An Improved Efficient Remote Password Authentication Scheme with Smart Card over Insecure Networks

Manoj Kumar[1], Mridul Kumar Gupta[2], and Saru Kumari[3]

(Corresponding author: Saru Kumari)

Department of Mathematics, R. K. College, Shamli (Muzaffarnagar), Uttar Pradesh, India[1]

Department of Mathematics, Chaudhary Charan Singh University, Meerut, Uttar Pradesh, India[2]

Department of Mathematics, Agra College, District-Agra, Pin-282004, Uttar Pradesh, India[3]

(Email: yamu_balyan@yahoo.co.in, mkgupta2002@hotmail.com, saryusiirohi@gmail.com)

(Received Oct. 18, 2010; revised and accepted Feb. 9 & Apr. 20, 2011)

Abstract

In 2006, Liao *et al.* proposed a scheme over insecure networks. In 2006, Yoon-Yoo, and in 2008, Xiang *et al.* analyzed Liao *et al.*'s scheme and both of them pointed out, more or less, same vulnerabilities: like offline password guessing attack, impersonating the server by replay attack, denial of service attack on password changing and insider attack on it. But none of them suggested any solution to the pointed out attacks. This paper proposes an improved scheme with enhanced security, maintaining advantages of the original scheme and free from the attacks pointed out by Yoon-Yoo and Xiang *et al.*.

Keywords: Authentication, password, remote login, smart card, insecure network

1 Introduction

Remote user authentication scheme allows a server to authenticate a remote user over insecure networks. Usually most of the systems use the following two methods to identify a user:

- Identity authentication of something known, such as password.
- Identity authentication of something possessed, such as smart card.

The technology using both of the above methods is termed as two factor authentication. A smart card based password authentication scheme involves an authentication server AS and a user U (with identity ID). The scheme generally consists of three phases: registration phase, login phase and authentication phase. In registration phase U sends a registration request to AS, and AS securely issue a smart card to U. This smart card is personalized with respect to ID and PW of U. This phase is carried out only once for each user and is carried on secure network. In login phase U sends a login request to AS in order to access the facilities provided by AS. Authentication phase deals with verifying the legitimacy of U and AS so as to achieve the mutual authentication.

As the login phase and authentication phase are conducted over insecure networks, so adversaries can modify, remove or insert messages into the channel. If mutual authentication is perfect between U and AS then the security goal of the scheme is achieved. Besides the above three phases, password change phase is also added to the scheme in case U needs to change his password. U can change his password using his smart card either interacting with AS or without interacting with AS. We promote the idea of letting U change the password at will without interacting with the AS.

1.1 Related and Proposed Work

Since Lamport [17] introduced a remote user authentication scheme in 1981, many smart card based password authentication schemes [5, 7, 12, 13, 16, 21, 22, 24, 25, 29, 32, 36, 40, 41, 42, 45, 46] came into existence. These schemes are aimed for different security goals and properties. As depicted in [1, 2, 3, 4, 8, 10, 11, 20, 27, 28, 31, 34, 35, 37, 38, 39, 44], many of the recently proposed schemes were broken shortly after they were first proposed, and few of them were further improved [6, 9, 15, 18, 19, 30, 47].

In 2006 Liao *et al.* proposed a new scheme [23] and claimed that their scheme satisfies a set of security requirements. In 2006, Yoon-Yoo [44] pointed out that Liao *et al.*'s scheme has drawbacks of masquerading attack by using replay attack, password guessing attack using lost smart card, insider attack and weakness in password changing. Later in 2008, Xiang *et al.* [38] found that Liao *et al.*'s scheme does not satisfy some of its claimed security requirements and showed the same attacks on it as were shown by Yoon-Yoo except the insider attack. Xiang *et al.* demonstrated Yoon-Yoo's masquerading server attack by using replay attack as impersonating the server by replay attack, Yoon-Yoo's password guessing attack using lost smart card as offline password guessing attack and Yoon-Yoo's weakness in password changing as denial of service attack on password changing. Consequently Liao *et al.*'s scheme is insecure for practical application. However, neither Yoon-Yoo nor Xiang *et al.* suggested any remedy to the pointed out attacks. This paper proposes an improvement to Liao *et al.*'s scheme [23], in terms of a new scheme. Our scheme is free from all weaknesses pointed above on Liao *et al.*'s scheme and yet carry forwards all advantages of the original scheme, like no database of entries is maintained at AS, users can freely choose and change their password at will etc. Our scheme achieves the following major security features:

- (User Authentication) AS is able to authenticate U and have surety that he is communicating with the registered user.
- (Server Authentication) The user is able to authenticate AS and have surety that he is communicating with the server to which he is registered.
- (Server Unknown of password) AS has no information of the password of a registered user.
- (Freedom of Password Choose and Change) The user is free to choose and change his password without interacting with AS.
- (Secure Generation of Session Key) U and AS agree on a secure session key to achieve the confidentiality of messages.

Rest of the paper is organized as follows: Section 2 is about the notations used throughout this paper. Section 3 reviews Liao *et al.*'s scheme and shows the attacks pointed out on it by Yoon-Yoo and Xiang *et al.*. Section 4 presents the proposed scheme. Section 5 is about the security analysis and achievements of the proposed scheme. Section 6 is about comparison of our scheme with other smart card based schemes. We conclude the article in Section 7.

2 Notations

The notations and descriptions used throughout this paper are summarized as follows:

- U: the user.
- ID: the identity of U.
- PW: the password of U.
- T_U, T_{U1}, T_{U2}: current timestamps of U.
- R: random number generated by U.
- SC: the smart card of U.
- AS: the authentication server.
- ADB: account database maintained by AS.
- S: random number chosen by AS.
- x: the secret key of AS.
- $K(.)$: a secret one-way function of AS.
- p: a large prime number selected by AS.
- g: a primitive element in $GF(p)$
- T_{AS}, T_{AS1}, T_{AS2}: current timestamps of AS.
- ΔT: the maximum time interval for transmission delay.
- U_A: the attacker.
- T_A: current timestamp of UA.
- $\oplus$: bitwise XOR operation.
- $h(.)$: a cryptographic hash function.
- $\Rightarrow$: a secure channel.
- $\rightarrow$: a common channel.
- $\|$: the string concatenation.
- S_{key}: the session key

3 Review of Liao *et al.*'s Scheme and its Security Analysis

3.1 Review of Liao *et al.*'s Scheme

3.1.1 Registration Phase

In this phase, everyone who wants to register at AS should obtain a SC.

