Remote User Authentication Scheme in Multi-server Environment using Smart Card

Jitendra Kumar Tyagi
I.E.T., Alwar, India

A.K. Srivastava
I.E.T., Alwar, India

Pratap Singh Patwal
I.E.T., Alwar, India

ABSTRACT
In a single server environment, one server is responsible for providing services to all the authorized remote users. However, the problem exists if a user wishes to access several network services. To overcome this weakness, various multi-server authentication schemes have been proposed. Though, these schemes are exposed to one or the other network security attack. This paper suggests robust multi-server authentication scheme using smart cards. Its security is based on cryptographic one-way hash function and the discrete logarithm problem. This scheme allows remote users to access multiple servers without separately registering with each server. Furthermore, it eliminates the use of verification table, allows users to choose and change the password securely without taking any assistance from the server or registration center, provides mutual authentication and establishes a common session key between user and the server. Additionally, the proposed scheme withstands user impersonation attack, server impersonation attack, replay attack, reflection and parallel session attacks, password guessing attack, insider attack, smart card loss attack and stolen verifier attack.

Keywords
Authentication, Multi-server, Nonce, Session key, Smart card

1. INTRODUCTION
In order to authenticate the remote users, password based schemes have been widely used. In these schemes, server keeps a verification table secretly to verify the legitimacy of a remote user. Based on one way hash function, Lamport [1] proposed a password authentication scheme to authenticate remote users. Though, this scheme has a security pitfall as an intruder can steal or modify the verification table and masquerade as a legitimate user. To solve the stolen problem related to verification table, various elegant single server smart card authentication schemes have been proposed. Wu [2] suggested a remote login authentication scheme based on a geometric approach and claimed that the scheme eliminates use of verification table, provides security against replay attack and impersonation attack. However, Hwang [3] proved that Wu's scheme has security flaws as an unauthorized user can easily forge a valid login request. Using RSA cryptosystem, Yang and Shieh [4] proposed ID-based scheme. Nevertheless, Chan and Cheng [5] showed that Yang-Shieh's scheme is weak against impersonation attack. Hwang and Li [6] presented a remote user authentication scheme based on ElGamal's cryptosystem and claimed that their scheme is free from maintaining verification table and resists replay attack. Though, Chan and Cheng [7] pointed out that Hwang-Li's scheme is susceptible to impersonation attack.

To improve efficiency, Sun [8] proposed a remote user authentication scheme using one way hash function. However, Hsu [9] found that Sun's scheme fails to resist offline and online password guessing attacks. Chien et al. [10] also suggested improvement over Sun's scheme and claimed that their scheme is free from maintaining verification table, allows users to choose the password freely and provides mutual authentication between remote user and the server. Nevertheless, Hsu [9] proved that Chien et al.'s scheme is vulnerable to parallel session attack. These entire schemes use timestamp to avoid replay attack and require synchronized environment which is very difficult to achieve in real world applications. For some applications, user and the server need a session key in order to keep the information secret for the successive communications after they authenticate each other. Considering these issues, Juang [11] proposed a nonce based scheme with an added feature of session key agreement. Though, it is analyzed that this scheme is weak against insider attack and user is not allowed to change the password freely. Das et al. [12] offered a dynamic ID-based scheme using one way hash function. They claimed that their scheme permits users to choose and change the passwords freely and provides security against replay attack, forgery attack, guessing attack, insider attack and stolen verifier attack. Nevertheless, Liao et al. [13] showed that Das et al.'s scheme is exposed to guessing attack, insider attack and does not provide mutual authentication. Based on symmetric key cryptography, Song [14] proposed an authentication scheme and claimed that the scheme provides mutual authentication and shared session key and it is secure against impersonation attack, parallel session attack, replay attack and modification attack. However, Pippal et al. [15] proved that Song's scheme fails to provide perfect forward secrecy and insecure against Denial-of-Service attack.

The remainder of this paper is organized as follows. Section 2 describes the work related to multi-server authentication scheme. The proposed scheme is discussed in section 3. Section 4 analyzes the security of proposed scheme and section 5 concludes the paper.

