http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 8 (3): 366-371, 2009
ISSN 1812-5638
© 2009 Asian Network for Scientific Information

Efficient Remote Mutual Authentication and Key Agreement with Perfect Forward Secrecy

[1,2]Han-Cheng Hsiang and [1]Wei-Kuan Shih
[1]Department of Computer Science, National Tsing Hua University, Taiwan, Republic of China
[2]Department of Information Management, Vanung University, Taiwan, Republic of China

Abstract: This study will demonstrate both Juang's scheme and Shieh-Wang's scheme do not provide perfect forward secrecy and are vulnerable to a privileged insider's attack. Besides, their scheme has the problem of slow wrong password detection and user cannot change his password freely. To remedy these flaws, this study proposes an efficient remote mutual authenticated and key agreement scheme with perfect forward secrecy. The proposed scheme not only provides perfect forward secrecy but also satisfies all the security requirements needed in remote mutual authentication and key agreement scheme.

Key words: Password, privileged insider's attack, smart card, slow wrong password detection

INTRODUCTION

In distributed computing environment, secure communication in insecure communication networks is a very important issue. Hence, user authentication and secret key distribution become the most important security service for communication networks.

Remote user authentication scheme allows a server to check the legitimacy of a remote user through insecure network. In addition, a smart card based remote mutual authentication scheme is very useful to authenticate remote users (Juang, 2006; Wang *et al.*, 2007). Hwang *et al.* (1990) initially proposed a non-interactive password authentication scheme and its enhanced version, which additionally uses smart cards. Since, then Tan and Zhu (1999), Yang and Shieh (1999) and Hwang and Li (2000) have proposed new schemes and improved the efficiency and security of remote authentication. For evaluating a strong protocol, perfect forward secrecy is considered to be an important security issue. A protocol providing perfect forward secrecy means that even if one entity's long-term secret key is compromised, it will never reveal any old short-term keys used previously. (Menezes *et al.*, 1997; Sun and Yeh, 2006).

In 2002, based on Sun's (2000), scheme Chien *et al.* (2002) proposed an efficient remote mutual authentication scheme using smart card allowing server and user to authenticate each other. The advantages in the scheme include freely chosen passwords, no verification tables, low communication and computation costs. However, as demonstrated by Hsu (2004) and Chien *et al.*'s (2002) scheme is vulnerable to the parallel session attack. Thereafter, Juang (2004) proposed another scheme preserving all the advantages of Chien *et al.*'s (2002) scheme. Juang's (2004) scheme is nonce based authentication and key agreement scheme. Hence, no synchronized clocks are required in the scheme. In addition, Juang's scheme generates a session key for the user and server in their subsequent communication.

Recently, Shieh and Wang (2006) pointed out the weakness of Juang's scheme and then proposed another similar scheme to improve the weakness. Shieh and Wang (2006) claimed that their scheme not only preserves all the advantages of Juang's scheme but also improves its efficiency. Later, Yoon and Yoo (2007) demonstrated that Shieh-Wang's scheme (Shieh and Wang, 2006) does not provide perfect forward secrecy (Menezes *et al.*, 1997; Sun and Yeh, 2006) and is vulnerable to a privileged insider's attack (Ku *et al.*, 2005; Ku and Chen, 2004). Besides, we find that Shieh-Wang's scheme has the problem of slow wrong password detection (Wang *et al.*, 2007) and user cannot change his password freely. These flaws also exist in Juang's (2006) scheme. To remedy these flaws, this study presented an efficient improvement on them with more security. Present scheme providing perfect forward secrecy, can withstand the previously proposed attacks. When the user inputs the wrong passwords, it can be immediately detected and the user can freely change his password. The proposed scheme not only inherits the merits of their scheme but also enhances the security of their scheme.

Corresponding Author: Han-Cheng Hsiang, Department of Computer Science, National Tsing Hua University, 101, Kuang Fu Road, Sec. 2, Hsing Chu, 300, Taiwan, Republic of China

REVIEW OF JUANG'S AND SHIEH-WANG'S SCHEMES

The notations in Table 1 are used throughout this study.

Review of Juang's scheme: Juang's (2004) scheme is based on the symmetric encryption and adopts nonce to avoid the time-synchronization problem of using time stamps. The scheme consists of two phases: the registration phase and the login and session key agreement phase.