1) U freely chooses his ID and PW, and calculates $h(PW)$.
2) U$\Rightarrow$ AS: Registration request $\{ID,\ h(PW)\}$.
3) AS calculates $B = g^{h(x\|ID)+h(PW)} \bmod p$.
4) $AS\Rightarrow$ U: SC containing $\{ID$, B, p, g$\}$.

3.1.2 Login Phase

When U wants to login AS, he should first insert his SC to the terminal, and keys in his ID and PW. Then, AS and SC will perform the following login steps.

1) $U \rightarrow AS$: $\{ID\}$.

2) AS generates a random number S, and calculates $B^{**} = g^{h(x||ID)S} \bmod p$.

3) $AS \rightarrow U$: $\{h(B^{**}), S\}$.

4) U calculates $B^{*} = (Bg^{-h(PW)})^{S} \bmod p$ and checks whether $h(B^{**}) = h(B^{*})$. If so, the identity of AS is authenticated. U then calculates $V = h(T_U||B^{**})$

5) U AS: Login request $\{ID$, V, TU$\}$ to AS.

3.1.3 Authentication Phase

This phase is executed by AS to determine whether U should be allowed to login or not. AS executes following steps to verify the legitimacy of U.

1) Checks the correctness of the format of ID. If it is invalid, the login request is rejected.

2) Generates the time stamp T_{AS} upon receiving U's login request. If $T_{AS} - T_U > \Delta T$, the login request is rejected, otherwise.

3) Computes $V^{*} = h(TU||B^{**})$, and then checks whether V is equal to V^{*}. The login request is accepted only if they are identical.

3.1.4 Password Change Phase

This phase is invoked if U wants to change his password from PW to PW^{*}.

1) U selects a new password PW^{*}.

2) U computes Y=$g^{h(PW*)}$mod p, and $Z = Bg^{Vh(PW)}$ mod p, and $\beta = YZ$, where PW is the old password and B is the variable stored in the SC.

3) Assigns $B = \beta$ in the smart card.

3.1.5 Extending to Key Agreement

The scheme can be extended to support Diffie-Hellman key agreement protocol if some minor modifications are made in the login phase and the authentication phase. In the login phase after receiving U's ID, AS selects a random number m, and calculates $M = g^{m} \bmod p$ and $h(B^{**}||M)$. Then AS sends $\{h(B^{**}||M), S, M\}$ to U. Upon receiving the message, U verifies whether $h(B^{*}||M)$ = $h(B^{**}||M)$. If the equality holds, AS is authenticated. U then selects a random number n and calculates $N = g^{n} \bmod p$ and $V = h(T_U||B^{*}||N)$. Then U sends $\{ID, V, N, T_U\}$ to AS. In the authentication phase, AS needs to calculate $V^{*} = h(T_U||B^{**}||N)$ and then checks whether V^{*} equals to V. If so, AS accepts the login request; otherwise rejects it.

Other procedures and calculations not mentioned here are same as in the previous description. After successful authentication, AS and U can share the common secret session key $S_{key} = M^{n} \bmod p = N^{m} \bmod p = g^{mn} \bmod p$ to perform encryption/decryption using traditional symmetric cryptosystems.

3.2 Cryptanalysis of Liao *et al.*'s Scheme by Yoon-Yoo and Xiang *et al.*

In Liao *et al.*'s scheme [23], the verification table is eliminated in the server. This elimination not only enhances the security of the scheme, but also alleviates the over head of computation and storage for the server. Users can freely choose their identity and password. Whats more, they can easily change the password without the participation of AS. However, some security loopholes pointed out by Yoon-Yoo [44] and Xiang *et al.* [38] are described as follows.

3.2.1 Yoon-Yoo's Masquerading Server Attack by Using Replay Attack/Xiang *et al.*'s Impersonating the Server by Replay Attack

In Liao *et al.*'s scheme, U's time stamp is utilized to resist replay attack for impersonating a legal user. However, the message sent from AS is time independent. Suppose U_A intercepts the message sent by AS, i.e. $\{h(B^{**}), S\}$, in the second step of login phase during U's authentication procedure. Next time, when U starts another new service request by sending ID to AS, then U_A intercepts it and sends the previously intercepted $\{h(B^{**}), S\}$ back to him. As $h(B^{**})$ contains the valid secret key x of AS and U's ID, so U cannot distinguish the replayed S. As a result, U_A will pass the authentication and is considered as a legal AS in step-4 of login phase.

According to Xiang *et al.*, a limitation of this method is that U_A can only impersonate himself as legal AS towards the user whose ID is contained in the intercepted $h(B^{**})$. For other user whose identity is ID^{*}, this attack will not work as $h(B^{*}) = g^{h(x||ID^{*})S} \neq h(B^{**}) = g^{h(x||ID)S}$. Even so, a sophisticated attacker U_A can intercept and store a list of triples $\{ID, h(B^{**}), S\}$. After intercepting a new service request containing ID, he finds an entry of record by the value of ID. If it is found, he then replays the corresponding $h(B^{**})$ and S to that user. Otherwise, he waits for the real legal AS's response, intercepts it, and adds the new triple to the list.

When key agreement is considered, U_A can still replay the intercepted $\{h(B^{**}||M), S, M\}$ to U and pass the authentication. However, Xiang *et al.* admitted that as U_A does not know the exponent m to generate M ($M = g^{m} \bmod p$) solving which is a discrete logarithm problem that is believed to be intractable. Therefore U_A cannot figure out the shared session key S_{key} during the following communication.

3.2.2 Yoon-Yoo's Password Guessing Attack Using Lost Smart Card/Xiang *et al.*'s Offline Password Guessing Attack

Suppose that U's smart card is lost, U_A can read all the data, including ID, B, p, and g, from SC via physically access to the storage medium. He then starts a service request by sending ID to AS. On receiving the request, AS will send him $\{h(B^{**}), S\}$. If AS is legal, U_A can perform the offline password guessing attack once he receives these data. For the legal AS, $B^{**} = B^*$, so $h(B^*) = h(B^{**}) = h((Bg^{-h(PW)})^S \bmod p)$. Since B, R, p, $h(.)$ and g are all known to U_A, he can guess the value of PW and verify his guess by the equation $h((Bg^{-h(PW)})^S \bmod p) = h(B^{**})$.

