Cryptanalysis and Improvement of a Two-Factor User Authentication Scheme

Devender Kumar, Satish Chand, and Bijendra Kumar

***Abstract*—Recently, Wang-Wang have discussed a two birds with one stone: two-factor authentication with security beyond conventional bound. We find that this scheme is vulnerable to the password exposure attack and also does not offer user anonymity, which is an important feature for some of the applications like e-healthcare services, e-banking, etc. In this paper, we provide the solution to these problems.**

***Index Terms*—Two-factor authentication, password exposure attack, user anonymity, smart card, offline password guessing attack, insider attack**

I. INTRODUCTION

In the era of internet, most of the resources and services are available online. However, the security is an important issue to access online resources and services. A remote user authentication scheme can help to access online resources and services securely. Such scheme allows a user and a server to authenticate each other over an insecure channel. In 1981, Lamport [1] developed the first remote user authentication scheme in which the server was required to keep a password table. Since then, many smart card based remote user authentication schemes [2], [3], [4], [5], [6], [7], [8] have been discussed that do not require password tables.

In 2009, Xu et al. discussed a user authentication scheme based on smart card [9] and claimed that it is secured even if the smart card is lost. Sood et al. [10] found that the scheme [9] is not resistant to forgery attack and they improved it by overcoming its weakness. The paper [11] cryptanalyzed the scheme [9] and found that it is not resistant to the impersonation attack if a valid but malicious user uses the information stored in his own smart card. They improved this scheme to overcome its limitation. Horng et al. [12] found that the scheme [11] is not resistant to the insider and offline password guessing attacks.

In 2014, Chen et al. [13] cryptanalyzed the schemes [10], [11] and they found that the scheme [10] does not offer mutual authentication and the scheme [11] is not resistant to the smart card loss and off-line guessing attacks. They designed an improved scheme to remove these flaws. Jiang et al. [14] found that the scheme [13] is not secured against the offline password guessing attack and designed an improved scheme to overcome this problem. Mishra et al. [15] discussed the security issues of the scheme [14] and showed that it is susceptible to the insider, user impersonation and password guessing attacks. They designed a scheme to overcome these security flaws.

Recently, Wang-Wang [16] have discussed a two factor authentication scheme and suggested twelve independent security criteria that a two-factor authentication scheme should satisfy as follows: (i) no verifier-table (ii) no password exposure (iii) no smart card loss attack (iv) password friendly (v) resistance to known attacks (vi) provision of key agreement (vii) sound repairability (viii) no clock synchronization (ix) mutual authentication (x) timely typo detection (xi) user anonymity (xii) forward secrecy. Out these, user anonymity and password exposure are the essential properties of an user authentication scheme. User anonymity means user identity-protection and un-traceability. That is the scheme should protect user identity and prevent user activities from tracing. Password exposure means that the privileged administrator cannot get the user's password. In this paper, we analyze the security of the scheme [16] and find that it is susceptible to the password exposure attack and also lacks user anonymity. We present an improved scheme to overcome its limitations.

A. Threat model

Here, we present the capabilities of an attacker A as follows:

- A can eavesdrop all the transmitted messages between the participants over a public channel.
- A can reroute, resend, delete, modify and insert the eavesdropped messages.
- A can take out all the information saved in the smart card of a valid user if it is obtained by A somehow [17], [18].
- A cannot know the user's password as well as steal the user's smart card at the same time.
- A can enumerate offline all possible elements in the cartesian product $D_{id} \times D_{pw}$ in a reasonable amount of time [16].
- The privileged administrator may act as an attacker A.

The remaining paper is arranged as follows: section II reviews the Wang-Wang's scheme in brief and section III presents its cryptanalysis. Section IV introduces our proposed scheme and its performance analysis is presented in section V. Its formal security analysis is same as that of the Wang-Wang's scheme as we do not change the parameters which are transmitted via a public channel and hence it is omitted. Finally, section VI concludes the paper.

II. REVIEW OF WANG-WANG'S SCHEME

Here, we briefly review the robust password authentication scheme using smart card by Wang-Wang [16] that consists of

D. Kumar is with the Division of Information Technology, NSIT, New Delhi, India-110078, Phone:+919013489217, Fax: +91-11-25099022, e-mail: dk_iitm@yahoo.co.in.