Registration phase: Assume a user U_i submits his identity ID_i and password PW_i to the server over a secure channel for registration. The server computes $V_i = h(ID_i, x)$, $W_i = V_i \oplus PW_i$ and issues U_i a smart card containing W_i, ID_i and h( ).

Login and session key agreement phase: When U_i wants to login to the server, he inserts his smart card into a card reader and inputs his identity ID_i and password PW_i.

- The smart card computes $V_i = W_i \oplus PW_i$, then sends the message $(N_1, ID_i, E_{Vi}(ru_j, C_i))$ to the server S, where, $C_i = h(ID_i \| N_1)$, E_{Vi} denotes a symmetric encryption algorithm using V_i as the secret key, N_1 is a nonce and ru_j is a random value chosen by the smart card to generate the session key K_j
- After receiving the message, S computes $V_i = h(ID_i, x)$ and $(ru_j, C_i) = D_{Vi}(E_{Vi}(ru_j, C_i))$, where, $D_{vi}(\ )$ denotes the corresponding symmetric decryption algorithm of E_{Vi} using V_i as the secret key. After decryption, if C_i is not equal to $h(ID_i \| N_1)$, or N_1 is not fresh, the server rejects U_i's request. Otherwise, the server sends the message $E_{Vi}(rs_j, N_1+1, N_2)$ to U_i, where, N_2 is a nonce and rs_j is a random value chosen by the server to generate the session key K_j
- When U_i receives the message, the smart card decrypts and checks whether N_1+1 is in it. If yes, the smart card computes the session key $K_j = h(rs_j, ru_j, V_i)$ and sends the message $E_{Kj}(N_2+1)$ back to S
- On receiving the last message, S computes $D_{Kj}(E_{Kj}(N_2+1))$ to check whether N_2+1 is in it. If N_2+1 is found, S and U_i have achieved mutual authentication and session key agreement

Table 1: Notations

Notation	Meaning
$h(\cdot)$	Secure one-way hash function
$h_k(\cdot)$	Represents a cryptographic keyed hash function with secret k
x	The secret key maintained by the server
p	A large prime number
g	A primitive element in GF(p)
$\oplus$	Exclusive-or operation
$\|$	String concatenation operation

Review of Shieh-Wang's scheme: Shieh and Wang's (2006) scheme consists of two phases: the registration phase and the login and key agreement phase. The scheme works as follows:

Registration phase: Assume a user U_i submits his identity ID_i and password PW_i to the server over a secure channel for registration. If the request is accepted, the server computes $R_i = h(Id_i \oplus x) \oplus PW_i$ and issues U_i a smart card containing R_i and $h(\cdot)$.

Login and key agreement phase: When the user U_i wants to login to the server, he first inserts his smart card into a card reader then inputs his identity ID_i and password PW_i. The smart card then performs the following steps to begin an access session:

- Compute $a_i = R_i \oplus PW_i$
- Acquire current time stamp T_u, store T_u temporarily until the end of the session and compute $MAC_u = h(T_u \| a_i)$
- Send the message (ID_i, T_u, MAC_u) to the server and wait for response from the server. If no response is received in time or the response is incorrect, report login failure to the user and stop the session

After receiving the message (ID_i, T_u, MAC_u) from U_i, the server performs the following steps to assure the integrity of the message, respond to U_i and challenge U_i to avoid replay:

- Check the freshness of Tu. If Tu has already appeared in a current executing session of user U_i, reject U_i's login request and stop the session. Otherwise, Tu is fresh
- Compute $a_i' = h(ID_i \oplus x)$, $MAC_u' = h(T_u \| a_i')$ and check whether MAC_u' is equal to the received MAC_u. If it is not, reject U_i's login and stop the session
- Acquire the current time stamp T_s. Store temporarily paired time stamps (T_u, T_s) and ID_i for freshness checking until the end of the session. Compute $MAC_s = h(T_u \| T_s \| a_i')$ and session key $K_s = h((T_u \| T_s) \oplus a_i')$. Then, send the message (T_u, T_s, MAC_s) back to U_i and wait for response from U_i. If no response is received in time or the response is incorrect, reject U_i's login and stop the session.