This attack is still effective when key agreement is added. In this case, upon receiving U_A's login request, the legal AS will send him $\{h(B^{**}||M), S, M\}$. Now UA needs to guess the PW to meet the requirement of $h(B^{**}||M) = h(((Bg^{-h(PW)})^S \bmod p)||M)$.

3.2.3 Yoon-Yoo's Insider Attack

In the registration phase of Liao *et al.*'s scheme, U sends his PW to AS with hashed value $h(PW)$. Thus the insider of AS can perform an off-line guessing attack on $h(PW)$ to obtain PW of U. If successful, the insider of AS can use PW to impersonate users in logging into other servers employing normal password authentication methods.

In practice, it is likely that U uses same PW to access several servers for his convenience. Liao *et al.* claimed that the insider of AS cannot get $h(PW)$ to obtain PW using an off-line guessing attack because the password of hashing table is encrypted by the administrator of AS. However in Liao *et al.*'s scheme verification table or password table is not stored inside the server computer.

3.2.4 Yoon-Yoo's Weakness in Password Changing/Xiang *et al.*'s Denial-of-service Attack on Password Changing

Suppose U_A temporarily gets access to U's SC; he then inserts the SC in a terminal device and performs the following operations for password change. He randomly selects two different passwords PW^* and PW^{**} as the old and the new password, respectively. Then he sends a request for changing password, to the SC. The SC will then compute $Y = g^{h(PW^{**})} \bmod p$, $Z = Bg^{h(PW*)} \bmod p$, and

$$\begin{aligned}\beta &= YZ \\ &= g^h(PW^{**})Bg^{-h(PW^*)} \bmod p \\ &= g^{h(PW^{**})}g^{h(x||ID)+h(PW)}g^{-h(PW^*)} \bmod p \\ &= g^{h(x||ID)+h(PW)-h(PW^*)+h(PW^{**})},\end{aligned}$$

then it replaces B with β without any further checking. From then on, U can never pass the password authentication by AS. This is because in the login phase, U cannot verify the legal AS. Moreover, he cannot be verified by the AS in the last step of authentication phase.

4 The Proposed Scheme

We propose an efficient and simple mechanism to defeat all the flaws demonstrated by Yoon-Yoo and Xiang *et al.* in Liao *et al.*'s scheme. In addition to timestamps, our scheme makes use of one time usable random numbers at user and server side. Our proposed protocol consists of four phases: the registration phase, the login phase, the authentication phase and the password change phase. We illustrate the detailed processes in sequence along with Figure 1 depicting the entire protocol structure of the proposed scheme.

4.1 Registration Phase

In this phase, the user U initially registers with AS.

1) U chooses his ID and PW, generates a random number b and computes $h(b||PW)$.

2) $U \Rightarrow AS$: Registration request $\{ID, h(b||PW)\}$.

3) AS checks the specific format of ID. If not, then asks U re-do from first step.

4) Otherwise AS computes $A_1 = h(ID)^{h(b||PW)} \bmod p$, $A_2 = (A_1)^{K(x)} \bmod p$, $EA_2 = A_2 \oplus h(b||PW)$, $B = (h(ID))^x \bmod p$, $B_K = K(B)$ and $EB_K = B_K \oplus h(b||PW)$.

5) $AS \Rightarrow U$: SC containing $\{A_1, EA_2, EB_K, p, h(.)\}$.

4.2 Login Phase

When U wants to login AS, he inserts his SC into the smart card device and keys in the PIN (Personal Identification Number) [33] to activate SC. If the PIN is entered incorrectly multiple times, then SC request a PUK (Personal Unblocking Key) [33]. Then U keys in his identity ID^*, password PW^* and random number b^*; then SC performs the following:

1) Computes $A_1^* = h(ID^*)^{h(b*||PW*)} \bmod p$ and if $A_1^* = A_1$, then

2) Extracts $A_2 = EA_2 \oplus h(b||PW)$ and $B_K = EB_K \oplus h(b||PW)$.

3) Computes $A_3 = A_2 \oplus h(B_K, T_{U1})$, $C_1 = R \oplus h(B_K, T_{U1})$, $C_2 = (A_2, B_K)^R \bmod p$ and $C_3 = h(C_2||T_{U1})$, where R is a random number generated by the SC of U using a secure random number generator.

4) $U \rightarrow AS$: Login request $\{ID, A_3, C_1, C_3, T_{U1}\}$.

Note: If U fails to enter the correct triple $\{ID, PW, b\}$ at number of times more than a predefined limit then SC denies to work further and displays need for re-registration.