S. Chand is with School of Computer and Systems Sciences, JNU, New Delhi, India-110067.

B. Kumar is with the Division of Computer Engineering, NSIT, New Delhi, India-110078.

DOI: 10.36244/ICJ.2018.1.2

the following four phases. The notations used in this paper are given in Table I.

TABLE I
NOTATIONS USED IN PAPER

Notations	Description
U_i	i^{th} User
S	Remote server
A	Attacker
ID_i	U_i's identity
PW_i	U_i's password
x	S's secret key
y	S's public key
p, q	Large prime numbers
n_0	An integer
$Honey\_List$	Link list
m_0	Number of items in $Honey\_List$
$H_i(.)$	One-way hash function
g	Generator of a prime order cyclic group G
$\|\|$	Concatenation operator
SC	Smart card
$\oplus$	XOR operator

A. Registration phase

User U_i performs the below steps to register with server S:

1) U_i selects his identity ID_i, password PW_i, and a random string b.
2) He sends $\{ID_i, H_0(b||PW_i)\}$ to S through a private channel.
3) After obtaining the request from U_i at time T, S chooses a random number a_i and calculates $A_i = H_0((H_0(ID_i) \oplus H_0(b||PW_i)) \ mod \ n_0)$. S verifies if U_i is a registered user. If not, then S stores the information $\{ID_i, T_{reg} = T, a_i, Honey\_List = NULL\}$ in its database; otherwise, it replaces the value of T_{reg} with T, a_i with newly selected a_i, and $Honey\_List$ with $NULL$ in its database corresponding to U_i. S then calculates $N_i = H_0(b||PW_i) \oplus H_0(x||ID_i||T_{reg})$.
4) S stores the information $\{N_i, A_i, A_i \oplus a_i, q, g, y, n_0, H_0(.), ...H_3(.)\}$ in a SC and transmits it to U_i securely.
5) After obtaining the smart card, U_i stores b into it; thus, the smart card contains $\{N_i, A_i, A_i \oplus a_i, q, g, y, n_0, H_0(.), ..., H_3(.), b\}$

B. Login phase

The steps are performed as below in this phase:

1) U_i inputs his identity ID_i^* and password PW_i^* after inserting his SC into the card reader attached with the system.
2) SC calculates $A_i^* = H_0((H_0(ID_i^*) \oplus H_0(b||PW_i^*)) \ mod \ n_0)$ and checks if $A_i^* = A_i$. If it is not true, the session is terminated.
3) SC selects a random number u and calculates $C_1 = g^u \ mod \ p$, $Y_1 = y^u \ mod \ p$, $k = H_0(x||ID_i||T_{reg}) = N_i \oplus H_0(b||PW_i^*)$, $a_i = (A_i \oplus a_i) \oplus A_i$, $CID_i = ID_i^* \oplus H_0(C_1||Y_1)$, $CAK_i = (a_i||k) \oplus H_0(C_1||Y_1)$, and $M_i = H_0(Y_1||k||CID_i||CAK_i)$.
4) U_i sends the message $\{C_1, CID_i, CAK_i, M_i\}$ to S through a public channel.

C. Verification phase

On getting the login message $\{C_1, CID_i, CAK_i, M_i\}$ from U_i, S performs the below steps:

1) S calculates $Y_1 = (C_1)^x \ mod \ p$ and $ID_i = CID_i \oplus H_0(C_1||Y_1)$. It verifies the format of ID_i. If it is not found in correct format, then the session is terminated.
2) S calculates $k = H_0(x||ID_i||T_{reg})$ and $M_i^* = H_0(Y_1||k||CID_i||CAK_i)$, where T_{reg} is excerpted from its database corresponding to the entry ID_i. It checks if $M_i^* = M_i$. If it is false, the session is terminated.
3) S computes $a_i^{`}||k^{`} = CAK_i \oplus H_0(C_1||Y_1)$ and verifies if $a_i^{`}$ is equal to the stored $a_i$. If it is false, $S$ rejects the request; otherwise, it check if $k^{`} = k$. If it is true, then perform next step; otherwise, if $a_i^{`} = a_i$ and $k_i^{`} \neq k_i$, then S concludes that the card of U_i is corrupted with a probability $1 - \frac{1}{2^{n_0}}$. In that case, S either enters $k^{`}$ into Honey_List if $|Honey\_List| < m_0$ (e.g. $m_0 = 10$) or suspends the smart card of U_i until he re-registers (i.e. when $|Honey\_List| = m_0$).
4) S creates a random number v and calculates the temporary key $K_S = (C_1)^v \ mod \ p, C_2 = g^v \ mod \ p$ and $C_3 = H_1(ID_i||ID_S||Y_1||C_2||k||K_S)$. S sends the message $\{C_2, C_3\}$ to U_i via a public channel.
5) After getting the message $\{C_2, C_3\}$ from S, SC calculates $K_U = (C_2)^u \ mod \ p, C_3^* = H_1(ID_i||ID_S||Y_1||C_2||k||K_U)$, and checks if $C_3^* = C_3$. If it is right, U_i authenticates S and calculates $C_4 = H_2(ID_i||ID_S||Y_1||C_2||k||K_U)$. U_i sends the message $\{C_4\}$ to S via an insecure channel.
6) On obtaining the message $\{C_4\}$ from U_i, S calculates $C_4{}^* = H_2(ID_i||ID_S \ ||Y_1||C_2||k||K_S)$, and checks if $C_4^* = C_4$. If it is true, S authenticates U_i and accepts the login request; otherwise, the session is terminated.
7) U_i and S share the session key $sk_U = H_3(ID_i||ID_S||Y_1||C_2||k||K_U) = H_3(ID_i||ID_S||Y_1|| C_2||k||K_S) = sk_S$ for secured future communication.

D. Password change phase

User U_i performs the below steps in this phase:

1) U_i inputs his ID_i and PW_i after inserting his SC into the card reader attached with the system.
2) SC calculates $A_i^* = H_0((H_0(ID_i) \oplus H_0(b||PW_i)) \ mod \ n_0)$ and checks if $A_i^* = A_i$. If it is not true, the request for changing password is rejected.
3) Smart card prompts U_i to enter a new password PW_i^{new} and calculates $N_i^{new} = N_i \oplus H_0(b||PW_i) \oplus H_0(b||PW_i^{new})$ and $A_i^{new} = H_0((H_0(ID_i) \oplus H_0(b||PW_i^{new})) \ mod \ n_0)$. It replaces the values of N_i, A_i and $a_i \oplus A_i$ with N_i^{new}, A_i^{new} and $a_i \oplus A_i^{new}$, respectively. Thus the password is changed successfully.

III. CRYPTANALYSIS OF WANG-WANG'S SCHEME

We cryptanalyze the Wang-Wang's scheme [16] based on the threat model as given in section I-A and find the following security problems:

A. Password exposure attack

Since user U_i sends $\{ID_i, H_0(b||PW_i)\}$ to S in step (2) of registration phase, a malicious privileged administrator A has knowledge of these two parameters. Assume that A somehow gets access to the U_i's smart card [19], then he can find U_i's password PW_i as follows:

1) Choose a password PW_i^* and compute $H_0(b||PW_i^*)$
2) Check if $H_0(b||PW_i^*) = H_0(b||PW_i)$. If it is true, then A gets the correct password PW_i of U_i and stops the procedure. Otherwise, repeat the steps (1) and (2).

Hence, user U_i's password is not safe from a malicious privileged administrator in this scheme.

B. User anonymity

Since user U_i sends the message $\{ID_i, H_0(b||PW_i)\}$ to S in step (2) of registration phase, his identity ID_i is transmitted in plaintext. Thus, his identity is not anonymous from a malicious privileged administrator A.

