A Review on Remote User Authentication Schemes Using Smart Cards

Keerti Srivastava, Amit K. Awasthi, and R.C. Mittal

[1] Department of Applied Mathematics, Gautam Buddh University, Greater Noida, 201310, UP, India
keert.cipet@gmail.com
[2] Department of Applied Mathematics, Gautam Buddh University, Greater Noida, 201310, UP, India
awasthi.amitk@gmail.com
[3] Indian Institute of Technology, Roorkee, UA, India
rcmmmfma@iitr.ernet.in

Abstract. Remote user authentication is a mechanism in which the remote server verifies the legitimacy of a user over an insecure communication channel. Password based authentication schemes have been widely deployed to verify the legitimacy of remote users as password authentication is one of the simplest and the most convenient authentication mechanism over insecure networks. In remote user authentication scheme, the user is assigned a smart card, which is being personalized by some parameters and provide the legal users to use the resources of the remote system. Until now, there have been ample of remote user authentication schemes published in the literature and each published schemes have its own merits and demerits. Recently, many schemes proposed are based on the one-way hash function. The computational complexity of their schemes is superior to the discrete logarithm-problem-based schemes. In our paper, we have defined all the security requirements and the goals. An ideal password authentication schemes should satisfy and achieve all of these. We have presented the results of our survey through five of the currently available secure one way hash function based remote user authentication schemes. We hope an ideal smart card (not storing (ID_i, PW_i), which meets all the security requirements and achieves all the goals can be developed.

1 Introduction

With large scale development of network technology, remote user authentication in e-commerce and m-commerce has become an indispensable part to access the precious resources. It provides the legal users to use the resources of the remote system. To access resources at remote systems, users should have proper access rights like in Remote Login Systems, Automated Teller Machines (ATM's), Personal Digital Assistants (PDA) and Database Management Systems, etc and to access these resources, each user should have an identity and

K. Singh, A.K. Awasthi, and R. Mishra (Eds.): QSHINE 2013, LNICST 115, pp. 729–749, 2013.

password (ID_i, PW_i). Traditionally, the (ID_i, PW_i) are maintained by the remote system and when a user wants to login to a remote server, he simply submits his (ID_i, PW_i) to the server. On receiving the login message, the remote server compares the submitted corresponding pair (ID_i, PW_i) with stored one in password table. If match found, user will be granted to access the server resources.

Due to their efficiency and one-way property, one way hash functions have been used and as the basis on which more and more cryptosystems including password authentication systems, are being deployed. Now two factor authentication is very common in practice. In most cases, user ID with password and smart card are component of two factor authentication. In an open network environment, the remote authentication scheme using smart card is a very practical solution to validate the legitimacy of a remote user. In 2003, Wu and Chieu presented a user friendly remote authentication scheme using smart card [22], but Wang et al [33] found that Wu-Chieu scheme is vulnerable to forged login attack. He also presented an improved remote authentication scheme using smart card which eliminate this vulnerability. In 2005, D.Z. Sun found Wang et al.'s scheme was not secure under the smart card loss assumption, and proposed a new improved user friendly remote authentication scheme using smart card [7]. In continuation, many remote user authentication schemes have been proposed. These all authentication schemes having a common features in Registration phase. In each case, server issues a smart card which is storing ID and by which an intruder can impersonate a legal user by stealing the user's ID from stolen smart card. To overcome this problem, various proposal of improved remote user authentication schemes have been proposed, which can withstand the stolen smart card attack, impersonation attack by not storing the ID (or any secret information) in smart card. So that, if an intruder gets the stolen smart card, he could not get any access to the server as a legal user. Keeping in this view, an ample of authentication schemes as Sun et al' scheme [11], Chein et al [12], Ku et al [30], Yoon et al [8], X.M.Wang et al's [31] have been proposed.

In this paper we have reviewed above mentioned authentication schemes and proposed a new set of security requirements and goals for remote user authentication scheme with smart card. Every security requirement and goal is clearly defined. The separation of security requirements set and goals set allows us to establish a systematic approach for proving security of password authentication scheme with smart card (not storing the secret information). We reviewed few remote user authentication scheme using smart card to capture the work done and also to recognize the security challenges in this area. We presented our results based on the proposed security requirements, goals and communication and computation costs.

The remainder of this paper is organized as follows: In Section 2, related research work in this field is presented. Security requirements and the goals are presented in Section 3. Review of remote user authentication schemes and their cryptanalysis is presented in Section 4. Performance comparison of the schemes is given in Section 5. Finally, we conclude the paper in Section 6.

2 Related Research Work

In 1981, Lamport [16] proposed a novel password authentication scheme using cryptography hash function. However, high hash overhead and the necessity for password resetting decrease its suitability for practical use. Since then, many improved password authentication schemes [19],[18],[1],[29] have been proposed. One of the common feature of these schemes is that the server has to securely store a verification table, which contains the verifiers of user's passwords. If the verification table is stolen by an adversary, the system may be partially or totally broken. To resist such a stolen verifier attack, Hwang et al[28] in 1990, proposed a non interactive password authentication scheme and its enhanced version, which additionally uses smart cards. In Hwang et al scheme, the server requires neither storing the verifiers of user's passwords nor keeping any secret of the user. In 2000, Hwang and Li [10] proposed a verifier-free password authentication scheme using smart cards based on ElGamal's public-key technique. However, Hwang-Li's scheme does't allow users to freely choose and change their passwords. Furthermore, Hwang-Li's scheme had found to be vulnerable to impersonation attack. To improve their efficiency, Sun [11] proposed a light-weight verifier-free password authentication scheme using smart cards based on cryptographic hash functions. The major drawbacks of Sun's scheme was that the password is not easily memorizable and the user can't freely choose/change his password. Later, in 2002 H.Y.Chien pointed out that Sun's scheme [11] achieves unilateral user authentication and he also proposed a protocol to achieve mutual authentication[12]. In addition, the user can freely choose his password and the smart card not containing user's ID_i can avoid the risk of ID_i-theft or impersonation. Unfortunately, Chien'et al scheme can't withstand a parallel session attack [30],[6],[27],[26].

Further Ku's et al [30] pointed out the Chien et al [12] scheme is vulnerable to reflection attack, insider attack, guessing attack and is not repairable. However, Yoon et al [8] showed that Ku et al [30] scheme was susceptible to parallel session attack and was insecure for changing the user's password, and also proposed an enhancement to Ku et al's scheme to overcome such problems. Due to the power constraints of smart cards, the cost of implementation should be low as the lower the cost, the great chance of success in practical realization. Among those smart card based schemes, Ku et al's and Yoon et al's schemes require only several hash operations instead of the costly modular exponentiations. Therefore, their schemes exhibits great application potentiality in smart card field, regardless of their security.