User		Server
Registration Phase: Selects ID & PW Generates b Computes h(b‖PW)	{ID, h(b‖PW)} →	Checks ID format $A_1 = h(ID)^{h(b\|\|PW)}$ modp, $A_2 = (A_1)^{K(x)}$ modp, $EA_2 = A_2 \oplus h(b\|\|PW)$, $B = (h(ID))^x$ modp, $B_K = K(B)$, and $EB_K = B_K \oplus h(b\|\|PW)$
	← {SC}	Stores A_1, EA_2, EB_K, p, and h(.) in SC
Login Phase: U: Generates b^* Inputs ID^*, PW^* & b^* SC: $A_1^* = h(ID^*)^{h(b^*\|\|PW^*)}$ modp Aborts if $A_1^* \neq A_1$ $A_2 \leftarrow EA_2 \oplus h(b\|\|PW)$ and $B_K \leftarrow EB_K \oplus h(b\|\|PW)$ $A_3 = A_2 \oplus h(B_K, T_{U1})$, $C_1 = R \oplus h(B_K, T_{U1})$, $C_2 = (A_2, B_K)^R$ modp, and $C_3 = h(C_2\|\|T_{U1})$	{ID, A_3, C_1, C_3, T_{U1}} →	***Verification Phase:*** Verify ID format, if so Verify freshness of T_{U1} $B_K = K(B) = K[((h(ID))^x$ modp] $A_2^* \leftarrow A_3 \oplus h(B_K, T_{U1})$ and $R^* \leftarrow C_1 \oplus h(B_K, T_{U1})$ $C_2^* = (A_2^*, B_K)^{R^*}$ modp and $C_3^* = h(C_2^*\|\|T_{U1})$ Aborts if $C_3^* \neq C_3$ $D_1 = S \oplus h(A_2, T_{AS2})$, $D_2 = (C_2)^S$ modp and $D_3 = h(D_2\|\|T_{AS2})$
Verify T_{AS2} $S^* \leftarrow D_1 \oplus h(A_2, T_{AS2})$ $D_2^* = (C_2)^{S^*}$ modp and $D_3^* = h(D_2^*\|\|T_{AS2})$ Aborts if $D_3^* \neq D_3$ $S_{key} = h(D_2\|\|A_2\|\|B_K\|\|R\|\|S\|\|T_{U1}\|\|T_{AS2})$	← {D_1, D_3, T_{AS2}}	$S_{key} = h(D_2\|\|A_2\|\|B_K\|\|R\|\|S\|\|T_{U1}\|\|T_{AS2})$
User		**Smart card**
Password change Phase: Generates b^* Inputs ID^*, PW^* & b^*	{ID^*, PW^*, b^*} →	$A_1^* = h(ID^*)^{h(b^*\|\|PW^*)}$ modp Aborts if $A_1^* \neq A_1$ $A_2 \leftarrow EA_2 \oplus h(b^*\|\|PW^*)$ and $B_K \leftarrow EB_K \oplus h(b^*\|\|PW^*)$ Asks U for new password
Selects new password PW^{**}	{PW^{**}} →	$A_1^{**} = h(ID^*)^{h(b^*\|\|PW^{**})}$, $A_2^{**} = (A_2^{(\text{inverse of } h(b^*\|\|PW^*))})^{(h(b^*\|\|PW^{**})}$ modp, $EA_2^{**} = A_2^{**} \oplus h(b^*\|\|PW^{**})$ and $EB_K^{**} = B_K \oplus h(b^*\|\|PW^{**})$ $A_1^{**} \leftarrow A_1$, $EA_2^{**} \leftarrow EA_2$ $EB_K^{**} \leftarrow EB_K$

Figure 1: Proposed scheme

4.3 Authentication Phase

In this phase U and AS authenticate each other. On receiving the login request AS performs the following steps:

1) Checks the specific format of ID. If incorrect then rejects the login request, otherwise;

2) Checks whether $T_{AS1}\&T_{U1} \leq \Delta T$. If so, then computes $B_K = K(B) = K[(h(ID))^x \bmod p]$.

3) Extracts $A_2^* = A_3 \oplus h(B_K, T_{U1})$ and $R^* = C_1 \oplus h(B_K, T_{U1})$.

4) Computes $C_2^* = (A_2^*, B_K)^{R*} \bmod p$ and $C_3^* = h(C_2^*||T_{U1})$. If $C_3^* \neq C_3$ then rejects the login request, otherwise;

5) Computes $D_1 = S \oplus h(A_2, T_{AS2})$, $D_2 = (C_2)^S \bmod p$ and $D_3 = h(D_2||T_{AS2})$, where S is a random number chosen by AS from Z_P^*.

6) AS →U: $\{D_1, D_3, T_{AS2}\}$ and computes $S_{key} = h(D_2||A_2||B_K||R||S||T_{U1}||T_{AS2})$ for further communications. On receiving $\{D_1$, D_3, $T_{AS2}\}$, SC performs as follows:

7) Checks whether $T_{AS2}\&T_{U2} \leq \Delta T$. If so, then extracts $S^* = D_1 \oplus h(A_2, T_{AS2})$.

8) Computes $D_2^* = (C_2)^{S*} \bmod p$ and $D_3^* = h(D_2^*||T_{AS2})$. If $D_3^* = D_3$, then the legality of AS gets verified. Then SC computes $S_{key} = h(D_2||A_2||BK||R||S||T_{U1}||T_{AS2})$ for further communications.

4.4 Password Change Phase

With this phase U can change his password whenever he wants.

1) U inserts his SC into the smart card device and then keys in his PIN to activate SC. If the PIN is entered incorrectly multiple times, then SC demands PUK (Personal Unblocking Key). Then U keys in his identity ID^*, password PW^*, and random number b^*; and requests SC to change the password.
2) Computes $A_1^* = h(ID^*)^{h(b^*||PW^*)} \bmod p$. If $A_1^* = A_1$, then U is allowed to enter the new password PW^{**}.
3) Extracts $A_2 = EA_2 \oplus h(b^*||PW^*)$ and $B_K = EB_K \oplus h(b^*||PW^*)$.
4) Computes $A_1^{**} = h(ID^*)^{h(b^*||PW^{**})}$, $A_2^{**} = (A_2^{(\text{inverse of } h(b^*||PW^*)))(h(b*||PW**)} \bmod p$, $EA_2^{**} = A_2^{**} \oplus h(b^*||PW^{**})$ and $EB_K^{**} = B_K \oplus h(b^*||PW^{**})$.
5) Replaces A_1, EA_2 and EB_K with A_1^{**}, EA_2^{**}, and BK^{**} respectively.

Note: With this phase U can also changes his random number b along with PW.

5 Security Analysis and Achievements of the Proposed Scheme

5.1 Security Analysis

5.1.1 Replay Attack

If U_A replays a previously intercepted login request $\{ID, A_3, C_1, C_3, T_{U1}\}$, it is detectable by checking the timestamp freshness. If U_A replays $\{ID, A_3, C_1, C_3, T_A\}$ then such replay will fail. It is because A_3, C_1 and C_3 contain timestamp T_{U1} and AS would perform all computations using T_A, consequently $C_3^* \neq C_3$. Moreover, C_3 includes one-time usable random number R. For similar reasons replay of the response message $\{D_1, D_3, T_{AS2}\}$ will not be a success As we have seen that every message traveling over insecure network has inbuilt contribution of one-time usable random numbers and current time stamps; thus replaying any message and getting success is not possible in the proposed scheme.

5.1.2 Masquerading Server by Using Replay Attack

Unlike Liao *et al.*'s scheme, in the proposed scheme the response message sent from AS to U is not independent of timestamp. The response message is $\{D_1 = S \oplus h(A_2, T_{AS2}),\ D_3 = h((C_2)^S \bmod p||T_{AS2}), T_{AS2}\}$ in which D_1 and D_3 both contain the timestamp T_{AS2}. Thus replaying it as $\{D_1, D_3, T_A\}$ with U_A's current timestamp will not work and replaying as $\{D_1, D_3, T_{AS2}\}$ will be rejected due to timestamp freshness test.