2. RELATED WORK
Conventional single server authentication schemes suffer from a significant inadequacy. In order to use numerous network services, the user must register to these servers separately which is a tedious job. For that, the user has to register its identity and password in different servers and also maintain several user IDs and PWs. To handle this problem, multi-server authentication scheme has been proposed. Li et al. [16] proposed first authentication scheme for multi-server architecture using neural networks. They claimed that their scheme allows users to choose the passwords freely. Lin et al. [17] found that Li et al.'s scheme takes long time on training neural networks and they suggested an improved scheme based on ElGamal digital signature and geometric properties on the Euclidean plane. However, Cao and Zhong [18]

proved that Lin et al.'s scheme is exposed to impersonation attack and is unrealistic as every user must have a large amount of memory to store the public parameters for authentication. Using one-way hash function and symmetric cryptosystem, Juang [19] proposed a nonce based scheme which covers all the previous advantages with an additional trait of generating a session key. Nevertheless, Ku et al. [20] proved that Juang's scheme is exposed to insider attack and fails to provide forward secrecy. Chang and Lee [21] suggested improvement over Juang's scheme and claimed that their scheme is able to resist stolen-verified attack, server spoofing attack, smart card loss attack, replay attack and provides mutual authentication and provides forward secrecy.

Liao and Wang [22] presented a dynamic ID-based remote user authentication scheme using one way hash function [22]. However, Chen et al. [23] pointed out that Liao-Wang's scheme does not provide forward secrecy. Hsiang and Shih [24] found that Liao-Wang's scheme fails to resist insider attack, impersonation attack, server spoofing attack and shows inadequacy in providing mutual authentication. To overcome these weaknesses, they also proposed an improved scheme. Though, Sood et al. [25] showed that Hsiang-Shih's improved scheme fails to provide security against replay attack, impersonation attack, stolen smart card attack and has incorrect password change phase. Tsai [26] offered a nonce based scheme using one way hash function. However, Zhu [27] proved that Tsai's scheme is vulnerable to denning-sacco attack, server spoofing attack and does not provide perfect forward secrecy. To overcome these drawbacks, they proposed an improved scheme also.

Based on these motivations, this paper presents secure multi-server authentication scheme using smart cards. Breaking this scheme is as complex as solving the discrete logarithm problem and cryptographic one-way hash function. This scheme allows remote users to access multiple servers without separate registration. It eliminates the need of verification table, permits users to choose and change the password securely, provides mutual authentication session key generation. Moreover, it provides security against user impersonation attack, server impersonation attack, replay attack, reflection and parallel session attacks, password guessing attack, insider attack, smart card loss attack and stolen verifier attack. Thus, the proposed scheme is really appropriate in distributed multi-server environment such as the Internet.

3. PROPOSED SCHEME

This section describes the proposed smart card authentication scheme. Suppose, there are n servers with which the new user can communicate. Every user and server has to register first with the registration center. The notations used throughout this paper are summarized as follows:

RC	→	registration center
U_i	→	i^{th} remote user
ID_i	→	identity of U_i
PW_i	→	password chosen by U_i
PW_i*	→	password guessed by the adversary
S_j	→	j^{th} authentication server ($1 \le j \le n$)
SID_j	→	identity of S_j
x	→	secret key of RC
d	→	secret number of RC
p	→	large prime number
g	→	primitive element
h(•)	→	cryptographic one way hash function
$\oplus$	→	bitwise XOR operation
$SKey_{ij}$	→	session key shared between U_i and S_j
N_1	→	random nonce generated by U_i
N_2	→	random nonce generated by S_j

The scheme consists of four phases: Registration phase, Login phase, Authentication phase and Password Change phase.

3.1 Registration phase

This phase is divided into two sub-phases: Server Registration phase and User Registration phase.

3.1.1 Server Registration phase

In this phase, S_j selects SID_j and submits it to RC over a secure channel. Upon receiving the registration request from S_j, RC computes the server secret parameter $SS_j = (g^{h(SIDj,\ h(x))} \bmod p) \oplus h(d)$ and sends $\{SS_j, h(d)\}$ to S_j through a secure channel.

3.1.2 User Registration phase

U_i selects ID_i and PW_i, computes $h(PW_i)$ and submits $\{ID_i, h(PW_i)\}$ to RC over a secure channel. Once the registration request is received, RC computes $x_i = (g^{h(PWi)} \bmod p) \oplus h(x)$, $y_i = h(ID_i, h(d))$, $z_i = y_i \oplus h(PW_i)$ and issues a smart card over secure channel to U_i by storing $\{x_i, y_i, z_i, p, g, h(\bullet)\}$ into smart card memory.