In continuation process 2005, Yoon et al [9] pointed out Lee et al scheme [27],[26] is also vulnerable to some insidious attacks, reflection attack, stolen verifier attack, parallel session attack, replay attack, etc. To remedy these pitfalls, In 2007, X-M Wang pointed out that Ku et al's and Yoon et al's scheme are still vulnerable to the guessing attack, forgery attack and denial of service attack. As a result, only requiring few additional hash operations, X-M-Wang scheme can withstand the previously proposed attacks. In addition, wrong passwords input by the users can be deducted immediately and session key is also

provided after authentication phase. The computational cost and efficiency of the improved scheme are encouraging for the practical implementation in the resource-constraints environment.

3 Security Requirements

In this section, we define and list out the security attacks that are required for an ideal password authentication scheme to withstand.

3.1 SR1.Denial of Service Attack

This attack rejects all or specific users by means of an offensive action on the server or by means of a falsification of user's password-verifier. In this attack, an attacker can inconvenience the user but cannot imitate the user.

3.2 SR2.Forgery Attack(Impersonation Attack)

An attacker attempts to modify intercepted communication and masquerade as the legal user so that he can access the resources of a remote system. To manipulate the sensitive data of the legal users, an attacker can also masquerade as the legal server.

3.3 SR3.Parallel Session Attack

Without knowing the user's password, an attacker by masquerade as the legal user, can create a valid login message out of some eavesdropped communication between the user and the server. The attacker may launch a parallel attack by replaying the server's response message as the user's login message at a later time.

3.4 SR4.Password Guessing Attack

Most passwords have such low entropy that it is vulnerable to password guessing attack, where an attacker intercepts authentication messages and store it locally and then attempts to use a guessed password to verify the correctness of his guess using these authentication messages.

3.5 SR5.Replay Attack

Having intercepted previous communications, an attacker can impersonate as the legal user and login to the system. The attacker can replay the intercepted messages. An attack in which a valid data transmission is maliciously or fraudulently repeated either by the originator or by an adversary who intercepts the data and retransmits it possibly as part of a masquerade attack.

3.6 SR6.Smart Card Loss Attack

When the smart card is lost or stolen then an unauthorized users can easily change the password of the smart card or can guess the password of the user using password guessing attack or can impersonate the user to login to the system.

3.7 SR7.Stolen-Verifier Attack

In most of the application, the server stores hashed passwords instead of clear text passwords. In the stolen verifier attack, an adversary who steals the password verifier(e.g.hashed password) from the server can use it directly to masquerade as a legitimate user during the user authentication phase.

3.8 SR8.Reflection Attack

A reflection attack is a method of attacking a challenge-response authentication system that uses the same protocol in both directions. That is the same challenge-response protocol is used by each side to authenticate the other side. The essential idea of the attack is to trick the target in to providing the answer to its own challenge.

3.9 SR9.Insider Attack

An insider attack is intentional misuse by individuals who are authorized to use the servers and the networks. Insider of the server can perform an off-line guessing attack to obtain password. If succeeds, the insider of the server can try to use password to impersonate users to login other servers employing normal password authentication methods.

4 Goals

An ideal password authentication scheme should withstand all of the above attacks. Besides, it should achieve the following goals:

4.1 G1.No Verification Table

The remote system should not have a dictionary of verification tables such as clear text passwords or hashed passwords to authenticate users.

4.2 G2.Freely Chosen Password by the Users

If the password is chosen by the remote server without the consent of the user, then the user has no choice to choose his own password, which is not a case in the real-life applications, e.g. email subscription and online banking, etc. Secondly,

password chosen by the server could be long and random (for example, 1024 or 2048 bits), which might be difficult for a registered user to remember easily and it is most likely that user may forged this long and random password, if he is not frequently using the system. So,users should be able to choose their password freely.

4.3 G3.No Password Reveal

If the user's password is revealed to the server during registration, then it is likely that user uses the same password to login several servers for his convenience. In this case, the insider, e.g.,the administrator of the server can try to use the same password to impersonate user to login other servers that adopt normal remote user password authentication schemes. Therefore, the passwords should not be revealed by the administrator of the server.

4.4 G4.Password Dependent

The password independent scheme means that the scheme is equivalent to no password scheme, because user with any random password may access the server. Suppose an intruder theft the smart card for a short duration and makes a duplicate of it, now he has no need to crack the password because he may insert any random password, server will authenticate the intruder as a valid user. So, the authentication scheme should be password dependent.

4.5 G5.Mutual Authentication

Mutual Authentication should be provided between the user and remote system. Not only the server verify the legal users, but the users should be able to verify the legal server. Mutual Authentication can help withstand the server impersonation attack, where an attacker pretends to be the server to manipulate the sensitive data of the legal users.

4.6 G6.Session Key Agreement

A Session key should be established during the password authentication process. It is pertinent that after the successful authentication process, both parties will communicate some secret message, which should be encrypted to provide the confidentiality and secrecy of transmitted data.

4.7 G7.Forward Secrecy

Suppose the server's secret key is revealed and if the attacker tries to get passwords or other login information from the stolen smart card, he can easily impersonate the user and login to the system. Therefore, the scheme should be capable to provide forward secrecy even if the smart card is lost or stolen.

4.8 G8. User Anonymity

In some authentication scenarios, it is very important to preserve the privacy of a user because an adversary sniffing the communication channel can eavesdrops the communication parties involved in the authentication process and can easily analyze the transaction being performed by user.

4.9 G9.Smart Card Revocation

It is one of the requirements of smart card-based authentication schemes that in case of lost of cards, there should be provision in the system for invalidating the further use of lost smart card, otherwise an adversary can impersonate valid registered user.

4.10 G10.Efficiency for Wrong Password Login

Even if the user inputs wrong password by mistake in login phase, without any delay client should notify the user with error message, instead of sending the user's login request unconditionally to the server. If the client sends the information to the server for password verification, then the authentication will be delayed.

To be called an ideal scheme, a password authentication scheme should be able to withstand all of the above attacks and achieve all of the above goals. Unfortunately, none of the existing password authentication schemes can withstand all the above attacks and achieve all the goals. So, still there are opportunities to develop an ideal remote user password authentication scheme, which satisfies all security requirements and which meets all the goals.