IV. OUR SCHEME

In this section, we present our improved scheme by overcoming the weaknesses of the scheme [16]. In the registration phase of our scheme, we send the hash value of the user's identity ID_i and random string b instead of sending ID_i directly in plaintext to provide user anonymity. To resist from password exposure attack, we store the encrypted value of the random string b using XOR operation in the memory of the smart card in the registration phase. Our scheme consists of the following four phases:

A. Registration phase

User U_i executes the below steps to register with server S:

1) U_i selects his identity ID_i, password PW_i, and a random string b.
2) He sends $\{H_0(b||ID_i), H_0(b||PW_i)\}$ to S through a secure channel.
3) After obtaining the registration message from U_i at time T, S selects a random number a_i and calculates $A_i = H_0((H_0(b||ID_i) \oplus H_0(b||PW_i))\ mod\ n_0)$. S verifies from its database whether U_i is a registered user. If not, S stores the information $\{H_0(b||ID_i), T_{reg} = T, a_i, Honey\_List = NULL\}$ in its database; otherwise, it replaces the value of T_{reg} with T, a_i with newly selected a_i, and $Honey\_List$ with NULL in its database corresponding to U_i. Then, S calculates $N_i = H_0(b||PW_i) \oplus H_0(x||H_0(b||ID_i)||T_{reg})$.
4) S stores the information $\{N_i, A_i, A_i \oplus a_i, q, g, y, n_0, H_0(.), ...H_3(.)\}$ in a SC and sends it to U_i securely.
5) After obtaining the smart card, U_i computes $c = b \oplus H_0(ID_i \oplus PW_i)\ mod\ n_0$ and stores c into it and finally the SC contains the data $\{N_i, A_i, A_i \oplus a_i, q, g, y, n_0, H_0(.), ...H_3(.), c\}$

B. Login phase

The following steps are executed in this phase:

1) User U_i inputs his identity ID_i^* and password PW_i^* after inserting his SC into the card reader attached with the system.
2) SC calculates $b = c \oplus H_0(ID_i^* \oplus PW_i^*)\ mod\ n_0$ and $A_i^* = H_0((H_0(b||\ ID_i^*) \oplus H_0(b||PW_i^*))\ mod\ n_0)$ and checks if $A_i^* = A_i$. If it is not true, the session is terminated.
3) SC selects a random number u and computes $C_1 = g^u\ mod\ p$, $Y_1 = y^u\ mod\ p$, $k = H_0(x||H_0(b||ID_i^*)||T_{reg}) = N_i \oplus H_0(b||PW_i^*)$, $a_i = (A_i \oplus a_i) \oplus A_i^*$, $CID_i = H_0(b||ID_i^*) \oplus H_0(C_1||Y_1)$, $CAK_i = (a_i||k) \oplus H_0(C_1||Y_1)$, and $M_i = H_0(Y_1||k||CID_i||CAK_i)$.
4) U_i sends the message $\{C_1, CID_i, CAK_i, M_i\}$ to S via a public channel.

C. Verification phase

On obtaining the login request $\{C_1, CID_i, CAK_i, M_i\}$ from U_i, S executes the following steps:

1) S calculates $Y_1 = (C_1)^x\ mod\ p$ and $H_0(b||ID_i) = CID_i \oplus H_0(C_1||Y_1)$. It checks the entry of $H_0(b||ID_i)$ in its database. If it is not found, the session is rejected.
2) S calculates $k = H_0(x||H_0(b||ID_i)||T_{reg})$ and $M_i^* = H_0(Y_1||k||CID_i||CAK_i)$, where T_{reg} is excerpted from its database corresponding to the entry $H_0(b||ID_i)$. It checks if $M_i^* = M_i$. If it is false, the session is terminated.
3) S computes $a_i^{`}||k^{`} = CAK_i \oplus H_0(C_1||Y_1)$ and verifies if $a_i^{`}$ is equal to the stored $a_i$. In case of inequality, $S$ denies the request; otherwise, it check if $k^{`} = k$. If it is true, then perform next step; otherwise, if $a_i^{`} = a_i$ and $k_i^{`} \neq k_i$, then S concludes that the card of U_i is corrupted with a probability $1 - \frac{1}{2^{n_0}}$. In that case, S either enters $k^{`}$ into Honey_List if $|Honey\_List| < m_0$ (e.g. $m_0 = 10$) or suspends the smart card of U_i until he re-registers (i.e. when $|Honey\_List| = m_0$).
4) S creates a random number v and calculates the temporary key $K_S = (C_1)^v\ mod\ p, C_2 = g^v\ mod\ p$ and $C_3 = H_1(H_0(b||ID_i)||ID_S||Y_1||C_2||k||K_S)$. S sends the message $\{C_2, C_3\}$ to U_i via an insecure channel.
5) After obtaining the message $\{C_2, C_3\}$ from S, the SC calculates $K_U = (C_2)^u\ mod\ p, C_3^* = H_1(H_0(b||ID_i)||ID_S||Y_1||C_2||k||K_U)$, and checks if $C_3^* = C_3$. If it is true, U_i authenticates S and calculates $C_4 = H_2(H_0(b||ID_i)\ ||ID_S||Y_1||C_2||k||K_U)$. U_i sends the message $\{C_4\}$ to S via a public channel.
6) After obtaining the message $\{C_4\}$ from U_i, S calculates ${C_4}^* = H_2(H_0(b||\ ID_i)||ID_S||Y_1||C_2||k||K_S)$, and checks if $C_4^* = C_4$. If it is true, S authenticates U_i and accepts his login request; otherwise, the session is terminated.
7) U_i and S share the session key $sk_U = H_3(H_0(b||ID_i))||\ ID_S||Y_1||C_2||k||K_U) = H_3(H_0(b||ID_i)||ID_S||Y_1||C_2\ ||k||K_S) = sk_S$ for secured future communication.