3.2 Login phase

U_i inserts the smart card to the card reader and keys in ID_i and PW_i. The reader computes $z_i' = y_i \oplus h(PW_i')$ and checks whether computed z_i' equals stored z_i or not. If true, reader generates a random nonce N_1, computes $a_i = g^{yi} \bmod p$, $b_i = a_i^{yi \times N1} \bmod p$, $c_i = a_i^{h(PWi) \times N1} \bmod p$, $d_i = g^{h(PWi)} \bmod p$, $Q_j = g^{h(SIDj,\ (xi \oplus di))} \bmod p$, $e_i = (h(PW_i) + y_i \times h(ID_i, a_i, b_i, c_i, d_i, N_1, Q_j)) \bmod (p-1)$ and sends the login request $\{ID_i, SID_j, d_i, e_i, N_1\}$ to S_j.

3.3 Authentication phase

Upon receiving the login request $\{ID_i, SID_j, d_i, e_i, N_1\}$; S_j first checks the validity of ID_i to accept/reject the login request. If true, S_j computes $y_i = h(ID_i, h(d))$, $a_i = g^{yi} \bmod p$, $b_i = a_i^{yi \times N1} \bmod p$, $c_i = d_i^{yi \times N1} \bmod p$, $Q_j = SS_j \oplus h(d)$ and checks whether $g^{ei} = d_i \times a_i^{h(IDi,\ ai,\ bi,\ ci,\ di,\ N1,\ Qj)} \bmod p$ is true or not.

$g^{ei} = g^{(h(PWi) + yi \times h(IDi,\ ai,\ bi,\ ci,\ di,\ N1,\ Qj))} \bmod p,$

$g^{ei} = g^{h(PWi)} \times g^{yi \times h(IDi,\ ai,\ bi,\ ci,\ di,\ N1,\ Qj)} \bmod p,$

$g^{ei} = g^{h(PWi)} \bmod p \times g^{yi \times h(IDi,\ ai,\ bi,\ ci,\ di,\ N1,\ Qj)} \bmod p,$

$g^{ei} = d_i \times a_i^{h(IDi,\ ai,\ bi,\ ci,\ di,\ N1,\ Qj)} \bmod p.$

If this equation holds, S_j checks whether $a_i^{ei \times N1} = c_i \times b_i^{h(IDi,\ ai,\ bi,\ ci,\ di,\ N1,\ Qj)} \bmod p$ is true or not.

$a_i^{ei \times N1} = a_i^{(h(PWi) + yi \times h(IDi,\ ai,\ bi,\ ci,\ di,\ N1,\ Qj)) \times N1} \bmod p,$

$a_i^{ei \times N1} = a_i^{h(PWi) \times N1} \times a_i^{yi \times h(IDi,\ ai,\ bi,\ ci,\ di,\ N1,\ Qj) \times N1} \bmod p,$

$a_i^{ei \times N1} = a_i^{h(PWi) \times N1} \bmod p \times a_i^{yi \times N1 \times h(IDi,\ ai,\ bi,\ ci,\ di,\ N1,\ Qj)} \bmod p,$

$a_i^{ei \times N1} = c_i \times b_i^{h(IDi, ai, bi, ci, di, N1, Qj)} \bmod p$.

If both the equations hold, S_j generates a nonce N_2, computes $X_1 = y_i \oplus N_1 \oplus N_2$, $X_2 = Q_j^{N2} \bmod p$ and sends the message $\{ID_i, X_1, X_2\}$ to U_i. After getting the message $\{ID_i, X_1, X_2\}$ from S_j, U_i computes $N_2 = y_i \oplus X_1 \oplus N_1$, $X_2' = Q_j^{N2} \bmod p$ and checks whether X_2 and X_2' are equal or not. If it holds, S_j is authentic otherwise terminate the session. Subsequently, U_i computes $X_3 = Q_j^{N1 \times N2} \bmod p$ and sends $\{ID_i, X_3\}$ to S_j. Once the message $\{ID_i, X_3\}$ is received, S_j computes $X_3' = Q_j^{N1 \times N2} \bmod p$ and checks whether X_3 and X_3' are equal or not. If it holds, mutual authentication is achieved. Both the parties agree upon a common shared session key $SKey_{ij} = h(ID_i, SID_j, Q_j, N_1, N_2)$.

3.4 Password Change phase

This phase is invoked when U_i wants to change the password. U_i inserts the smart card to the card reader and keys in ID_i and PW_i'. The reader computes $z_i' = y_i \oplus h(PW_i')$ and checks whether computed z_i' equals stored z_i or not. If true, U_i enters a new password PW_{inew}. The card reader computes $z_{inew} = y_i \oplus h(PW_{inew})$, $x_{inew} = x_i \oplus g^{h(PWi)} \oplus g^{h(PWinew)} \bmod p$ and stores z_{inew}, x_{inew} instead of z_i, x_i respectively in the smart card memory. Thus, U_i can change the password without taking any assistance from S_j.