5 Review of Five Remote User Authentication Schemes Based on Smart Cards

In this section, we review five smart card based password authentication schemes, which are based on hash function. Each password authentication scheme is composed of four phases. They are Registration phase, Login phase, Authentication phase and Password change phase. In the Registration phase, the user U registers with the remote server S and obtains a smart card through secure channel for future use. In the Login phase, When U wants to login to S for using resources of S, he inserts his smart card in to card reader and keys in his identity ID and password PW to access services. In the Authentication phase, S verifies the validity of the login request. Password change phase is invoked, whenever the user U wants to change his password. He can easily change his password without taking any assistance from the remote system. Now,we review some smart card based password authentication scheme.

The notations used throughout this paper are described as in the following.

U :	An User
(ID_u, PW_u) :	User $U's$ identifier and password respectively.
$CARD$:	$U's$ Smart Card.
S :	A Remote Server.
x :	Sserver's Secret Key.
T_U, T_S :	User's and Server's Current Time stamp respectively.
$h(.)$:	A Hash Function.
$\oplus$:	Bitwise XOR Operation.
$X \rightarrow Y\{M\}$:	X Send a message M to Y over an insecure channel.

5.1 Review of Sun's Scheme(Sun's Scheme[11])

A Registration Phase:
The user submits his ID_i to the remote system upon receiving the registration request, the remote system performs the following steps:

R1 Compute $PW_i = h(ID_i, x)$, where x be a secret key maintained by Remote system and h is a one-way function.

R2 Personalizes the smart card with the parameter $h(.)$.

R3 $S \Rightarrow U_i$:Smart card.

R4 $S \Rightarrow U_i$:PW_i through a secure channel.

B Login Phase:
The userU_i inserts his smart card to the card reader of a terminal, and keys his (ID_i, PW_i). Then Smart Card will perform the following operations:

L1 Compute $C_1 = h(T \oplus PW_i)$ where T is the current date and time of the input device.

L2 $U_i \Rightarrow S$:$C = (ID_i, C_1, T)$.

C Authentication Phase:
Upon receiving the login message $C = (ID_i, C_1, T)$ at time T' the remote system authenticates the userU_i with the following steps:

A1 Check the validity of ID_i.

A2 Verify the validity of the time interval between T and T'.if$(T' - T) \geq \Delta T$, then the remote system rejects the login request.

A3 Computes $PW_i = h(ID_i, x)$ and $C_1' = h(T \oplus PW_i)$.If $(C_1' = C_1)$ then the system accepts the login request. Otherwise, it rejects the login request.

Cryptanalysis of H.M.Sun'Scheme(Sun,2000)[11]

1. Mutual Authentication(breaks): Sun's scheme only achieves unilateral user authentication that, only authentication Server can authenticate the legitimacy of the remote user while the user cannot authenticate the legitimacy of Authentication Server. An attacker pretends to be the server to manipulate sensitive data of the legal users.
2. Replay Attack(Supports):Replay attack is not possible since uses time stamps. The idea behind the use of time stamps is to generate a synchronization mechanism between the client and the server. Replaying attacks(replaying an old (ID_i, C_1, T) in login phase)can not work because this will make A2 of Authentication phase fail.
3. Forward Secrecy(breaks):Suppose that an intruder has stolen the remote systems secret key x.It is obvious in this scheme that an intruder can easily compute each user's secret hash value as $PW_i = h(ID_i, x)$ and can impersonate any legitimate user. Therefore, in future, the server $S's$ secret key x may be changed to prevent an intruder's malicious activity. However, it would be much costs at a time and too expensive to re-compute all secret hash values at a time and communicate them to the users. Therefore, Sun et al's scheme does not guarantee a system's secret key forward secrecy.
4. Efficiency for wrong password login(breaks):If the user U_i inputs a wrong password in login phase by mistake,this wrong password will not be detected by the smart card at login phase. It is transferred unconditionally to the server. Server will check whether entered PW_i is wrong or right at step A3 of the Authentication phase. So the authentication will be delayed and inefficient.
5. Denial of Service Attack(breaks):Due to the unchangebility of $h(ID \oplus x)$ in Sun et al.'s scheme [11],a forged login request can not be prohibited even when U detected that his C_1 has been compromised. Accordingly, Ku et al. extended ID with $EID = (ID.n)$ and replace $C_1 = h(ID \oplus x)$ with $C_1 = h(EID \oplus x)$ in their improved scheme, so that C_1 can be changed by EID with different n when C_1 has been compromised. Unfortunately, the number n is stored in an entry table in server side, which is somewhat equivalent with using verification table, and suffers from the risk of modified entry table and the cost of protecting and maintaining the entry table. Once the intruder modifies n in entry table, the user's login message C_2 keeps $h(h(EID \oplus x) \oplus T_u)$ as before while the authentication message C_2 computed by system will change to $h(h(EID' \oplus x) \oplus T_u)$.
6. Impersonation attack(breaks):In Sun et al.'s[11] scheme, an adversary can obtain the corresponding password PW_i by performing a password guessing attack. The adversary intercepts the login request $C = (ID_i, C_1, T)$. First, he guesses a password PW_i^* and then computes $C_1^* = h(PW_i^* \oplus T)*$. If $C_1^* = C_1$, then the adversary has correctly guessed the password $(PW_i^* = PW_i)$. Once the adversary has correctly obtain PW_i, then he can impersonate the legal user.

7. Smart Card Lost Attack(supports):In Sun et al.'s[11] scheme only $h(.)$ is stored in smart card, which is one way function, the Smart Card Lost Attack is not possible in Sun et al.'s[11] scheme.
8. Password Guessing attack(breaks):In Sun et al.'s[11] scheme, an adversary can obtain the corresponding password PW_i by performing a password guessing attack. The adversary intercepts the login request $C = (ID_i, C_1, T)$. First, he guesses a password PW_i^* and then computes $C_1^* = h(PW_i^* \oplus T)^*$. If $(C_1^* = C_1)$, then the adversary has correctly guessed the password $(PW_i^* = PW_i)$. Once the adversary has correctly obtain PW_i, then he can impersonate the legal user.
9. Stolen Verifier attack(breaks):In the Registration phase,ID is passed to the server and PW is passed to the user. We assume that an adversary A can obtain the one way hash function $h(.)$ from stolen smart card. An adversary A can exhaustively examine all possible random number x until $PW_i = h(ID_i \oplus x)$. So, the scheme is vulnerable to stolen verifier attack.
10. Reflection Attack(supports):Reflection attack is possible only when the adversary gets both the messages(user to server and server to user). In Sun et al.'s[11],Reflection Attack is not possible since this scheme does not support mutual authentication. In this scheme, only the server authenticates the user, but the user does not authenticate the server. An adversary can get the message $C = (ID_i, C_1, T)$, which was transferred from the client to the server, but he cannot get the message, which was transferred from the server to the user. Reflection attack is possible only when the adversary gets both the messages.
11. Insider Attack(supports):The scheme is not vulnerable to insider attack. In the Registration Phase, PW_i is not in plain text form and calculated by $PW_i = h(ID_i, x)$ using secret key x which is known to server only, so any insider of S can not calculate the password $PW_i = h(ID_i, x)$.
12. Parallel Session Attack(supports):Parallel Session attack is possible only when the message structures between the user and the server are same.In Sun et al.'s[11],Parallel Session Attack is not possible since this scheme does not support mutual authentication. In this scheme, only the server authenticates the user, but the user does not authenticate the server. An adversary can get the message $C = (ID_i, C_1, T)$, which has transferred from the client to the server, but he cannot get the message which has transferred from the server to the user. .

