An Improved Remote User Authentication Scheme with Elliptic Curve Cryptography and Smart Card without using Bilinear Pairings

S.Ramesh [#1], Dr.V.Murali Bhaskaran [*2]

[1#] Assistant Professor, Paavai College of Engineering, Pachal, Namakkal, Tamil Nadu, India
[*2] Professor, Dhirajlal Gandhi College of Technology,Omalur, Salem, Tamil Nadu, India
[#1] raameshs@gmail.com, [*2] murali66@gmail.com

***Abstract*—Login to the remote server over unreliable insecure network demands secured password a secured password authentication with less computational cost. We have proposed a remote user authentication scheme based on ECC that establish strong authentication with key agreement. As the password verifier table is vulnerable to security attacks and the bilinear pairing occupies more computation time, the proposed scheme avoids the usage of both, but is improved with the utility of smart card and ECC. The function and performance efficiency of our scheme was analysed and proved to provide a strong mutual authentication between user and server when compared with the existing methods.**

Keyword-Cryptanalysis, Smart card, Authentication, Key agreement, ECC

I. INTRODUCTION

The rapid development of network technologies, various communication systems are evolved for the people's life is made more and more convenient. Legal users want to access the remote resources, remote services such as Internet Banking, online shopping, online voting, online game and Pay - TV at any time and any places provided by the remote servers use secure authentication mechanism [19]. In electronic transaction remote client authentication has become more promiscuous technique in public and insecure channel such as Internet. Password based remote authentication scheme uses password table which is maintained by the server. This scheme suffers from replay attack, offline password guessing attack, impersonation attack and it is difficult to protect and maintain the password table that incur more cost [5]. Remote user authentication is implemented by traditional public key cryptography in which computational of modular exponentiation is needed. Among many public key cryptographic technique Elliptic curve cryptography has significant advantages like smaller keys, faster computation [20]. Several remote schemes based on ECC have been proposed to reduce the computation cost while preserving security strength. Bilinear pairings derived from the Weil pairings or Tate pairings on elliptic curves have been used in cryptography to construct identity and password based cryptographic schemes. The relative cost of the Bilinear parings is approximately 20 times higher than that of the scalar multiplication over elliptic curve group [21].

To solve the above said issues and improve the system security many password authentication schemes were developed using smart cards. Most of the smart card based remote schemes still vulnerable to offline password guessing attack, smart card stolen attack, server spoofing attacks, parallel session attack, power analysis attacks [14-18]. A strong smart card based password authentication should satisfy the following security requirements [5] (i) Withstand to replay, eavesdropping and modification attacks. (ii) Resist to impersonation and server spoofing attacks. (iii) Resist to parallel session attack, password disclosure attack. It also satisfies the functional requirements (i) Allow users freely to change the password consult with the server. (ii) To prevent the time synchronization problem (iii) Provide mutual authentication and session key agreement. (iv) Provision of forward secrecy and user anonymity. The serious security problem in the password remote user authentication scheme that all registered user's sensitive password can be easily derived by the privileged insider of the server. The scheme will not use any exponential operation which incurs high computational cost. Hence a good remote scheme is required to provide low computational cost and less communication cost.

The rest of the paper is organized as follows. Section 2 shows the related work of the remote user authentication scheme. In Section 3 we review Li-et al.'s new password based remote user authentication scheme using ECC. Section 4 describes a cryptanalysis of Li-et al.'s scheme. The proposed remote user authentication scheme and the corresponding security, performance and functional analysis are discussed in sections 5, 6 and 7 respectively. Finally we conclude this article in section 8.