On receiving the message (T_u, T_s, MAC_s) from the server, the smart card performs the following steps to authenticate the server, achieve session key agreement and respond to the server:

- Check if the received T_u is equal to the stored T_u to assure the freshness of the received message. If it is not, report login failure to the user and stop the session
- Compute $MAC_s' = h(T_u||T_s||a_i)$ and check whether it is equal to the received MAC_s. If not, report login failure to the user and stop. Otherwise, conclude that the responding party is the real server
- Compute $MAC_u'' = h(T_s||(a_i+1))$ and session key $K_s = h((T_u||T_s)\oplus a_i)$, then send the message (T_s, MAC_u'') back to the server. Note that, in the message (T_s, MAC_u''), Ts is a response to the server

When the message (T_s, MAC_u'') from U_i is received, the server performs the following steps to authenticate U_i and achieve key agreement:

- Check if the received T_s is equal to the stored T_s. If it fails, reject U_i's login request and stop the session
- Compute $MAC_u1''' = h(T_s||(a_i'+1))$ and check whether it is equal to MAC_u''. If it is not, reject U's login request and stop the session. Otherwise, conclude that U_i is a legal user and permit the user U_i's login

At this moment, mutual authentication and session key agreement between U_i and the server are achieved. From now on, the user U_i and the server can use the session key Ks in their further secure communication until the end of the access session.

SECURITY FLAWS ON JUANG'S AND SHIEH-WANG'S SCHEMES

Here, we will show that both Juang's and Shieh-Wang's scheme do not provide perfect forward secrecy (Menezes *et al*., 1997; Sun and Yeh, 2006) and are vulnerable to a privileged insider's attack (Ku *et al*., 2005; Ku and Chen, 2004). Besides, the schemes have the problem of slow wrong password detection.

Perfect forward secrecy problem: In Juang's scheme, if an attacker Eve obtains the secret key x from the compromised server and intercepts transmitted values (N_1, ID_i, E_{vi} (ru_j, C_i)) and E_{Vi} (rs_j, N_1+1, N_2), from an open network. It will be easy to obtain the information since it is exposed over an open network. Then Eve can compute $V_i = h(ID_i, x)$ by using ID_i. After decryption, Eve can obtain rs_j, ru_j and can compute the shared session key $K_j = h(rs_j, ru_j, V_i)$ by using V_i, rs_j and ru_j.

In Shieh-Wang's (2006) scheme, if an attacker Eve obtains the secret key x from the compromised server and intercepts transmitted values (ID_i, T_u, T_s), from an open network. It will be easy to obtain the information since it is exposed over an open network. Then Eve can compute $a_i = h(ID_i\oplus x)$ by using ID_i and can compute the shared session key $K_s = h((T_u || T_s)\oplus a_i)$ by using a_i, T_u and T_s.

Once Eve obtains the shared session key, by using the shared session key, Eve can get all previous communication messages. Clearly, both Juang's and Shieh-Wang's (2006) scheme do not provide perfect forward secrecy.

Privileged insider's attack: In real environments, a user uses the same password to access several servers for his convenience. If a privileged insider of the server has learned the user's password, he may try to impersonate the user to access other servers. These situations can be found in the registration phase of Juang's and Shieh-Wang's scheme. For example, U_i's password PW_i will be revealed to the remote server S because it is transmitted directly to S. Then, the privileged insider of S may try to use PW_i to impersonate U_i to login the other servers that U_i has registered with outside this system. If the targeted outside server adopts the normal password authentication scheme, it is possible that the privileged insider of the server can successfully impersonate U_i to login it by using PW_i.

Slow wrong password detection: The Juang's scheme and the Shieh-Wang's scheme also have the problem of slow wrong password detection. If U_i inputs a wrong password by mistake, the smart card still sends U_i's login request unconditionally to server. The wrong password is not detected by the remote system S checks $C_i = h(ID_i || N_1)$ at the login and session key agreement phase of Juang's scheme and checks $MAC_u' = MAC_u$ at the login and key agreement phase of Shieh-Wang's scheme. Therefore, their scheme is slow to detect the user's wrong password. In other words, the password authentication is delayed and inefficient.

THE PROPOSED SCHEME

Here, we propose a remote mutual authentication and key agreement scheme with perfect forward secrecy. The proposed scheme is an improvement to Shieh-Wang's (2006) scheme that can withstand the security flaws described in above. The system selects two large prime numbers p and q, such that q divides p-1. One generator g with order q in the Galois field GF(p), where, GF(p) is the set of integers {0, 1, . . ., p-1} with arithmetic operations defined modulo p. The proposed scheme consists of four phases: the registration phase, the login phase, the authentication and key agreement phase and the password change phase. The scheme works as follows.