5.2 Review of Chien et al.'s Remote User Authentication Scheme(Chien et al.'s scheme[12])

A.Registration Phase:
The user submits hisID_i and the password PW_i to the remote system through a secure channel. Upon receiving the registration request, the remote system performs the following steps:
R1.Computes a secret number $(R = h(ID_i \oplus x) \oplus PW_i)$ and checks an entry for the user U_i in his account database. Here x is a secret key of Remote Server and

$h(.)$ is a one-way function
R2.Personalizes the smart card with the parameters $h(.)$ and the secret number R
R3.$S \Rightarrow U_i$:Smart card.

B.Login Phase:
The userU_i inserts his smart card to the card reader of a terminal, and keys his ID_i, PW_i. Then Smart Card will perform the following operations:
L1.Compute $C_1 = (R \oplus PW_i)$ and $C_2 = h(C_1 \oplus T)$where T is the current date and time of the input device.
L2.$U_i \Rightarrow S$:$C = (ID_i, C_2, T)$.

C.Authentication Phase:
Upon receiving the login message $C = (ID_i, C_2, T)$ at time T' the remote system authenticates the user U_i with the following steps:
A1.Check the validity of ID_i.
A2.Verify the validity of the time interval between T and T'. If $(T' - T) \geq \Delta T$, then the remote system rejects the login request.
A3.Computes $C_2' = h(h(ID_i \oplus x) \oplus T)$. If $C_2' = C_2$, then the system accepts the login request. Otherwise, it rejects the login request.
A4.Computes $C_3 = h(h(ID_i \oplus x) \oplus T'')$ where T'' is the current time stamp.
A5.$S \Rightarrow U_i$:$D = (C_3, T'')$ for Mutual Authentication.
A6.On Receiving the message D from Remote Server, the user U_i computes $C_3' = h(C_1 \oplus T'')$.If $C_3' = C_3$ holds,the legitimacy of AS is verified.

Cryptanalysis of Chien et al.'s Remote User Authentication Scheme (Chien et al.'s scheme[12])

1. Parellel Session Attack(breaks):Consider the scenario of the parallel session attack [17] that an intruder U_a without knowing user's password wants to masquerade as a legal user U_i by creating a valid login message from the eavesdropped communication between AS and U_i when U_i wants to login the Authentication server AS, U_i sends the login message $C = (ID_i, C_2, T)$ to AS, if valid then the identification of U_i is authenticated and AS responses $D = (C_3, T'')$ to U_i. Once U_a intercepts this message, he masquerades as the legal user U_i to start a new session with AS by sending $C^* = (ID_i, C_2^*, T'')$ back to AS, where $(C_2^* = C_3)$. The login message $C^* = (ID_i, C_2^*, T'')$ will pass the user authentication of Chien et al.'s scheme[12] due to the fact that $C_2^* = C_3 = h(h(ID_i \oplus x) \oplus T'')$. Finally, AS responses the message (T''', C_3^*) to U_i, where$C_3^* = h(C_1' \oplus T''')$ and T''' is the current timestamp. The intruder intercepts and drops this message.
2. Reflection Attack(breaks):A malicious user intercepts the login request $C = (ID_i, C_2, T)$ and replaces the pair (C_3, T'') with (C_2, T) in verification phase. When the user U receives the pair (C_2, T), he verifies $C_2 = h(C_1 \oplus T)$, which holds truly. In this way, a malicious user reflects AS and U will be

fooled. Thus, Chien et al.'s scheme fails to provide mutual authentication and vulnerable to the reflection attack.

3. Impersonation attack(breaks):In Chein et al.'s scheme an adversary can obtain the corresponding password PW_i by performing a password guessing attack. The adversary intercepts the login request $C = (ID_i, C_2, T)$. First, he guesses a password PW_i^* and then computes $C_1^* = (R \oplus PW_i^*) = h(ID_i \oplus x)^*$ and $C_2^* = h(C_1^* \oplus T)$.if $(C_2^* = C_2)$, then the adversary has correctly guessed the password $(PW_i^* = PW_i)$ and $(C_1^* = C_1)$. Once the adversary has correctly obtain C_1, then he can impersonate the legal user.
4. Reparability of password(breaks):Since the password PW_i is the function of the identity ID_i of the user and the secret key x of AS. Therefore, to change the password PW_i for U_i, AS has to change ID_i or x. However, since x is commonly used for all users rather than specifically used for only U_i. It is not reasonable and efficient to change the secret key x for the security of a single user. Additionally, it is also impractical to change identity of the user. Thus, they claimed that the Chien et al.'s scheme[12] is non reparable.

5. Insider Attack(breaks):The password of the user U_i will be reveal to AS in the registration phase. If user U_i uses the same password to access other servers for convenience, the insider of AS can impersonate the user U_i to access other services.
6. Replay Attack(supports):Replay attack is not possible since this scheme uses time stamps. The idea behind the use of time stamps is to generate a synchronization mechanism between the client and the server. Neither the replay of an old login message $C = (ID_i, C_2, T)$ in the login phase nor the replay of the remote server's response message $D = (C_3, T'')$ in the verification phase.
7. No Password Reveal(supports):Since the user's password is not revealed to the server during registration, therefore impersonation attack by Authentication server is not possible in Chein et al.'s scheme.
8. Mutual Authentication(supports):The Chein et al.'s scheme can mutually authenticate each other between user and server by A1,A2,A3,A6 step in Authentication Phase.
9. Efficiency for wrong password login(breaks):If user U inputs a wrong password by mistake, this wrong password will not be detected by the client, instead it transfers login information unconditionally to the server. Even though server checks for valid login at step A2 of Authentication Phase.
10. Denial of Service Attack(breaks):Due to the unchangebility of $h(ID, x)$ in Chien et al.'s scheme [12], a forged login request can not be prohibited even when U detected that his C_1 has been compromised. Accordingly, Ku et al. extended ID with $EID = (ID.n)$ and replace $C_1 = h(ID \oplus x)$ with $C_1 = h(EID \oplus x)$ in their improved scheme, so that C_1 can be changed by EID with different n when C_1 has been compromised. Unfortunately, the number n is stored in an entry table in server side, which is somewhat equivalent with using verification table, and suffers from the risk of modified entry table and the cost of protecting and maintaining the entry table. Once the intruder modifies n in entry table, the user's login message C_2 keeps

$h(h(EID \oplus x) \oplus T_u)$ as before while the authentication message C_2 computed by system will change to $h(h(EID' \oplus x) \oplus T_u)$.