5.1.3 Stolen Verifier Attack

In our scheme AS maintains no "verification table/database of entries" so the scheme is free from stolen verifier attack.

5.1.4 User Impersonation Attack by Forging a Login Request

U_A cannot create a valid login request using an intercepted login request $\{ID, A_3, C_1, C_3, T_{U1}\}$ to pass the authentication phase. For this U_A must know either the correct value of B_K or must be able to compute B_K. To compute B_K, U_A must have secret key $x$and secret function $K(.)$ of AS. However ID is apparent from the intercepted login request but neither any single value nor any combination of values A_3, C_1 and C_3 is helpful in guessing x and $K(.)$. From $A_3 = A_2 \oplus h(B_K, T_{U1})$, U_A has to guess A_2 and B_K simultaneously; from $C_1 = R \oplus h(B_K, T_{U1})$, U_A has to guess R and B_K simultaneously; if he XORes A_3 and C_1, he obtains $A_3 \oplus C_1 = A_2 \oplus R$ and has to guess $A_2 \& R$ simultaneously. But it is not possible to guess two values simultaneously in real time polynomial. Another problem with this efforts is that for a given equation $p \oplus q = r$, the solution in terms of p and q is not unique i.e. many values of p and q can satisfy the equation. Moreover guessing two values simultaneously is not possible in real time polynomial.

5.1.5 Server Impersonation Attack by Forging a Response Message

Suppose U_A has intercepted the response message $\{D_1, D_3, T_{AS2}\}$. To create a valid response message U_A should have A_2 and C_2 (or A_2 and B_K). But from $\{D_1, D_3, T_{AS2}\}$, U_A can neither extract nor guess these values. From $D_1 = S \oplus h(A_2, T_{AS2})$, U_A needs to guess S and A_2 simultaneously in order to correctly guess A_2. But it is not possible to guess two values simultaneously in real time polynomial. For similar reasons guessing any value from D_3 is more complex as it consists of four unknown values: A_2, B_K, R and S. From $D_3 = h(D_2||T_{AS2})$, however it is nearly impossible to guess the complex value D_2, but even after guessing D_2, U_A cannot successfully forge a valid response message $D_3 = h(D_2||T_A)$ because D_2 contains the old timestamp T_{AS2}.

5.1.6 Offline Password Guessing Attack

Suppose U_A steals or picks the lost SC of U. Now U_A can know $A1$, EA_2, EB_K, p and $h(.)$ from the SC by monitoring the power consumption [14] or by analyzing the leaked information [26]. U_A cannot guess PW of U from these values because of the following reasons:

1) To guess PW of U from $A_1 = h(ID)^{h(b||PW)} \bmod p$, U_A needs to know three values ID, b and PW. However login request contains ID in plaintext form but U_A has no way to relate possessed SC with possessed intercepted login requests. Thus getting the

ID corresponding to the possessed SC is very difficult for U_A. For instance, let us assume that U_A manages in doing so; still the guessing of password is far-far away because PW is protected in A_1 under the discrete logarithm problem. Had A_1 been equal to $h(ID)^{h(PW)} \bmod p$, then knowing $ID/h(ID)$, U_A could guess PW of U in the following manner: U_A chooses a value PW^*, computes $h(ID)^{h(PW^*)} \bmod p$ and if it is equal to A_1 then his guess PW^* is the correct one. But A_1 being $h(ID)^{h(b||PW)} \bmod p$, it is not possible to simultaneously guess two values b and PW correctly in real time polynomial.

2) To make PW guessing possible using EA_2 or EB_K an attacker has to know about A_2 and B_K, and has to guess b and PW simultaneously. So this effort is also useless. If U_A possess SC, login request and response message corresponding to the same user; even then he will not be able to guess the PW of U.

5.1.7 Smart Card Loss/Stolen Attack

Any invalid user must know the correct PIN set by the legal U to activate the SC. He should have the knowledge of PUK too because if PIN is entered incorrectly multiple times then SC request for PUK. If anyhow he is successful in activating the SC then in the proposed scheme old PW is required to be accepted before being allowed to enter the new password. If he fails to enter the correct triple $\{ID, PW, b\}$ at number of times more than a predefined limit then SC denies to work further and displays need for re-registration. Thus U_A cannot succeed.

- In changing the password inside SC.
- In online guessing the password.
- In impersonating the user to login the system.

5.1.8 Man in the Middle Attack

As described earlier that U_A fails to mount user /server impersonation attacks in any way; by replaying the intercepted messages or by forging valid messages using them. Therefore, it is not possible for U_A to mount man in the middle attack on the proposed protocol.

5.1.9 Denial of Service (DoS) Attack/Weakness in Password Changing

This attack can cause permanent error on authentication by introducing unexpected data during the procedures of authentication.

- In the proposed scheme, no information is stored at AS, therefore mounting this attack by updating false verification information of a legal user for the next login at AS side is not possible.
- DoS attack can be mounted during the password changing phase since it usually refreshes the data stored. In our scheme password change phase does not involves interaction with the AS, so no message transmission over insecure networks and hence no insertion of unwanted data and no manipulation of existing data. Moreover if U_A temporarily gets access to the SC of U then as discussed in 5.1.7, it is not possible for U_A to manipulate the data stored inside the SC.

5.1.10 Parallel Session and Reflection Attack

U_A can't create a valid login request out of the intercepted communication between U and AS without exactly knowing A_2 and B_K. And using fake values of A_2 and B_K will cause failure of step-4 of the authentication phase. Besides, every value travelling over insecure network has inbuilt inclusion of either the current timestamp or one time usable random number or both. We have also seen above that the proposed scheme is resistant to replay and impersonation attacks. Thus the scheme is free from parallel session and reflection attack.

5.1.11 Server's Secret Key (x) Guessing Attack

Even a valid user who can extract $A_2 = (A_1)^{K(x)} \bmod p$ and $B_K = K(B) = K((h(ID))^x \bmod p)$ from his SC, cannot guess the secret key x of AS. AS's secret key x is very well protected under the discrete logarithm problem (DLP) and the secret function $K(.)$ of AS, so this attack is useless.