Registration phase: Assume a user U_i selects a random number b, password PW_i and computes h ($b \oplus PW_i$). He submits his identity ID_i and h ($b \oplus PW_i$) to the server over a secure channel for registration. If the request is accepted, the server computes a_i = h ($ID_i \oplus x$), $R_i = a_i \oplus h(b \oplus PW_i)$ and $N = h_{ai}(h(b \oplus PW_i))$. Then issues U_i a smart card containing R_i, N, p, g, h(·) and $h_k(\cdot)$. U_i enters b into his smart card so that he does not need to remember b anymore.

Login phase: When the user U_i wants to login to the server, he first inserts his smart card into a card reader then inputs his identity ID_i, password PW_i and a large random integer d<p-1. The smart card then performs the following steps to begin an access session:

- Compute $a_i = R_i \oplus (b \oplus PW_i)$ and check whether $h_{ai}(h(b \oplus PW_i)) \oplus = N$ holds. If not, smart card terminates this session
- Acquire current time stamp T_u, store T_u temporarily until the end of the session and compute $MAC_u = h(T_u \| a_i)$
- Compute $C = g^d \oplus a_i$
- Smart card generates a random number r and computes $C_1 = a_i \oplus h(r \oplus b)$
- Send the message (ID_i, T_u, MAC_u, C, C_1) to the server S and wait for response from the server. If no response is received in time or the response is incorrect, report login failure to the user and stop the session

Authentication and key agreement phase: After receiving the message (ID_i, T_u, MAC_u, C, C_1) from U_i, the server S performs the following steps to assure the integrity of the message, respond to U_i and challenge U_i to avoid replay:

- Check the freshness of T_u. If T_u has already appeared in a current executing session of user U_i, reject U_i's login request and stop the session. Otherwise, T_u is fresh
- Compute $a_i' = h(ID_i \oplus x)$, $C_1' = a_i' \oplus C_1$, $MAC_u' = h(T_u \| a_i')$ and check whether MAC_u' is equal to the received MAC_u. If it is not, reject U_i's login and stop the session
- S chooses a large random integer e<p-1 and computes g^e (mod p). S retrieves g^d by computing $g^d = C \oplus a_i'$. After all parameters are known, S computes g^{de}(mod p) by raising the received g^d to e
- Acquire the current time stamp T_s. Store temporarily paired time stamps (T_u, T_s) and ID_i for freshness checking until the end of the session. Compute $MAC_s = h(T_u \| T_s \| a_i)$, $C_3 = g^e \oplus a_i'$ and $C = g^{de} \oplus C_1' \oplus K_s$, which K_s is the session key. Then, send the message (T_u, T_s, MAC_s, C_2, C_3) back to U_i and wait for response from U_i. If no response is received in time or the response is incorrect, reject U_i's login and stop the session

On receiving the message (T_u, T_s, MAC_s, C_2, C_3) from the server, the smart card performs the following steps to authenticate the server, achieve session key agreement and respond to the server:

- Check if the received T_u is equal to the stored T_u to assure the freshness of the received message. If it is not, report login failure to the user and stop the session
- Compute $MAC_s' = h(T_u \| T_s \| a_i)$ and check whether it is equal to the received MAC_s. If not, report login failure to the user and stop. Otherwise, conclude that the responding party is the real server.
- Compute $g^e = C_2 \oplus a_i'$ and derive g^{de} (mod p) by raising the received g^e to d. Next, retrieve session key K_s by computing $K_s = g^{de} \oplus h(b \oplus r) \oplus C_3$
- Compute $MAC_u'' = h(T_s \| K_s \| (a_i+1))$, then send the message (T_s, MAC_u'') back to the server. Note that, in the message (T_s, MAC_u''), T_s is a response to the server

When the message (T_s, MAC_u'') from U_i is received, the server performs the following steps to authenticate U_i and achieve key agreement:

- Check if the received T_s is equal to the stored T_s. If it fails, reject U_i's login request and stop the session
- Compute $MAC_u1''' = h(T_s \| K_s \| (a_i'+1))$ and check whether it is equal to MAC_u''. If not, reject U_i's login request and stop the session. Otherwise, conclude that U_i is a legal user and permit the user U_i's login.