11. Password Guessing attack(supports):In Chien et al.'s[12] scheme, the password guessing attack is not possible because if an adversary intercepts the login request $C = (ID_i, C_2, T)$, he could not guess a correct password PW_i from C_2 because $C_2 = h(C_1 \oplus T)$, $C_1 = (R \oplus PW_i)$ and $R = h(ID_i \oplus x) \oplus PW_i$.
12. Smart Card Lost Attack(breaks):A user U_i may lose this smart card, which is found by an attacker. then he could extract the stored values through some technique such as by monitoring their power consumption and reverse engineering techniques as pointed out in Kocher et al[21]. He can extract the stored message $(R, h(.))$ from smart card then by intercepting login message $C = (ID_i, C_2, T)$ he can compute and $C_2^* = h(R \oplus PW_i^* \oplus T)$ by guessing password PW_i^* and if $(C_2^* = C_2)$ then the guessed password will be correct.
13. Stolen Verifier attack(breaks):In the Registration phase, (ID_i, PW_i) is passed to the server. We assume that an adversary A can obtain the secret value $(R, h(.))$ from stolen smart card. A can exhaustively examine all possible random number x until $R = h(ID_i \oplus x)$. So, the scheme is vulnerable to stolen verifier attack.

5.3 Review of Ku and Chen's Scheme[30]

This scheme has four phase:the Registration Phase,Login Phase,Verification Phase and the Password change phase.

A.Registration Phase:This phase is invoked whenever U initially registers to S. Let n denote the number of times U registers to S.
R1.U selects a random number b and computes $h(b \oplus PW_i)$
R2.$U \Rightarrow S$:$ID, h(b \oplus PW_i)$
R3.If it is $U's$ initial registration, S create an entry for U in the account database and stores $n = 0$ in this entry. Otherwise, S sets as $n = n + 1$ in the existing entry for U. Next, S performs the following computations:$R = h(EID \oplus x) \oplus h(b \oplus PW_i)$, where $EID = (ID.n)$.
R4.$S \Rightarrow U$:a smart card containing $(R, h(.), b)$.

B.Login Phase:The userU_i inserts his smart card to the card reader of a terminal, and keys his (ID_i, PW_i). Then Smart Card will perform the following operations:
L1.Compute $C_1 = R \oplus h(b \oplus PW_i)$,$C_2 = h(C_1 \oplus T)$ where T is the current date and time of the input device.
L2.$U_i \Rightarrow S$:$C = (ID_i, C_2, T)$.

C.Authentication Phase:
Upon receiving the login message $C = (ID_i, C_2, T)$ at time T' the remote system authenticates the user U_i with the following steps:
A1.Check the validity of ID_i.
A2.Verify the validity of the time interval between T and T'.if$(T' - T) \geq \Delta T$,

then the remote system rejects the login request.
A3.S computes $C_2' = h(h(EID \oplus x) \oplus T)$ If $(C_2' = C_2)$, then Server S accepts $U's$ login request. Otherwise, it rejects the login request.
A4.Computes $C_3 = h(h(EID \oplus x) \oplus T'')$ where T'' denotes $S's$ current time stamp.
A5.$S \Rightarrow U_i$:$D = (C_3, T'')$ for Mutual Authentication.
A6.On Receiving the message D from Remote Server, checks the validity of T_s
A7.The user U_i computes $C_3' = h(C_1 \oplus T'')$. If $(C_3' = C_3)$ holds, the legitimacy of AS is verified.

D.Password change Phase:
This phase is invoked whenever U wants to change his password PW with a new one, say PW_{new}.
P1.Uinserts his smart card in to card reader, enters ID and PW, and requests to change password. Next, U enters PW_{new}.
P2.Smart Card computes$R_{new} = R \oplus h(b \oplus PW_i) \oplus h(b \oplus PW_{new})$, which yields $h(EID \oplus x) \oplus h(b \oplus PW_{new})$ and then replaces R with R_{new}.

Cryptanalysis of Ku et al.'s Scheme[30]. 1.Smart card loss Attack(breaks): In Ku et al's scheme, $U's$ smart card contains $(R, b, h(.))$. Due to the fact that adversary could have extracted the secret information stored in the smart card by monitoring the power consumption[21] or by analysing the leaked information, the adversary can obtain $R = h(EID \oplus x) \oplus h(b \oplus PW)$ as well as b. Suppose that the adversary also has intercepted one of $U's$ past login messages, i.e,$C = (ID_i, C_2, T)$ he can perform a guessing attack to obtain PW_i by guessing a password PW_i^* and comparing $C_2' = h(R \oplus h(b \oplus PW_i^*) \oplus T_u)$ with the received C_2if $(C_2' = C_2)$, the adversary has correctly guessed $(PW_i^* = PW_i)$, otherwise, the adversary tries another candidate password. Since password PW_i are selected by users, they are usually short and simple for catchiness. Hence, PW_i could be obtained by off-line guessing attack.

2.Forgery Attack (breaks):Once the adversary obtain (PW, R, b) by guessing attack then he can compute $C_1 = R \oplus h(b \oplus PW_i)$ and then impersonates U by forging $U's$ login message $(ID, h(C_1 \oplus T_u^{'}), T_u^{'})$ at time $T_u^{'}$.

3.Denial of service attack(breaks):Due to the unchangebility of $h(ID \oplus x)$ in Chein et al.'s scheme [12],a forged login request can not be prohibited even when U detected that his C_1 has been compromised. Accordingly, Ku et al. extended ID with $EID = (ID.n)$ and replace $C_1 = h(ID \oplus x)$ with $C_1 = h(EID \oplus x)$ in their improved scheme, so that C_1 can be changed by EID with different n when C_1 has been compromised. Unfortunately, the number n is stored in an entry table in server side, which is somewhat equivalent with using verification table, and suffers from the risk of modified entry table and the cost of protecting and maintaining the entry table. Once the intruder modifies n in entry table, the user's login message C_2 keeps $h(h(EID \oplus x) \oplus T_u)$ as before while the authentication message C_2 computed by system will change to $h(h(EID' \oplus x) \oplus T_u)$.