5.2 Achievements

5.2.1 Mutual Authentication

In the proposed protocol mutual authentication between U and AS is achieved by sending challenges C_3 and D_3. U's SC sends C_3 as challenge to AS to authenticate U. No one other than the legal user (together with his SC) can create a valid C_3. If U inserts the correct triple $\{ID, PW, b\}$, only then SC of U extracts the correct A_2 and B_K to construct C_3. Such C_3 will pass the authentication test of AS because only AS can compute B_K. Thus AS believes that he is communicating with the valid user. On the other end AS sends D_3 as challenge to U to authenticate itself to U. D_3 contains the fresh challenge C_3 and freshly generated (to be shared) one time usable random number S. Such D_3 will pass the authentication test of U. Thus U believes that he is communicating with the valid AS. Thus the proposed scheme achieves mutual authentication i.e. not only AS authenticates the valid user, but U can also authenticate the legal server.

5.2.2 Perfect Forward Secrecy

If the secret key x of AS is revealed accidentally even then in spite of possessing U's SC, U_A can neither behave like legal AS nor like a legal U. It is because of the involvement of the secret function $K(.)$ of AS in the entire scheme.

5.2.3 Secure and Easy Password Change

Password change phase involves no interaction with the AS hence no use of insecure network, it imparts security and ease of password change. Also due the requirement of PIN and PUK, no one but U can activate his SC.

5.2.4 Denial of Service (DoS) Attack

In the proposed scheme, no information is stored at AS. Therefore, mounting this attack by updating false verification information of a legal user for the next login at AS side is not possible.

DoS attack can be mounted during the password changing phase since it usually refreshes the data stored. In our scheme password change phase does not involves interaction with the AS, so no message transmission over insecure networks and hence no insertion of unwanted data and no manipulation of existing data. Moreover if U_A temporarily gets access to the SC of U then as discussed in 5.1.7, it is not possible for U_A to manipulate the data stored inside the SC.

5.2.5 Quick Wrong Password Detection Mechanism

In the proposed scheme, validity of password can be checked by comparing the calculated and stored value of A_1. If triple $\{ID, PW, b\}$ is entered incorrectly at number of times more than a predefined limit then SC denies to work further and displays need for re-registration. Quick wrong password detection mechanism at the smart card level saves both SC and AS from exhaustion of computational resources. In the absence of such mechanism an unauthorized user can make an attempt to: online password guessing, forged login to AS which may be a success or may be detected at some later stage (up to that stage both SC and AS would have gone through lot of calculations causing unnecessary exhaustion of their computational resources). Thus, in our scheme both smart card and server are free from unnecessary computational load.

5.2.6 Secure Generation of Session Key

It is apparent from the construction of session key $S_{key} = h(D_2||A_2||B_K||R||S||T_{U1}||T_{AS2})$ that only valid U and valid AS have the correct information necessary to generate a valid session key. S_{key} need not travel via insecure network; it is the mutual authentication which guarantees that both AS & U have generated the same S_{key}.

5.2.7 AS's Ignorance of U's Password

U's password is not submitted to AS in plaintext form, rather it is submitted as $h(b||PW)$, well protected under one-way hash function $h(.)$ along with the random number b (known only to U).

6 Comparison with other Schemes

To further examine the proposed scheme we compare it with other smart card based schemes. We compare Liao *et al.*'s scheme [23], Kim-Chung's scheme [13], Sood *et al.*s scheme [30], Yeh *et al.*'s [42] improved version of Hsiang-Shihs scheme [9] and our scheme regarding security features, achievements and performance. Comparison results are depicted through Table 1, Table 2 and Table 3 respectively. The computation cost of registration (C_1) is the total time of all operations executed in the registration phase. The computation cost, of SC in login phase and session key establishment (C_2) and of AS during authentication phase and session key establishment (C_3) is the time spent by U and AS during the process of login/authentication procedure. We consider the computation cost of secret function $K(.)$ of AS equivalent to that of one-way hash function. Undoubtedly, among the compared schemes, Kim-Chung's scheme [13], Sood *et al*'s scheme [30], Yeh *et al.*'s [42] improved version of Hsiang-Shih's scheme [9] are low at computational costs because they do not use modular exponential function. However security of these schemes is not very high as they are based only on one-way hash function. We have proposed scheme based on both one-way hash function and discrete logarithm problem. The discrete logarithm problem is still an open problem and is more secure than one-way hash function. Compared to the original scheme i.e. Liao *et al.*'s scheme, our scheme makes additional use of only $8h(.)$ and $10\oplus$. It is clear that without using complex symmetric key cryptosystem or time consuming public key cryptosystem, our scheme is more secure and achieves more features than other relevant studies. Our scheme achieves almost all features that are essentially required in implementing a practical and universal remote user authentication scheme using smart cards.

7 Concluding Remarks

In this paper, we propose a scheme which is an improvement to the Liao *et al.*'s scheme. The proposed scheme overcomes all the problems identified by Yoon-Yoo and Xiang *et al.* in Liao *et al.*'s scheme: offline password guessing attack, impersonating the server by replay attack, insider attack, denial of service attack and weakness in password changing. Besides, our scheme withstands replay attack, stolen verifier attack, forged login attack, smart card loss/stolen attack, man-in-the middle attack and parallel session/reflection attack. The proposed scheme provides mutual authentication, freedom to the users to choose and change their password at will, secure and easy password change phase without interacting with AS, quick wrong password detection mechanism within the SC and perfect forward secrecy. It also provides confidentiality to the communication by means of the secure session key generation. Security analysis proved that the improved scheme is more secure and prac-

Table 1: Comparison of security features

Schemes → **↓ Attacks**	**Liao *et al.*'s [23]**	**Kim-Chung's [13]**	**Sood *et al.*'s [30]**	**Yeh *et al.*'s [42]**	**Our Scheme**
Replay	Yes	No	No	No	No
Stolen verifier	No	No	Yes	No	No
User impersonation	No	Yes	Yes	Yes	No
Server impersonation	Yes	Yes	Yes	No	No
Offline PW guessing	Yes	Yes	No	No	No
SC loss/stolen	Yes	Yes	No	Yes	No
Man-in-the middle	No	Yes	Yes	No	No
Insider	Yes	Yes	Yes	Yes	No
Denial of service	Yes	Yes	No	No	No
Parallel session	No	No	No	No	No
Reflection	No	No	No	No	No
Guessing secret key x	Yes	Yes	Yes	Yes	No

Table 2: Comparison of achievements

Schemes → **↓ Attacks**	**Liao *et al.*'s [23]**	**Kim-Chung's [13]**	**Sood *et al.*'s [30]**	**Yeh *et al.*'s [42]**	**Our Scheme**
Mutual authentication	No	Yes	Yes	Yes	Yes
Perfect forward secrecy	No	No	No	No	Yes
Secure PW changing	No	Yes	Yes	No	Yes
Quick wrong PW detec.	No	Yes	Yes	No	Yes
Secure S_{key} generation	Yes	No	No	Yes	Yes
U freely chooses PW	Yes	Yes	Yes	Yes	Yes
U freely changes PW	Yes	Yes	Yes	Yes	Yes

tical. Hence our scheme provides a reliable and trustworthy remote user authentication system.