At this moment, mutual authentication and session key agreement between U_i and the server are achieved. From now on, the user U_i and the server can use the session key K_s in their further secure communication until the end of the access session.

Password change phase: This phase is invoked whenever a user U_i wants to change his password PW_i with a new one, say Pw_{new}.

- U_i inserts his smart card into card reader, enters ID_i and PW_i and requests to change password
- U_i's smart card computes $a_i = R_i \oplus h(b \oplus PW_i)$ and $N^* = h_{ai}(h(b \oplus PW_i))$

- Check whether N* equals to the stored N or not. If not, reject the password change request, otherwise U_i chooses a new password PW_{new}
- Compute $R_{new} = a_i \oplus h(b \oplus PW_{new})$ and $N_{new} = h_{ai}(h(b \oplus PW_{new}))$, then stores R_{new}, N_{new} into the user's smart card and replaces the old values R_i, N, respectively. The new password is successfully updated and this phase is terminated

SECURITY ANALYSIS

Analyzes the security of present scheme as following.

The proposed scheme inherits the security features of Shieh-Wang's (2006) remote user authentication and key agreement protocol. Challenge tokens C, C_2 and C_3 are used to ensure authenticity of the server S and the freshness of the communication and prevent replay attacks. Under these circumstances, the messages in the scheme are fresh and replays of old messages can be detected. Moreover, the protocol can withstand the attacks presented in previous sections.

Claim 1: The proposed scheme can provide perfect forward secrecy.

Proof: Present scheme is based on the following well-known hard problem, which is believed infeasible to solve in polynomial time. Given a prime p, a generator g and two numbers g^d mod p and g^e mod p, try to find g^{de} mod p, it is computationally infeasible due to the Diffie-Hellman (1976) problem. We assume that the user U_i's password PW_i, the server S's secret key x are all known by an attacker. Then the attacker can decrypt C to obtain g^d and decrypt C_2 to obtain g^e. But he cannot calculate g^{de} because the difficulty is similar to solve the Diffie-Hellman problem. So, the attacker does not have any opportunity to get the session key K_S. Therefore, the session key is still secure. Hence, present scheme provides perfect forward secrecy with high security (Sun and Yeh, 2006).

Claim 2: The proposed scheme can resist privileged insider's attack.

Proof: In the registration phase of the proposed scheme, a user U_i selects a random number b, password PW_i and computes $h(b \oplus PW_i)$. He submits ID_i and $h(b \oplus PW_i)$ to the remote server S. If the privileged insider of S wants to use U_i's password to impersonate U_i to login the other servers, the action will fail. Since U_i registers to S by presenting $h(b \oplus PW_i)$ instead of PW_i, the insider of S can not directly obtain PW_i. Furthermore, as b is not revealed to S, the privileged insider of S can not obtain PW_i by performing an off-line guessing attack on $h(b \oplus PW_i)$. Thus, the proposed scheme can resist the privileged insider attack (Ku *et al.*, 2005; Ku and Chen, 2004).

Table 2: Comparisons of various security attributes

	Properties						
Schemes	1	2	3	4	5	6	7
Present scheme	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Shieh (2006)	Yes	Yes	Yes	No	No	No	No
Juang (2004)	Yes	Yes	Yes	No	No	No	No
Chien (2002)	Yes	Yes	No	No	No	No	No
Sun (2000)	No	No	No	No	No	No	No

1: Freely chosen password, 2: Mutual authentication, 3: Generate authenticated session key, 4: Perfect forward secrecy, 5: Privileged insider's attack, 6: Fast wrong password detection and 7: Secure password change

Claim 3: The proposed scheme is high efficiency in password authentication.

Proof: In login phase, if U inputs an error password PW*, the smart card computers $a_i = R_i \oplus (b \oplus PW^*)$ and checks equation $h_{ai}(h(b \oplus PW^*)) \oplus = V$ in step 1. Obviously, the result is negative when $Pw_i \neq PW^*$ and smart card terminates the login session. Thus, the wrong password will be immediately detected by smart card yet need not wait for server authenticating, which results in high efficiency.

Claim 4: The user can freely change his/her password.