4.Forward Secerecy Attack(supports):Suppose that an intruder has stolen the remote systems secret keys x.However in this scheme that an intruder can not compute each user's secret hash value$h(EID_u \oplus x)$, as $EID_u = h(ID_u \oplus n)$ and S sets as $(n = n + 1)$. Therefore, Hsiang et al.'s Scheme is not vulnerable to Forward secerecy.

5.Inefficiency for error password login(breaks):Even if Uinputs an error password in login phase, the smart card still sends $U's$ login request unconditionally to server. This error is not detected until the server checks $C_2? = h(h(EID \oplus x) \oplus T_u)$ at authentication phase. Therefore, the password authentication is delayed and inefficient.

6.Parellel Session Attack(breaks):Parellel Session Attack is possible since the message structures between the user and the server are same.

7.Replay Attack(supports):Replay attack is not possible since this scheme uses time stamps. The idea behind the use of time stamps is to generate a synchronization mechanism between the client and the server. Neither the replay of an old login message $C = (ID_i, C_2, T)$ in the login phase nor the replay of the remote server's response message $D = (C_3, T'')$ in the verification phase.

8.Stolen verifier attack(supports):In Registration phase,ID is passed in plain text form to the server, We assume that an adversary A intercepts and gets ID and $h(b \oplus PW_i)$. He can obtain R from lost smart card. Adversary cannot compute $h(EID \oplus x)$ as $h(EID \oplus x) = h(b \oplus PW_i) \oplus R$, $EID = (ID.n)$ and S set as $n = n + 1$.

9.Reflection attack(breaks):A malicious user intercepts the login request ID_i, C_2, T_u and replaces the pair (C_3, T_s) with (C_2, T_u) in verification phase. When the user U receives the pair (C_2, T_u), he verifies $C_2 = h(C_1 \oplus T)$, which holds truly. In this way, a malicious user reflects AS and U will be fooled. Thus, Chien et al.'s scheme fails to provide mutual authentication and vulnerable to the reflection attack.

10.Insider Attack(supports):The scheme is not vulnerable to insider attack because In the Registration Phase, PW_i is not in plain text form and calculated by $h(b \oplus PW_i)$ using secret key b which is known to user only and need not to remember after this step. So any insider of S can not calculate the password $h(b \oplus PW_i)$.

11.Password guessing attack(supports):The scheme is not vulnerable to insider attack because an adversary can not guess the password from the login message (ID_i, C_2, T_u).

5.4 Review of X-M.Wang et al.' Scheme [31]

A.Registration Phase:
The user selects a random number b and computes $h(b \oplus PW)$. An User submit his/her $ID_i, h(b \oplus PW)$ to the remote system. Upon receiving the registration request, the remote system performs the following steps:
R1.Compute $p = h(ID_i, x), R = p \oplus h(b \oplus PW), V = h_p(h(b \oplus PW))$, where x be a secret key maintained by Remote system and h is a one-way function.
R2.Personalizes the smart card with the parameters $R, V, h(.), h_p(.)$.

R3.$S \Rightarrow U_i$:Smart card.
R3.U enters b into his smart card so that he does not need to remember b anyone.

B.Login Phase:
The userU_i inserts his smart card to the card reader of a terminal and keys his (ID_i, PW_i). Then Smart Card will perform the following operations:
L1.Compute $p = R \oplus (b \oplus PW)$ and checks whether $h_p(h(b \oplus PW)) = V$, if not hold then reject the login request.
L2.Smart card generates a random number r, and performs the following computations:
$C_1 = p \oplus h(r \oplus b)$,$C_2 = h_p(h(r \oplus b) \oplus T_u)$
L3.$U_i \Rightarrow S$:$M = (ID_i, C_1, C_2, T_u)$ where T_u denotes $U's$ current time stamp.

C.Authentication Phase:
Upon receiving the login message $M = (ID_i, C_1, C_2, T_u)$ at time T_s the remote system authenticates the user U_i with the following steps:
A1.Check the validity of ID_i.
A2.Verify the validity of the time interval between T_u and T_s. If$(T_s - T_u) \geq \Delta T$, then the remote system rejects the login request.
A3.S Computes $p = h(ID_i \oplus x)$ and $C_1' = p \oplus C_1$.If $C_1' = C_1$ then check whether equation $h_p(C_1' \oplus T_u) = C_2$ holds or not. If holds, it means user is authentic and S accepts the login request, and performs step 4. Otherwise, S rejects login request.
A4.For the mutual authentication, S computes $C_3 = h_p(C_1' \oplus T_s)$ and then sends mutual authentication message C_3, T_s to user U.
A5.Upon receiving the message (C_3, T_s),U verifies either T_s is invalid or $(T_s = T_u)$, U terminates this session, otherwise performs step6.
A6.U computes $C_3' = h_p(h(r \oplus b) \oplus T_s)$ and computes $C_3' = C_3$ holds then user believes that the remote party is authentic system and the mutual authentication between U and S is completed, otherwise U terminates the operation. In addition, since r is randomly generated in each login phase, $C_1' = h(r \oplus b)$ shared between $U$and S can be used as the session key for the subsequent private communication.

D.Password Change Phase:U inserts his smart card in to card reader, enters ID, PW and requests to change password, then the smart card performs the following steps without any help of server.
P1.Compute $p^* = R \oplus h(b \oplus PW)$ and $V^* = h_p^*(h(b \oplus PW))$
P2.Check whether V^* equals to the stored V or not. If not, rejects the password change request. Otherwise U chooses a new password PW_{new}.
P3.Compute $R_{new} = p^* \oplus h(b \oplus PW_{new})$ and $V_{new} = h_p^*(h(b \oplus PW_{new}))$, and then stores R_{new}, V_{new} in to the user's smart card and replaces the old values R, V respectively. Now, new password is successfully updated and this phase is terminated.