References

[1] R. Arshad and N. Ikram, "Cryptanalysis and improvement on remote user mutual authentication scheme with smart cards," *Eleventh International Conference on Advanced Communication Technology (ACT2009)*, pp. 1202-1206, Phoenix Park, 2009.

[2] C. K. Chan and L. M. Cheng, "Cryptanalysis of a remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 46 no. 4, pp. 992-993, 2000.

[3] C. C. Chang and K. F. Hwang, "Some forgery attacks on a remote user authentication scheme using smart cards," *Informatica*, vol. 14, no. 3, pp. 289-294, 2003.

[4] C. K. Chang and L. M. Cheng, "Cryptanalysis of a timestamp-based password authentication scheme," *Computers & Security*, vol. 21, no. 1, pp. 74-76, 2002.

[5] 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.

[6] H. R. Chung, W. C. Ku, and M. J. Tsaur, "Weaknesses and improvements of Wang *et al.*s remote user password authentication scheme for resource-limited environments," *Computer Standards & Interfaces*, vol. 31, no. 4, pp. 863-868, 2009.

[7] 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.

[8] D. He, J. Chen, and J. Hu, "Weaknesses of a remote user password authentication scheme using smart card," *International Journal of Network Security*, vol. 13, no. 1, pp. 58-60, 2011.

[9] H. C. Hsiang and W. K. Shih, "Weaknesses and improvements of the Yoon-Ryu-Yoo remote user authentication scheme using smart cards," *Computer Communications*, vol. 32, no. 2, pp. 649-652, 2009.

[10] M. S. Hwang and L. H. Li, "A new remote user authentication scheme using smart card," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 28-30, 2000.

[11] M. S. Hwang, "Cryptanalysis of remote login authentication scheme," *Computer Communications*, vol. 22, no. 8, pp.742-744, 1999.

Table 3: Comparison of computational cost and complexity

	Liao et al.'s [23]	**Kim-Chung's [13]**	**Sood *et al.*'s [30]**	**Yeh *et al.*'s [42]**	**Our Scheme**
C_1	$1E$+$2h(.)$	$4h(.)$+$7\oplus$	$4h(.)$+$6\oplus$	$4h(.)$+$5\oplus$	3E+$4h(.)$+$2\oplus$
C_2	$4E$+$3h(.)$	$4h(.)$+$6\oplus$	$7h(.)$+$5\oplus$	$4h(.)$+$6\oplus$	3E+$6h(.)$+$5\oplus$
C_3	$3E$+$3h(.)$	$4h(.)$+$9\oplus$	$5h(.)$+$3\oplus$	$5h(.)$+$6\oplus$	2E+$6h(.)$+$3\oplus$
Sum	$8E + 8h(.)$	$12h(.) + 22\oplus$	$16h(.) + 14\oplus$	$13h(.) + 17\oplus$	$8E + 16h(.) + 10\oplus$

[12] M. S. Hwang, C. C. Lee, and Y. L. Tang, "An improvement of SPLICE/AS in WIDE against guessing attack," *Informatica*, vol. 12 no. 2, pp. 297-302, 2001.

[13] S. K. Kim and M. G. Chung, "More secure remote user authentication scheme," *Computer Communications*, vol. 32, no. 6, pp. 1018-1021, 2009.

[14] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," *Proceedings of Advances in Cryptology (Crypto' 99)*, pp. 388-397, 1999.

[15] W. C. Ku and S. M. Chen, "Weakness and improvements of an efficient password based remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 204-207, 2004.

[16] M. Kumar, "A new secure remote user authentication scheme with smart cards," *International Journal of Network Security*, vol. 11, no. 2, pp. 88-93, 2010.

[17] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, vol. 24, no. 11, pp. 770-771, 1981.

[18] N. Y. Lee and Y. C. Chiu, "Improved remote authentication scheme with smart card," *Computer Standards & Interfaces*, vol. 27, no. 2, pp. 177-180, 2005.

[19] S. W. Lee, H. S. Kim, and K. Y. Yoo, "Improvement of Chien *et al.*'s remote user authentication scheme using smart cards," *Computer Standards & Interfaces*, vol. 27, no. 2, pp. 181-183, 2005.

[20] C. T. Li and Y. P. Chu, "Cryptanalysis of threshold password authentication against guessing attacks in ad hoc networks," *International Journal of Network Security*, vol. 8, no. 2, pp. 166-168, 2009.

[21] C. T. Li, "An enhanced remote user authentication scheme providing mutual authentication and key agreement with smart cards," *Fifth International Conference on Information Assurance, and Security (IAS2009)*, pp. 517-520, Xi'An, China, 2009.

[22] C. H. Liao, H. C. Chen, and C. T. Wang, "An exquisite mutual authentication scheme with key agreement using smart card," *Informatica*, vol. 33, no. 2, pp. 125-132, 2009.

[23] I. E. Liao, C. C. Lee, and M. S. Hwang, "A password authentication scheme over insecure networks," *Journal of Computer and System Sciences*, vol. 72, no. 4, pp. 727-740, 2006.

[24] S. Lin and Q. Xie, "A secure, and efficient mutual authentication protocol using hash function," *International Conference on Communications, and Mobile Computing (CMC 2009)*, pp.545-548, Yunnan, 2009.

[25] J. Y. Liu, A. M. Zhou, and M. X. Gao, "A new mutual authentication scheme based on nonce and smart cards," *Computer Communications*, vol. 31, no. 10, pp. 2205-2209, 2008.