4. SECURITY ANALYSIS AND DISCUSSION

This section demonstrates the proof of correctness of the proposed authentication scheme on the basis of following possible attacks and user needed features.

4.1 User impersonation attack

The login request contains $\{ID_i, SID_j, d_i, e_i, N_1\}$ where $d_i = g^{h(PWi)} \bmod p$ and $e_i = (h(PW_i) + y_i \times h(ID_i, a_i, b_i, c_i, d_i, N_1, Q_j)) \bmod (p-1)$. To securely perform impersonation attack, the attacker needs to guess the correct values of PW_i, a_i, c_i and y_i. Let's assume attacker guesses the password PW_i*, the correct values of x_i and 'd' are still required to forge the login request. In addition, attacker is unable to extract any of the nonce values from the eavesdropped response message as the value of y_i is unknown. It is difficult to derive $h(PW_i)$ from d_i because of discrete logarithm problem. Hence, attacker is unable to forge the login request to impersonate a valid U_i.

4.2 Server impersonation attack

It is not possible for an adversary to masquerade as a legitimate server and try to cheat an authentic user because the server response message $\{ID_i, X_1, X_2\}$ is prepared by using the secret parameter Q_j which can be computed by S_j only. Hence, the proposed authentication scheme prevents server spoofing.

4.3 Replay attack

The replay attack will fail because the freshness of the messages transmitted in the login and authentication phases is provided by the random nonces N_1 and N_2. These are generated independently, and their values differ among sessions. As a result, attackers cannot enter the system by resending the previously transmitted messages to impersonate legal users. Assume that the intercepted login request $\{ID_i$, aintain any verification table, the proposed authentication scheme is secure against stolen-verifier attack.

$SID_j, d_i, e_i, N_1\}$ is replayed to pass the authentication phase. Attacker is unable to retrieve N_2 correctly from the response message $\{ID_i, X_1, X_2\}$ to compute the correct message $\{ID_i, X_3\}$ for mutual authentication. Consequently, S_j rejects the message by comparing X_3 with X_3'.

4.4 Reflection and parallel session attacks

To resist reflection and parallel session attacks, the given scheme employs asymmetric structure of communicating messages, i.e., $\{ID_i, SID_j, d_i, e_i, N_1\}$, $\{ID_i, X_1, X_2\}$ and $\{ID_i, X_3\}$. There is no symmetry in the values of $d_i = g^{h(PWi)} \bmod p$, $e_i = (h(PW_i) + y_i \times h(ID_i, a_i, b_i, c_i, d_i, N_1, Q_j)) \bmod (p-1)$, $X_1 = y_i \oplus N_1 \oplus N_2$, $X_2 = Q_j^{N2} \bmod p$ and $X_3 = Q_j^{N1 \times N2} \bmod p$. Hence, attacker is unable to launch parallel session attack by replaying server response message as the user login request or reflection attack by resending user login request as the server response message.

4.5 Password guessing attack

In the proposed scheme, $h(PW_i)$ is used to compute $d_i = g^{h(PWi)} \bmod p$ and $e_i = (h(PW_i) + y_i \times h(ID_i, a_i, b_i, c_i, d_i, N_1, Q_j)) \bmod (p-1)$. Let us assume that the adversary intercepts login request $\{ID_i, SID_j, d_i, e_i, N_1\}$ during the transmission from U_i to S_j. It is hard to guess the all three parameters a_i, c_i and y_i correctly at the same time to check whether each of the guessed passwords is correct or not. Moreover, to derive PW_i from d_i, adversary needs to solve the discrete logarithm problem and break the security of one way hash function. Therefore, the scheme is secure against password guessing attack.

4.6 Insider attack

Since, U_i registers to RC by presenting $h(PW_i)$ instead of PW_i, the insider of RC cannot directly obtain U_i's password PW_i because of the property of one-way hash function. Hence, the proposed scheme is able to resist insider attack.

4.7 Security of the session key

The session key $SKey_{ij} = h(ID_i, SID_j, Q_j, N_1, N_2)$ is associated with Q_j and N_2, which are unknown to the adversary. Even though the past session key is compromised, the adversary cannot extract these parameters due to the security of one-way hash function. Moreover, it is infeasible to guess both of these values simultaneously. Thus, the adversary cannot obtain any further session key.