Cryptanalysis of X-M-Wang et al.'s Remote User Authentication Scheme[31]:

1. Guessing attack resistance(supports):Firstly,R is stored in smart card with $R = h(ID \oplus x) \oplus h(b \oplus PW)$ since x and PW are unknown to adversary, one can get neither $h(ID \oplus x)$ nor $h(b \oplus PW)$ even if R, b are extracted from the smart card. Similarly, even if the stored information $V = h_p(h(b \oplus PW))$ is revealed, both $p = h(ID_i \oplus x), h(b \oplus PW)$ are still secure. Next, suppose the login message $M = (ID_i, C_1, C_2, T_u)$ sent by U be eavesdropped in a common channel. However, even under the advanced hash collision attack proposed by Wang et al's [32], the secret information$h(ID \oplus x)$ is still secure due to the fact that $C_1 = p \oplus h(r \oplus b)$,$C_2 = h_p(h(r \oplus b) \oplus T_u)$ are combined with $h(r \oplus b)$, which is randomized in each login request and one has no way to get it. Moreover,C_1, C_2, R, V are all combined with two random items.
2. Forgery/Impersonation attack resistance(supports):Impersonation Attack is not possible in this scheme, if an adversary attempts to modify $U's$ login message $M = (ID_i, C_1, C_2, T_u)$ in to $M = (ID_i, C_1^*, C_2^*, T_u*)$. However, this impersonation attempt will fail in step A3 of the Authentication phase, because there is no way to obtain the values of $h(ID_i \oplus x), h(r \oplus b)$ to compute the valid value of C_2.
3. Replay Attack Resistance(supports): Neither the replay of an old login message (ID_i, C_1, T_u) nor replay of the remote system's response C_3, T_s will work. It would have failed in step A2 and A5 of authentication phase, because of the time interval validation respectively.
4. Denial Of Service attack resistance(supports):In this scheme secret hash value i.e.$h(ID \oplus x)$ is not stored directly into smart card but is combined with the other hash values, such as $h(b \oplus PW)$ in R or $h(r \oplus b)$ in C_1, or act as the secret key of keyed hash function in V and C_2. Clearly, $h(ID \oplus x)$ can't be derived from any revealed value (R, V, C_1, C_2) or their combined values. So, the assumption of $h(ID \oplus x)$ being revealed is impractical or impossible for this scheme. That is, no entry table is necessary any more in this scheme.
5. Server spoofing attack resistance(supports) :The spoofing attack is completely solved by providing mutual authentication between user and remote system. Remote system S sends mutual authentication message C_3, T_s to the user. If an attacker intercepts it and re-sends the forge message i.e.(C_3^*, T_s^*) to user U, it will be verified in steps A5 and A6 of the authentication phase because the value of $C_3'$is computed by $C_3' = h_p(h(r \oplus b) \oplus T_s)$. In addition, replay of this message can be exposed because of the time stamp.
6. High efficiency in password authentication(supports):In login phase, If U inputs an error password PW', the smart card computes $p' = R \oplus h(b \oplus PW')$ and checks equation $h_p'(h(b \oplus PW'))? = V$ in 2 step. Obviously, the result is negative when $(PW \neq PW')$ and smart card terminates the login session. Thus, the validity of input password can be immediately detected by smart card yet need not wait for server authentication.
7. Forward secrecy(breaks):Suppose that an intruder has stolen the remote systems secret keys x. It is obvious in this scheme that an intruder can

easily compute each user's secret hash value as Compute $p = h(ID_i, x), R = p \oplus h(b \oplus PW), V = h_p(h(b \oplus PW))$, where x be a secret key maintained by Remote system and h is a one-way function and can impersonate any legitimate user. Therefore, in the future, the server's secret key x may be changed to prevent an intruder's malicious activity. However,it would be much too expensive to re-compute all secret hash values at a time and communicate them to the users. Therefore, this scheme does not guarantee a system's secret key forward secrecy.

8. Parallel Session Attack(supports):Without knowing a user's password, an attacker cannot masquerade as the legal user due to the typical structure of $V = h_p(h(b \oplus PW))$and (R, b).
9. Smart card lost attack(supports):Firstly,R is stored in smart card with $R = h(ID \oplus x) \oplus h(b \oplus PW)$. Since x and PW are unknown to adversary, one can get neither $h(ID \oplus x)$ nor $h(b \oplus PW)$ even if R, b are extracted from the smart card. Similarly, even if the stored information $V = h_p(h(b \oplus PW))$ is revealed, both $p = h(ID_i \oplus x), h(b \oplus PW)$ are still secure. Next, suppose the login message$M = (ID_i, C_1, C_2, T_u)$ sent by U be eavesdropped in a common channel. However, even under the advanced hash collision attack proposed by Wang et al's [32],the secret information $h(ID \oplus x)$ is still secure due to the fact that $C_1 = p \oplus h(r \oplus b)$,$C_2 = h_p(h(r \oplus b) \oplus T_u)$ are combined with $h(r \oplus b)$, which is randomized in each login request and one has no way to get it. Moreover, (C_1, C_2, R, V) are all combined with two random items.
10. Stolen verifier attack(supports):In this scheme all the secret values are stored in hashed way. So an adversary can not steal the secret information during the transaction of information.
11. Reflection Attack(supports):Reflection attack is not possible since message structures between the user and the server are different. Here the user computes $C_1 = p \oplus h(r \oplus b)$,$C_2 = h_p(h(r \oplus b) \oplus T_u)$ by step L3,L4 and sends (C_1, C_2) to the server. The server intern sends mutual authentication message (C_3, T_s) by step A3,A4 of Authentication phase. Both the messages have different structures. So the adversary will not be able to perform this attack.
12. Insider Attack(supports):The scheme is not vulnerable to insider attack because In the Registration Phase, PW_i is not in plain text form and calculated by $h(b \oplus PW_i)$ using secret key b which is known to user only and need not to remember after this step. So any insider of S can not calculate the password $h(b \oplus PW_i)$.

6 Performance Comparison

In This section, we compare the schemes in terms of security requirements.

By checking out all the security requirements that are listed in the section 3, we can judge if a scheme deserves the title of an ideal password authentication scheme. Comparison of security requirements:

Security Requirements Schemes	SR1	SR2	SR3	SR4	SR5	SR6	SR7	SR8	SR9
Sun et al.[11]	N*	N*	Y*	N*	Y	Y*	N*	Y*	Y*
Chien et al.[12]	N*	N*	N[2]	Y*	Y*	N*	N*	N*	N*
Ku et al.[30]	N[31]	N[31]	N*	Y*	Y*	N[31]	Y*	N*	Y*
Wang et al.[31]	Y[31]	Y[31]	Y*	Y[31]	Y[31]	Y*	Y*	Y*	Y*

6.1 Security Requirements Analysis

In table 1,a comparison of security requirements is shown in which the following notations are used.
SRi:Proposed security requirements are in Section 3.
Y:Meets the security requirement,cryptanalysis done by the corresponding authors.
N[n]:Not meets the security requirements,cryptanalysis done by the [n] authors.
Y*:Meets the security requirement,cryptanalysis done by us.
N*:Not meets the security requirement,cryptanalysis done by us.

