheme and related schemes

		Chien et al.	Das et al.	Hu et al.	Ours
Computational cost in the registration phase	U_i	-	-	$2T_h$	T_h
	S	T_h	$2T_h$	$1T_h$	$4T_h$
Computational cost in the login and verification phases	U_i	$2T_h$	$4T_h$	$5T_h$	$5T_h + T_{sym}$
	S	$3T_h$	$3T_h$	$4T_h$	$4T_h + T_{sym}$
Communication cost in the login and verification phases	U_i	232bits	520bits	352bits	640bits
	S	200bits	-	160bits	320bits
Storage capacity	U_i	160bits	320bits	712bits	480bits
	S	160 + (40 * n)bits	320bits	160 + (272 * n)bits	320bits

Table 3: Performance comparison between our scheme and other schemes

Table 3 shows that our scheme require more computational cost than Chien et al.'s scheme and Das et al.'s scheme in registration phase. However, our scheme is resistance to insider, leak of password, and masquerade attacks. Note that server S computes four one-way hash functions while user U_i computes one-way hash function, during the registration phase, considering the computational power of current servers, the execution time of four one-way hash functions is extremely very low.

In addition, Table 3 shows that our scheme requires two symmetric encryption operations, in the verification phase, while Hu et al.'s scheme does not require any symmetric operation. However, our scheme establishes a session key between U_i and S, and prevents parallel session and replay attacks.

The storage capacity analysis demonstrated that our scheme is feasible for smart cards because U_i's smart card requires 480 bits while Hu et al.'s scheme requires 712 bits. Furthermore, the server requires 320 bits to store secret keys x and y, and does not maintain a verification table while Chien et al.'s scheme and Hu et al.'s scheme require a verification table to store ID_i, and U_i, e, T_i and O_i, respectively.

Moreover, we summarize the functionality of the proposed scheme in Table 4. It demostrates that our scheme is efficient for electronic activities.

	Chien et al.	Das et al.	Hu et al.	Ours
Mutual authentication	Yes	No	Yes	Yes
No verification table	No	Yes	No	Yes
No time-synchronization	No	No	Yes	Yes
No counters-synchronization	Yes	Yes	No	Yes
Secure password change	No	Yes	Yes	Yes
Session key agreement	No	No	No	Yes
Single registration	Yes	Yes	Yes	Yes
Two factor security	Yes	No	Yes	Yes

Table 4: Efficiency comparison between our scheme and other schemes

Table 4 shows that our scheme achieves the essential requirement for remote user authentication schemes. The proposed scheme does not require any type of synchronization. On the other hand, Hu et al.'s scheme, Chien et al.'s scheme, and Das et al.'s scheme require synchronization which implies additional communication- and computational-cost to users.

6 Conclusions

In this paper, we have proposed a remote user authentication scheme using smart cards, which provides mutual authentication and establishes a session key between the user and the server. The security analysis proved that the proposed scheme can resist insider, leak of password, masquerade, parallel, replay, and server spoofing attacks. Moreover, we demonstrated that the proposed scheme is feasible for practical implementation, because the scheme is efficient in terms of communication-cost, computational-cost, and storage capacity. In addition, the scheme provides more services for electronic activities than Hu et al.'s scheme, Das et al.'s scheme, and Chien et al.'s scheme.

Acknowledgement

This work has been partially supported by the Spanish public funded projects ARES (CONSOLIDERINGENIO-2010 CSD2007-00004) and ITACA (TSI2006-13409-C02-02), and graduate scholarship from CONACYT (Mexico).

References

[CJT02] Chien, H. Y., Jan, J. K. and Tseng, Y. M.: An Efficient and practical solution to remote authentication: smart card, Computers & Security, vol. 21, 2002; pp. 372-375.

[CW91] Chang, C. C. and Wu, T. C.: Remote password authentication with smart cards, IEE Proceedings-E, vol. 138, 1991; pp. 165-168.

[DSG04] Das, M. L., Saxena, A. and Gulati, V. P.: A Dynamic ID-based remote user authentication scheme, IEEE Transactions on Consumer Electronics, vol. 50, 2004; pp. 629-631.

[HCL90] Hwang, T., Chen, Y. and Laih, C. S.: Non-interactive password authentication without password tables. in IEEE Region 10 Conference on Computer and Communication System. 1990; pp. 429-431.

[HNY07] Hu, L. I., Niu, X. X. and Yang, Y. X.: Weaknesses and improvements of a remote user authentication scheme using smart cards, The Journal of China Universities of Posts and Telecommunications, vol. 14, 2007; pp. 91-94.

[Hsu03] Hsu, C. L.: Security of two remote user authentication schemes using smart cards, IEEE Transaction on Consumer Electronics, vol. 49, 2003; pp. 1196-1198.

[KC04] 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, vol. 50, 2004; pp. 204-207.

[KC09] Kim, S. K. and Chung, M. G.: More secure remote user authentication scheme, Computer Communications, vol. 32, 2009; pp. 1018-1021.

[Koc99] Kocher, P., Jaffe, J. and Jun, B.: Differential power analysis. In Advances in Cryptology - Crypto'99, vol. LNCS 1666, 1999; pp. 388-397.

[LLH02] Lee, C. C., Li, L. H. and Hwang, M. S.: A remote user authentication scheme using hash functions, ACM SIGOPS Operating Systems Review, vol. 36, 2002; pp. 23-29.

[LSH03] Lin, C. W., Shen, J. J. and Hwang, M. S.: Security enhancement for optimal strong-password authentication protocol, ACM SIGOPS Operating Systems Review, vol. 37, 2003; pp. 12-16.

[MDS02] Messerges, T. S., Dabbish, E. A. and Sloan, R. H.: Examining smart-card Security under the threat of power analysis attacks, IEEE Transactions on Computers, vol. 51, 2002; pp. 541-552.

[Wa09] Wang, Y. Y., et al.: A more efficient and secure dynamic ID-based remote user authentication scheme, Computer Communications, vol. 32, 2009; pp. 583-585.

[WL06] Wang, B. and Li, Z. Q.: A forward-secure user authentication scheme with smart cards, International Journal of Network Security, vol. 3, 2006; pp. 116-119.