4.8 Smart card loss attack

When a smart card is lost or stolen, unauthorized user, who obtains U_i's smart card, can guess the password of U_i by using password guessing attacks or impersonate U_i to login into S_j. In the proposed scheme, if U_i's smart card is lost or stolen, it is difficult for any attacker to derive or change the password PW_i. Moreover, no one can impersonate the smart card owner to login into S_j without knowing the correct ID_i and PW_i of U_i.

4.9 Stolen verifier attack

An adversary can penetrate the server and modify the contents of the verification table if the server maintains a verification table. As the servers and the registration center do not m

4.10 Single registration

It allows a valid user to register once and then the user can access all the registered servers.

4.11 No verification table

None of the registered servers need to maintain a verification table.

4.12 User can choose and change the password securely without taking any assistance from the server or RC

In the scheme, a valid user can change the password freely and securely without any assistance from the servers or registration center. As the card reader verifies the old password first in the password change phase, unauthorized users cannot change the authorized user's password even if they get the corresponding smart card.

4.13 Early wrong password detection

If the user U_i inputs a wrong password by mistake, this password will be quickly detected by the card reader itself since reader compares $z_i' = y_i \oplus h(PW_i')$ with the stored z_i during the login phase. Hence, the scheme provides early wrong password detection.

4.14 Each server uses unique secret parameter

In the scheme, each server has unique secret parameter $SS_j = (g^{h(SID_j,\ h(x))} \bmod p) \oplus h(d)$ used to authenticate the user. Hence, there is no need to store the secret parameter of all the servers in the smart card memory.

4.15 Mutual authentication and session key agreement without the support of RC

Valid users and valid servers can authenticate each other and then agree on a session key without any support from the registration center. The generated session key $SKey_{ij} = h(ID_i, SID_j, Q_j, N_1, N_2)$ will be different for each login session.

4.16 The scheme solves time synchronization problem

The proposed scheme uses randomly generated nonces N_1 and N_2 instead of timestamps to avoid time synchronization problem.

5. COMPARISON

This section describes comparison among various multi-server authentication schemes with our proposed scheme on the basis of security features as well as possible attacks. Table 1 shows this comparison. Here,

F1 = Free from maintaining verification table
F2 = User is allowed to choose the password
F3 = User is allowed to change the password
F4 = Free from involvement of RC/server during password change phase
F5 = Provides mutual authentication
F6 = Provides early wrong password detection
F7 = Provides mutual authentication without the support of RC
F8 = Provides session key agreement
F9 = Resists user impersonation attack
F10 = Resists server spoofing attack
F11 = Resists replay attack
F12 = Resists password guessing attack
F13 = Resists reflection attack
F14 = Resists parallel session attack
F15 = Resists known session key attack

Table 1. Comparison among various multi-server authentication schemes with our proposed scheme

Security Features	Juang [19]	Liao-Wang [22]	Hsiang-Shih [24]	Sood et al. [25]	Proposed Scheme
F1	No	Yes	Yes	No	Yes
F2	Yes	Yes	Yes	Yes	Yes
F3	No	Yes	Yes	Yes	Yes
F4	Yes	Yes	Yes	Yes	Yes
F5	Yes	Yes	Yes	Yes	Yes
F6	No	Yes	Yes	Yes	Yes
F7	Yes	Yes	No	No	Yes
F8	Yes	Yes	Yes	Yes	Yes
F9	Yes	No	No	Yes	Yes
F10	Yes	No	No	Yes	Yes
F11	Yes	Yes	No	Yes	Yes
F12	Yes	Yes	No	Yes	Yes
F13	Yes	Yes	Yes	Yes	Yes
F14	Yes	Yes	Yes	Yes	Yes
F15	Yes	Yes	Yes	Yes	Yes

6. CONCLUSION

This paper portraits robust smart card authentication scheme for multi-server environment. It is shown that the proposed scheme satisfies all of the essential security requirements as it is secure against user and server impersonation attacks, replay attack, reflection and parallel session attacks, password guessing attack, stolen verifier attack, insider attack and smart card loss attack. The other merits include: (1) it eliminates the use of verification table; (2) it allows users to choose and change their passwords freely without taking any assistance from the server or registration center; (3) it permits users to access multiple servers without separately registering with each server; (4) it detects wrong password early; (5) it provides mutual authentication and session key agreement; (6) it is a nonce based scheme to avoid the time-synchronization problem.