7 Conclusion

In this paper, the survey of the five smart card based authentication schemes over insecure networks has been done. We have defined the security requirements and An ideal password authentication scheme should satisfy and achieve it. Survey results are based on the cryptanalysis done by other researchers and also done by us. We have done the security and functionality comparison of schemes based on the 9 security requirements. Except one of them, Wang et al.'s[31], all the schemes do not meet all the security requirements , Wang et al.'s[31] scheme satisfies all the security requirements but do not achieve all the goals, So we can not say that this is an ideal password authentication scheme. Therefore, there is a need to look into these goals in future research work. Unfortunately, none of the schemes can satisfy all the security requirements and all the goals. We hope our work will provide a better understanding of the security challenges of smart card based remote user authentication and pave the way for further research in this area.

References

1. Shimizu, A.: A dynamic password authentication method by one way function. IEICE Transactions 173-D-1(7), 630–636 (1990)
2. Hsu, C.-L.: Security of chien et al.'s remote user authentication scheme using smart cards. Computer Standards and Interfaces 26, 167–169 (2004)
3. Hsu, C.L.: A user friendly Remote User Authentication scheme with smart cards against impersonation attacks. Applied Mathematical and Computer 170, 135–143 (2005)
4. Chan, C.K., Chang, L.M.: Cryptanalysis of a Remote user authentication scheme using smart card using smart cards. IEEE Trans, Consumer Electron 46, 992–993 (2000)

5. Li, C.T., Hwang, M.S.: An efficient biometric based remote user authentication scheme using smart cards. Journal of Network and Computer Applications 33(1), (5) (January 2010)
6. Hsu, C.L.: Security of Chien et al's remote user authentication scheme using smart cards. Computer Standard and Interfaces 26(3), 167–169 (2004)
7. Sun, D.-Z., et al.: Weakness and improvement on wang-Li-Tie's user friendly remote authentication scheme. Applied Mathematics and Computation 170, 1185–1193 (2005)
8. Yoon, E.J., Ryu, E.K., Yoo, K.Y.: Further improvement of an efficient password based remote user authentication scheme using smart cards. IEEE Trans on Consumer Electronics 50(2), 612–614 (2004)
9. Yoon, E., Yoo, K.: More efficient and secure remote user authentication scheme using smart card. In: Proceeding of 11th International Conference on Parallel and Distributed System, vol. 2, pp. 73–77 (2005)
10. Hwang, Hwang, L.M.S., Li, L.H.: A new remote user authentication scheme using smart card. IEEE Transactions on Consumer Electronics 46(1), 28–30 (2000)
11. Sun, H.M.: An efficient remote user authentication scheme using smart card. IEEE Trans on Consumer Electronic 46(4) (2000)
12. Chien, H.Y., Jan, J.K., Tseng, Y.M.: An efficient and practical solution to remote authentication: smart cards. Computer and Security 21(4), 372–375 (2002)
13. I-En-Liao, C.-C., Lee, N.-S., Hwang, N.-S.: A password authentication scheme over insecure networks. Journals of Computer and System Sciences 72, 727–740 (2006)
14. Shen, J.J., Lin, C.W., Hwang, M.S.: A modified remote user authentication scheme using smart cards. IEEE Trans, Consumer Electron 49(2), 414–416 (2003)
15. Xu, J., Zhu, W.-T., Feng, D.-G.: An improvement smart card based Password Authentication scheme with provable security. Computer Standard and Interfaces 31, 723–728 (2009)
16. Lampot: Password authentication with insecure communication. ACM 24(11), 770–772 (1981)
17. Gong, L.: A security risk of depending on synchronized clocks. Operating Systems Review 26(1), 49–53 (1992)
18. Sandirigama, M., Shimiz, A., Noda, M.T.: Simple and secure password authentication protocol. (SAS), IEICE Transactions on Communication E83-B(6), 1363–1365 (2000)
19. Haller, N.H.: The S/KEY(TM) one time password system. proc. In: Proc. Internet Society Symposium on Network and Distributed System Security, pp. 151–158 (1994)
20. Lee, N.Y., Chin, Y.C.: Improved RAS with smart cards. Computer Standards and Interface 27(2), 177–180 (2005)
21. Kocher, P.C., Jaffe, J., Jun, B.: Differential power analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, p. 388. Springer, Heidelberg (1999)
22. Wu, S.T., Chieu, B.C.: A user friendly remote user authentication scheme with smart cards. Computers and Security 22(6), 547–550 (2003)
23. Lee, S.W., Kim, H.S., Yoo, K.Y.: Improvement of chien etal's remote user authentication scheme using smart card. Computer Standards and Interface 27(2), 181–183 (2005)
24. Kim, S.K., Chung, M.G.: More secure remote user authentication scheme using smart cards. Journal of Computer and Communications, doi:10.10161-1 coman 2008.11.026
25. Kim, S.K., Chung, M.G.: More secure remote user authentication scheme. Computer Communication (2009)

26. Lee, S., Kim, H., Yoo, K.: Improvement of chen et's remote user authentication scheme using smart cards. Computer Standards and Interface 27, 181–183 (2004)
27. Lee, S., Kim, H., Yoo, K.: Improved efficient remote user authentication scheme using smart card. IEEE Trans on Communication Electronics 50(2), 565–567 (2004)
28. Hwang, T., Chen, Y., Laih, C.S.: Non interactive password authentication without password tables. In: Proc. IEEE Region 10 Conference on Computer and Communication Systems, Hong Kong, pp. 429–431 (September 1990)
29. Chen, T.H., Lee, W.B.: A new method for using hash functions to solve remote user authentication. Computers and Electicals Engineering 34, 53–62 (2008)
30. Ku, W.C., Chen, S.N.: weakness and improvement of an efficient password based remote user authentication scheme using smart cards. IEEE Trans on Consumer Electronics 50(1), 204–207 (2004)
31. Wang, X.M., Zhang, W.F., Zhang, J.S., Khan, M.K.: Cryptanalysis and improvement on two efficient remote user authentications scheme using smart cards. Computer Standards and Interfaces 29(5), 507–512 (2007)
32. Wang, X., Yin, Y.L., Yu, H.: Finding Collisons in the full SHA1 (February 2005), http://www.infosec.sdu.edu.en/paper/sha1
33. Wang, Y.J., Li, J.H., Tie, L.: Security analysis and improvement of a user friendly remote authentication protocol. Applied Mathematics and Computer (in press)