D. Password change phase

User U_i performs the following steps to change his password:

1) U_i inputs his ID_i and PW_i after inserting his SC into the card reader attached with the system.
2) SC calculates $b = c \oplus H_0(ID_i \oplus H_0(PW_i)$ and $A_i^* = H_0((H_0(b||ID_i) \oplus H_0(b||PW_i)) \ mod \ n_0)$ and checks if $A_i^* = A_i$. If it is not true, the request for changing password is rejected.
3) Smart card prompts U_i to enter a new password PW_i^{new} and calculates $N_i^{new} = N_i \oplus H_0(b||PW_i) \oplus H_0(b||PW_i^{new})$ and $A_i^{new} = H_0((H_0(b||ID_i) \oplus H_0(b||PW_i^{new})) \ mod \ n_0)$. It replaces the values of N_i, A_i and $a_i \oplus A_i$ with N_i^{new}, A_i^{new} and $a_i \oplus A_i^{new}$, respectively. Thus the password is changed successfully.

V. PERFORMANCE ANALYSIS

In this section, we compare our scheme with that of the related schemes [20], [14], [21], [22], [16] in terms of communication cost, computational cost and security fetures. Like in other works, we have not considered the cost of lightweight operations like exclusive-or and concatenation operations. We have taken the length of parameter n_0 as $32\,bits$ and the user identity ID_i, password PW_i, random numbers, timestamps, and output of hash function have taken as $128\,bits$ long each; while the lengths of y and g are taken as $1024\ bits$ each, similar to that in the scheme [16].

From Table II, it is evident that the scheme [14] has the highest communication cost $(3456\,bits)$. The communication cost of our scheme is same as that of the scheme [16]; however, the scheme [16] does not provide the security features like password exposure and user anonymity as shown in Table IV. The scheme [22] has the least communication cost, i.e. $(1792\ bits)$; however, it does not provide the security features like password exposure, smart card loss attack, sound repairability and user anonymity. Thus, our scheme has better performance than the related schemes [20], [14], [21].

TABLE II
COMMUNICATION COST

Scheme	Communication cost(bits)
Islam [20]	$1408 + 1408 = 2816$
Jiang et al. [14]	$2304 + 1152 = 3456$
Bym [21]	$2176 + 1152 = 3328$
Truong [22]	$640 + 1152 = 1792$
Wang-Wang [16]	$1536 + 1152 = 2688$
Ours	$1536 + 1152 = 2688$

Table III presents the computational cost of our scheme along with the related schemes [20], [14], [21], [22], [16] in login and authentication phases. The computational cost of the schemes [20], [14], [21], [22], [16] and our scheme are, respectively, $5t_e+6t_h$, $6t_e+8t_h$, $10t_e+2t_s+8t_h$, $4t_c+14t_h$, $6t_e+16t_h$ and $6t_e+17t_h$. The scheme [21] has the higher computaional cost as compared to that of ours and does not offer the security features like verifier table, password friendly and timely typo detection. The scheme [20] has the least computation cost; but it suffers from smart card loss attack. The scheme [16] only takes one hash function less than ours; however, it does not provide the security features like password exposure and user anonymity as shown in Table IV. Thus, our scheme satisfies all the security features while others do not as given in Table IV.