Proof: In present scheme, each user can choose her/his favorite password in the registration phase. It will make users easy to remember their own passwords. The proposed scheme also provides the mechanism of changing password. When any user wants to change password, the user's smartcard will compute $N^* = h_{ai}(h(b \oplus PW))$ and verify N^* compares with the stored N in smart card, respectively. If they are equal, i.e., N* = N, the password change action will be executed. Otherwise, the smartcard rejects the password change request. Thus, the proposed scheme could effectively provide the mechanism of password change.

After above discussion, we summarize the comparisons of the proposed scheme with related schemes in Table 2.

CONCLUSIONS

This study have demonstrated that both Juang's scheme and Shieh-Wang's scheme do not provide perfect forward secrecy and are vulnerable to a privileged

insider's attack. It is also found that the schemes have the problem of slow wrong password detection and user cannot change the password freely. Then, this research presented an efficient remote mutual authenticated and key agreement scheme with perfect forward secrecy to remedy these flaws. The important merits of the proposed scheme include: no verification table in the server, users being able to choose own password freely, both the server and the user can do mutual authentication, no synchronization required, generation of a session key, perfect forward secrecy, fast wrong password detection and users being able to change the password freely. After the analysis, it is concluded that the proposed scheme is more secure than the other schemes examined in this study.

REFERENCES

Chien, H.Y., J.K. Jan and Y.M. Tseng, 2002. An efficient and practical solution to remote authentication: Smart card. Comput. Secur., 21: 372-375.

Diffie, W. and M. Hellman, 1976. New directions in cryptography. IEEE Tran. Inform. Theor., 22: 644-654.

Hsu, C.L., 2004. Security of Chien *et al.*'s remote user authentication scheme using smart cards. Comput. Stand. Interfac., 26: 167-169.

Hwang, M.S. and L.H. Li, 2000. A new remote user authentication scheme using smart cards. IEEE Trans. Consum. Electron., 46: 28-30.

Hwang, T., Y. Chen and C.S. Laih, 1990. Non-interactive password authentication without password tables. IEEE Region 10 Conference on Computer and Communication Systems (TENCON'90), September 1990, Hong Kong, pp: 429-431.

Juang, W., 2006. Efficient user authentication and key agreement in ubiquitous computing. Proceeding of the 2006 International Conference on Computational Science and its Applications (ICCSA'06), LNCS., 3983, May 8-11, Springer-Verlag Press, German, pp: 396-405.

Juang, W.S., 2004. Efficient password authenticated key agreement using smart cards. Comput. Secur., 23: 167-173.

Ku, W. and S. Chen, 2004. Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards. IEEE Trans. Consum. Electron., 50: 204-207.

Ku, W.C., H.M. Chuang and M.J. Tsaur, 2005. Vulnerabilities of wu-chieu's improved password authentication scheme using smart cards. IEICE Trans. Fundamentals, E88-A: 3241-3243.

Menezes, A.J., P.C. Oorschot and S.A. Vanstone, 1997. Handbook of Applied Cryptograph. 1st Edn., CRC Press, New York, ISBN: 0-8493-8523-7.

Shieh, W.G. and J.M. Wang, 2006. Efficient remote mutual authentication and key agreement. Comput. Secur., 25: 72-77.

Sun, H.M., 2000. An efficient remote use authentication scheme using smart cards. IEEE Trans. Consum. Electron., 46: 958-961.

Sun, H.M. and H.T. Yeh, 2006. Password-based authentication and key distribution protocols with perfect forward secrecy. J. Comput. Sys. Sci., 72: 1002-1011.

Tan, K. and H. Zhu, 1999. Remote password authentication scheme based on cross-product. Comput. Commun., 18: 390-393.

Wang, X.M., W.F. Zhang, J.S. Zhang and M.K. Khan, 2007. Cryptanalysis and improvement on two efficient remote user authentication scheme using smart cards. Comput Stand Interfac., 29: 507-512.

Yang, W.H. and S.P. Shieh, 1999. Password authentication schemes with smart card. Comput. Secur., 18: 727-733.

Yoon, E.J. and K.Y. Yoo, 2007. Two security problems of efficient remote mutual authentication and key agreement. Proceedings of the Future Generation Communication and Networking (FGCN 2007), December 6-8, IEEE Computer Society, pp: 66-70.