II. RELATED WORK

Lamport [1] proposed a remote user authentication scheme using easily remembered password in 1981 that mutually authenticates the client and server successfully. In 1991 Chan and Wu and Chang [11] suggested to use smart card in remote user authentication scheme. Yang and Shieh [10] developed time stamp based two password authentication schemes in 1999. Hwang and Li [8] proposed a new remote authentication scheme that does not use password table at the server and withstand the replay attack. Later, Fan et al. [12] proposed in 2002 an enhancement scheme to improve the security of Yang and Shieh's password authentication scheme. In 2004, Kumar [13] proposed a new remote user authentication scheme does not support mutual authentication, session key generation phase for secure communication. Das et al. [7] proposed a remote user authentication protocol with smart card using bilinear pairings that allows user change their password without any assistance from the remote system. Feng et al. [9] discussed to improve the smart card based password authentication scheme with provable security but vulnerable to internal and impersonation attack. In 2010 Song [4] defined efficient and strong remote authentication protocol that involves modular exponentiations. In the same year, Sood et al. [23] found that Xu et al.'s [9] scheme is vulnerable to offline dictionary attack and forgery attack, and they also presented an enhanced scheme. Time based authentication protocol was proposed by Chaterjee et al. [18] in 2011 which produce multiple session keys. Islam and Biswas [3] proposed in 2011 ECC based secure remote user authentication scheme but it did not provide user anonymity. Islam Biswas scheme is vulnerable to offline password guessing, stolen verifier and insider attacks. In 2012 Awasthti [16] proposed a improved remote user authentication scheme with smart cards using bilinear pairings improve Feng et al.[9] scheme. Chun-Ta Li proposed in 2012 a slightly modified version of Islam Biswas scheme [3] so as to remedy the identified deficiencies and also provides user anonymity. Chao [14] proposed an improved password remote user authentication scheme with smart card that avoids well known attacks. In Li et al.[5] scheme use password verifier table and bilinear operations at the server side. However in this paper, we find that Li et.al scheme is still vulnerable to insider attack and consumes more computational cost. The spotted security flaw, we would like to propose an improved scheme that also inherits the facility of Li et al.'s password authentication scheme and resistance of insider's attack with the removal of modular computations involved in bilinear pairing operations. Li's scheme is inefficient in error password login and when the public key of the server is compromised, the adversary can obtain all the previous session keys between user U_i and the server S.

III. REVIEW OF LI ET AL.'S SCHEME

In this section we review Li's password based authentication scheme [5] and Li's scheme is consists of five phases: registration, password authentication, session key distribution, password change and user eviction phase. Fig.1. shows the entire protocol structure of Li et. al scheme. For the convenience of description, terminology and notations have been summarised as follows.

TABLE I
Notations used in the schemes

Symbol	Description
C	Client / User
S	Server
ID_i	User i's Identity
PW_i	C_i's Password
h(.)	One way hash function
$E_k()$	Symmetric Key Encryption
$D_k()$	Symmetric Key Decryption
X_s	Server S_i's secret key
X_c	User or ClientC_i's secret key
R_c	User or Client C_i's random number
R_s	Server S_i's random number
||	Concatenation operator
⊕	XOR operator
+/-	Elliptic curve point addition/subtraction
.	Scalar multiplication

3.1 Registration Phase

The user C wants to register in the remote server S and become a legal user initially registers at the server S.

Step 1. User C offer s ID_i, PW_i from the server S for registration as an authenticated user for further login and authentication process.

Step 2. The server S compute the password-verifier U_i as follows $U_i = r_i.PW_i.G$, Where G is the base point of the Elliptic curve chosen by the server S and r_i is a random number of the user.

Step 3. User C collect the server's public key U_s.

Step 4. Server stores each legal client's identity ID_i, password verifier U_i and a status bit in a write protected file is described as follows, where status bit indicate whether user is logged in to the server or not. When the user login to the server the status bit set to 1 otherwise status bit set to 0. Moreover S issues a smart card containing $G_s, U_s, H(.), E_k(.) / D_k(.)$. User does not need to remember r_i.

TABLE II
Verifier table

Identity	Password-verifier	Status-bit
ID_1	$U_1 = PW_1.r_1.G$	0/1
ID_2	$U_2 = PW_2.r_2.G$	0/1
ID_3	$U_3 = PW_3..r_3.G$	0/1
..	……	….

3.2 Authentication Phase

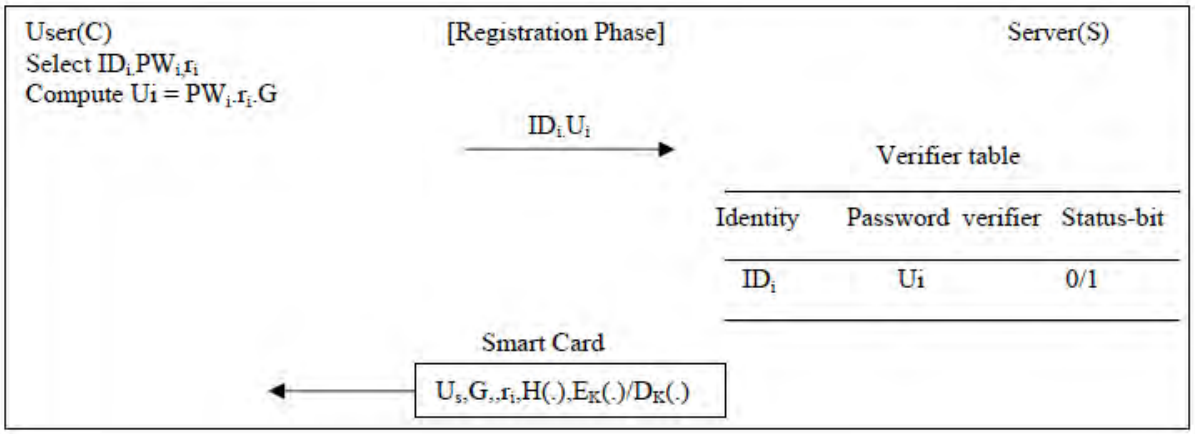
When the user C wants to access the server S, the user C has to insert the personal smart card into smart card reader and the server are able to authenticate each other through C's password-verifier and secret key K_x. The following steps are performed during the authentication phase.