TABLE III
COMPUTATION COST

Scheme	User	Server	Sum
Islam [20]	$3t_e + 3t_h$	$2t_e + 3t_h$	$5t_e + 6t_h$
Jiang et al. [14]	$4t_e + 4t_h$	$2t_e + 4t_h$	$6t_e + 8t_h$
Bym [21]	$5t_e + t_s + 5t_h$	$5t_e + t_s + 3t_h$	$10t_e + 2t_s + 8t_h$
Truong [22]	$t_c + 7t_h$	$3t_c + 7t_h$	$4t_c + 14t_h$
Wang-Wang [16]	$3t_e + 9t_h$	$3t_e + 7t_h$	$6t_e + 16t_h$
Ours	$3t_e + 10t_h$	$3t_e + 7t_h$	$6t_e + 17t_h$

t_h: time complexity of hash operation; t_e: time complexity of exponentiation operation; t_s: time complexity of encryption/decryption of symmetric key cryptography; t_c: time complexity of Chebysev polynomial

TABLE IV
SECURITY FEATURES

Security features	[20]	[14]	[21]	[22]	[16]	Ours
Verifier-table	Yes	Yes	No	Yes	Yes	Yes
Password exposure	Yes	No	Yes	No	No	Yes
Password friendly	Yes	Yes	No	Yes	Yes	Yes
Smart card loss attack	No	Yes	Yes	No	Yes	Yes
Known attacks	Yes	Yes	Yes	Yes	Yes	Yes
Provision of key agreement	Yes	Yes	Yes	Yes	Yes	Yes
Timely typo detection	Yes	Yes	No	Yes	Yes	Yes
Clock synchronization	Yes	No	Yes	Yes	Yes	Yes
Sound repairability	Yes	No	Yes	No	Yes	Yes
Mutual authentication	Yes	Yes	Yes	Yes	Yes	Yes
Forward secrecy	Yes	No	Yes	Yes	Yes	Yes
User anonymity	Yes	No	Yes	No	No	Yes

VI. CONCLUSION

In this paper, we have cryptanalyzed the security of the Wang-Wang's scheme and found that it does not provide user anonymity and suffers from the password exposure attack. We have improved this scheme by overcoming its limitations. Further, we have shown that our scheme is more secured than the existing schemes.

REFERENCES

[1] L. Lamport, “Password authentication with insecure communication,” *Communications of the ACM*, vol. 24, no. 11, pp. 770–772, 1981.

[2] H.-Y. Chien, J.-K. Jan, and Y.-M. Tseng, “An efficient and practical solution to remote authentication: smart card,” *Computers & Security*, vol. 21, no. 4, pp. 372–375, 2002.

[3] E.-J. Yoon, E.-K. Ryu, and K.-Y. Yoo, “Further improvement of an efficient password based remote user authentication scheme using smart cards,” *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 612–614, 2004.

[4] C.-I. Fan, Y.-C. Chan, and Z.-K. Zhang, “Robust remote authentication scheme with smart cards,” *Computers & Security*, vol. 24, no. 8, pp. 619–628, 2005.

[5] X.-M. Wang, W.-F. Zhang, J.-S. Zhang, and M. K. Khan, “Cryptanalysis and improvement on two efficient remote user authentication scheme using smart cards,” *Computer Standards & Interfaces*, vol. 29, no. 5, pp. 507–512, 2007.

[6] K.-H. Yeh, C. Su, N.-W. Lo, Y. Li, and Y.-X. Hung, “Two robust remote user authentication protocols using smart cards,” *Journal of Systems and Software*, vol. 83, no. 12, pp. 2556–2565, 2010.

[7] S. Kumari and M. K. Khan, “Cryptanalysis and improvement of ’a robust smart-card-based remote user password authentication scheme’,” *International Journal of Communication Systems*, vol. 27, no. 12, pp. 3939–3955, 2014.

[8] V. Odelu, A. K. Das, M. Wazid, and M. Conti, "Provably secure authenticated key agreement scheme for smart grid," *IEEE Transactions on Smart Grid*, 2016.