[26] 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.

[27] O. Qingyu, H. Kai, and L. Guang, "Cryptanalysis, and improvement of a remote user authentication scheme," *Second International Conference on Intelligent Computation Technology and Automation (ICTA 2009)*, pp. 49-52, Changsha, China, 2009.

[28] M. Scott, "Cryptanalysis of an ID-based password authentication scheme using smart cards and fingerprints," *ACM SIGPOS Operating Systems Review*, vol. 38, no. 2 pp. 73-75, 2004.

[29] R. Song, "Advanced smart card based password authentication protocol," *Computer Standards & Interfaces*, vol. 32, no. 5-6, pp. 321-325, 2010.

[30] S. K. Sood, A. K. Sarjee, and K. Singh, "An improvement of Liao *et al.*'s authentication scheme using smart card," *IEEE 2nd International Advance Computing Conference (IACC 2010)*, pp. 240-245, Patiala, India, 2010.

[31] H. M. Sun and H. T. Yeh, "Further cryptanalysis of a password authentication scheme with smart cards," *IEICE Transactions on Communications*, vol. E86-B, no. 4, pp. 1412-1415, 2003.

[32] H. M. Sun, "An efficient remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 4, pp. 958-961, 2000.

[33] M. Timothy, Jurgensen, and Scott B. Guthery, *Smart cards: The Developer's Toolkit*, Prentice Hall PTR, 2002.

[34] C. S. Tsai, C. C. Lee, and M. S. Hwang, "Password authentication schemes: Current status and key issues," *International Journal of Network Security*, vol. 3, no. 2, pp. 101-115, 2006.

[35] B. Wang, J. H. Li, and Z. P. Tong, "Cryptanalysis of an enhanced timestamp-based password authentication scheme," *Computers & Security*, vol. 22, no. 7, pp. 643-645, 2003.

[36] R. C. Wang, W. S. Juang, and C. L. Lei, "Robust authentication and key agreement scheme preserving the privacy of secret key," *Computer Communications*, vol. 34, no. 3, pp. 274-280, 2010.

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

[38] T. Xiang, K. W. Wong, and X. Liao, "Cryptanalysis of a password authentication scheme over insecure networks," *Journal of Computer and system Sciences*, vol. 74, no. 5, pp. 657-661, 2008.

[39] C. C. Yang, H. W. Yang, and R. C. Wang, "Cryptanalysis of security enhancement for the timestamp-based password authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 578-579, 2004.

[40] C. Yang, Z. Jiang, and J. Yang, "Novel access control with authentication using smart cards," *Third International Joint Conference on Computational Science and Optimization (CSO2010)*, pp. 387-389, Huangshan, China, 2010.

[41] G. Yang, D. S. Wong, H. Wang, and X. Deng, "Two-factor mutual authentication based on smart cards and passwords," *Journal of Computer and System Sciences*, vol. 74, no. 7, pp. 1160-1172, 2008.

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

[43] Z. Yong, M. Jianfeng, and S. J. Moon, "An improvement on a three-party password-based key exchange protocol using weil pairing," *International Journal of Network Security*, vol. 11, no. 1, pp. 17-22, 2010.

[44] E. J. Yoon, and K. Y. Yoo, "Drawbacks of Liao *et al.*'s password authentication scheme," *International Conference on Next Generation Web Services Practices (NWeSP 2006)*, Seoul, Korea, 2006.

[45] E. J. Yoon, and K. Y. Yoo, "New authentication scheme based on a one-way hash function and Diffie-Hellman key Exchange," *4th international conference of Cryptology and Network Security (CANS 2005)*, LNCS 3810, pp. 147-160, Springer-Verlag, 2005.

[46] E. J. Yoon, E. K. Ryu, and K. Y. Yoo, "Efficient remote user authentication scheme based on generalized EIGamal signature scheme," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 568-670, 2004.

[47] 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.

Manoj Kumar is an Assistant Professor Department of Mathematics, Rashtriya Kishan Post Graduate College Shamli, Muzaffarnagar, Chaudhary Charan Singh University Meerut, India. He is a member of Indian Mathematical Society, Indian Society of Mathematics and Mathematical Science, Ramanujan Mathematical society, and Cryptography Research Society of India. He is working as a reviewer for various International peer review Journals: Journal of System and Software, Journal of Computer Security, International Journal of Network Security, The computer networks, computer and security, The Computer Journal and Applied Mathematics Journal of Chinese University etc. He is also working as a Technical Editor for some International peer review Journals- Asian Journal of Mathematics & Statistics, Asian Journal of Algebra, Trends in Applied Sciences Research, Journal of Applied Sciences. He is also the member of Technical Programme Committee of various national and international conferences. He has published his research works at national and international level. His current research interests include Cryptography and Applied Mathematics.

Mridul Kumar Gupta received his Ph.D in Mathematics in 1998 from Agra University, Agra(India). Currently he is a Professor & Head of Mathematics Department, Institute of Advanced Studies, C.C.S. University, Meerut, U.P., India. He is also Dean, Faculty of Engineering & Technology, C.C.S. University, Meerut (India). He worked as Recorder of the Section of Information and Communication Science & Technology for the years 2004-2005 and 2005-2006 of the Indian Science Congress Association. He has 25 years of teaching experience. He is life member of Indian Mathematical Society, Indian Science Congress Association, Ramanujan Mathematical Society and Bharata Ganita Parisad, Lucknow, India. His more than 35 research papers are published in International journals. His research interests include general topology, approximation theory and cryptography.

Saru Kumari received her M.Sc, M.Phil degrees in mathematics from Institute of Advanced Studies, C.C. S. University, Meerut (India) in 2000, 2005 respectively. Currently she is an Assistant Professor with Department of Mathematics, Agra College Agra, U.P., India. She is pursuing her Ph.D Degree in mathematics from Institute of Advanced Studies, Chaudhary Charan Singh University, Meerut U.P. India. She also served as an Assistant Professor, Department of Mathematics, Vijay Singh Pathik Govt. P.G. College, Kairana, Muzaffarnagar, U.P., India and Govt. Polytechnic, Firozabad, India. She is life member of Indian Mathematical Society and Indian Science Congress Association. Her research interests include cryptography, information security, and applied mathematics.