7. ACKNOWLEDGMENTS

The authors are thankful to Institute of Engineering and Technology, Alwar, India for providing the academic support.

8. REFERENCES

[1] Lamport, L. 1981. Password authentication with insecure communication. Communications of the ACM 24, 770-772.

[2] Wu, T. C. 1995. Remote login authentication scheme based on a geometric approach. Computer Communications 18, 959-963.

[3] Hwang, M. S. 1999. Cryptanalysis of a remote login authentication scheme. Computer Communications 22, 742-744.

[4] Yang, W. H, Shieh, S. P. 1999. Password authentication schemes with smart cards. Computers & Security 18, 727-733.

[5] Chan, C. K, Cheng, L. M. 2002. Cryptanalysis of timestamp-based password authentication scheme. Computers & Security 21, 74-76.

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

[7] Chan, C. K, Cheng, L. M. 2000. Cryptanalysis of a remote user authentication scheme using smart cards. IEEE Transactions on Consumer Electronics 46, 992-993.

[8] Sun, H. M. 2000. An efficient remote user authentication scheme using smart cards. IEEE Transactions on Consumer Electronics 46, 958-961.

[9] Hsu, C. L. 2003. Security of two remote user authentication schemes using smart cards. IEEE Transactions on Consumer Electronics 49, 1196-1198.

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

[11] Juang, W. S. 2004. Efficient password authenticated key agreement using smart cards. Computers & Security 23, 167-173.

[12] Das, M. L., Saxena, A., Gulati, V. P. 2004. A dynamic ID-based remote user authentication scheme. IEEE Transactions on Consumer Electronics 50, 629-631.

[13] Liao, I. E, Lee, C. C. and Hwang, M. S. 2005. Security enhancement for a dynamic ID-based remote user authentication scheme. In Proceedings of the International Conference on Next Generation Web Services Practices.

[14] Song, R. 2010. Advanced smart card based password authentication protocol. Computer Standards & Interfaces 32, 321-325.

[15] Pippal, R. S., Jaidhar, C. D. and Tapaswi, S. 2010. Comments on symmetric key encryption based smart card authentication scheme. In Proceedings of the 2nd International Conference on Computer Technology and Development.

[16] Li, L., Lin, I., Hwang, M. S. 2001. A remote password authentication scheme for multi-server architecture using neural networks. IEEE Transaction on Neural Networks 12, 1498-1504.

[17] Lin, I. C., Hwang, M. S., Li, L. H. 2003. A new remote user authentication scheme for multi-server architecture. Future Generation Computer Systems 19, 13-22.

[18] Cao, X., Zhong, S. 2006. Breaking a remote user authentication scheme for multi-server architecture. IEEE Communications Letters 10, 580-581.

[19] Juang, W. S. 2004. Efficient multi-server password authenticated key agreement using smart cards. IEEE Transactions on Consumer Electronics 50, 251-255.

[20] Ku, W. C., Chuang, H. M., Chiang, M. H. and Chang, K. T. 2005. Weaknesses of a multi-server password authenticated key agreement scheme. In Proceedings of the 2005 National computer Symposium.

[21] Chang, C. C., Lee, J. S. 2004. An efficient and secure multi-server password authentication scheme using smart cards. In Proceedings of the International Conference on Cyberworlds.

[22] Liao, Y. P., Wang, S. S. 2009. A secure dynamic ID based remote user authentication scheme for multi-server environment. Computer Standards & Interfaces 31, 24-29.

[23] Chen, T. Y., Hwang, M. S., Lee, C. C. and Jan, J. K. 2009. Cryptanalysis of a secure dynamic ID based remote user authentication scheme for multi-server environment. In Proceedings of the 4th International Conference on Innovative Computing, Information and Control.

[24] Hsiang, C., Shih, W. K. 2009. Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment. Computer Standards & Interfaces 31, 1118-1123.

[25] Sood, S. K., Sarje, A. K., Singh, K. 2011. A secure dynamic identity based authentication protocol for multi-server architecture. Journal of Network and Computer Applications 34, 609-618.

[26] Tsai, J. L. 2008. Efficient multi-server authentication scheme based on one-way hash function without verification table. Computers & Security 27, 115-121.

[27] Zhu, H, Liu, T. and Liu, J. 2009. Robust and simple multi-server authentication protocol without verification table. In Proceedings of the 9th International Conference on Hybrid Intelligent Systems.