[9] J. Xu, W.-T. Zhu, and D.-G. Feng, "An improved smart card based password authentication scheme with provable security," *Computer Standards & Interfaces*, vol. 31, no. 4, pp. 723–728, 2009.

[10] S. K. Sood SK, Sarje AK, "An improvement of xu et al.'s authentication scheme using smart cards," in *Proceedings of The Third Annual ACM Bangalore Conference, Bangalore, Karnataka, India*, pp. 1–5, 2010.

[11] R. Song, "Advanced smart card based password authentication protocol," *Computer Standards & Interfaces*, vol. 32, no. 5, pp. 321–325, 2010.

[12] W.-B. Horng, C.-P. Lee, and J.-W. Peng, "Security weaknesses of song's advanced smart card based password authentication protocol," in *Progress in Informatics and Computing (PIC), 2010 IEEE International Conference on*, vol. 1, pp. 477–480, IEEE, 2010.

[13] B.-L. Chen, W.-C. Kuo, and L.-C. Wuu, "Robust smart-card-based remote user password authentication scheme," *International Journal of Communication Systems*, vol. 27, no. 2, pp. 377–389, 2014.

[14] Q. Jiang, J. Ma, G. Li, and X. Li, "Improvement of robust smart-card-based password authentication scheme," *International Journal of Communication Systems*, vol. 28, no. 2, pp. 383–393, 2015.

[15] D. Mishra, A. Chaturvedi, and S. Mukhopadhyay, "Cryptanalysis and improvement of jiang et al.'s smart card based remote user authentication scheme," *arXiv preprint arXiv:1312.4793*, 2013.

[16] D. Wang and P. Wang, "Two birds with one stone: Two-factor authentication with security beyond conventional bound," *IEEE Transactions on Dependable and Secure Computing*, 2016.

[17] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Annual International Cryptology Conference*, pp. 388–397, Springer, 1999.

[18] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE transactions on computers*, vol. 51, no. 5, pp. 541–552, 2002.

[19] A. K. Das, "A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks," *Peer-to-peer Networking and Applications*, vol. 9, no. 1, pp. 223–244, 2016.

[20] S. Islam, "Design and analysis of an improved smartcard-based remote user password authentication scheme," *International Journal of Communication Systems*, vol. 29, no. 11, pp. 1708–1719, 2016.

[21] J. W. Byun, "Privacy preserving smartcard-based authentication system with provable security," *Security and Communication Networks*, vol. 8, no. 17, pp. 3028–3044, 2015.

[22] T.-T. Truong, M.-T. Tran, A.-D. Duong, and I. Echizen, "Chaotic chebyshev polynomials based remote user authentication scheme in client-server environment," in *IFIP International Information Security Conference*, pp. 479–494, Springer, 2015.

Devender Kumar received his M.Sc. (Mathematics) from Panjab University, Chandigarh and M.Tech. (Computer Science and Engineering) from IIT, Madras. Currently, he is an Assistant Professor in Netaji Subhas Institute of Technology, New Delhi and pursuing his Ph.D. in Information Technology from University of Delhi. His current research interests include cryptography and network security.

Satish Chand did his M.Sc. (Mathematics) from IIT, Kanpur and M.Tech. (Comp. Sc.) from IIT, Kharagpur and Ph.D. (Comp. Sc.) from Jawaharlal Lal Nehru University, Delhi. Currently he is a Professor in School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi. He is Director, placement cell, Jawaharlal Nehru University, besides being a member in several other committees. He has been made Chairman, All India Board of Information Technology, by AICTE. He has credit to publish about hundred research papers in international/national journals and conferences of high repute. He works in different areas that include video processing, image processing, video broadcasting, Digital steganography, Image forensics, wireless sensor networks, network security, etc.

Bijendra Kumar did his Bachelor of Engineering from H.B.T.I. Kanpur, Uttar Pradesh, India and his Ph.D. from University of Delhi, New Delhi, India. Presently, he is a Professor in Division of Computer Engineering at Netaji Subhas Institute of Technology, New Delhi, India. His areas of research interests are Video applications, cryptography, watermarking, and design of algorithms.