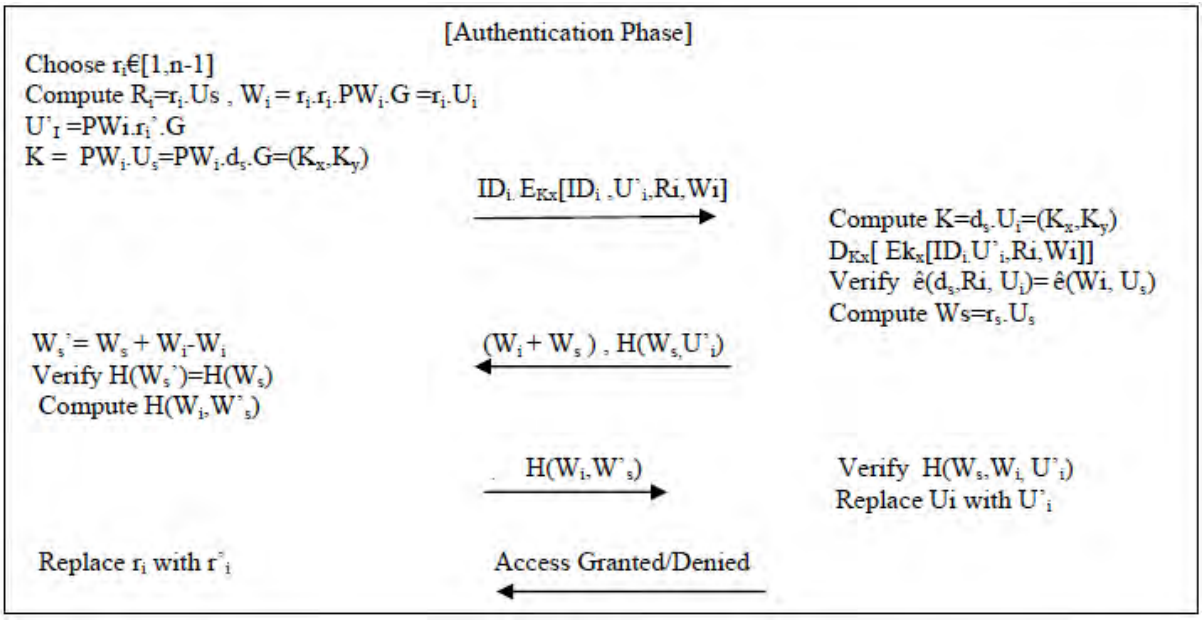
Step 1. User C enters his/her identity ID_i and password PW_i into the terminal. The client selects a random number r_i from $[1, n-1]$, computes $R_i = r_i.U_s$ and $W_i = r_i.r_i.PW_i.G$. User C calculates the encryption key K_x by $K = PW_i.U_s = PW_i.d_s.G = (K_x, K_y)$ where K_x is the symmetric key. Then encrypts $ID_i, R_i, W_i, U_i^{'}$ using K_x and sends to the server S. The smart card retrieves r_i generates a new random number $r_i^{'}$.

Step 2. The server S computes the decryption key K_x by calculating $K = d_s.U_i = PW_i.d_s.G = (K_x, K_y)$ and then decrypts $E_{k_x}(ID_i, R_i, W_i, U_i^{'})$ using K_x . The server S compares decrypted ID_i with received ID_i and retrieves R_i from verifier table. Next compare $\hat{e}(d_s, R_i, U_i)$ with $\hat{e}(W_i, U_s)$ where ê is bilinear paring operation. Bilinear pairing is used to assure correctness of the scheme.
$\hat{e}(d_s, R_i, U_i) = \hat{e}(d_s, r_i.G, r_i.G.PW_i) = \hat{e}(G, G)^{r_i.r_i.PW_i, d_s}$
$\hat{e}(W_s, U_s) = \hat{e}(r_i, r_i.PW_i.G, d_s.G) = \hat{e}(G, G)^{r_i.r_i.PW_i, d_s}$. If all the conditions are satisfied the server S selects a random number r_s and computes $W_s = r_s.U_s = r_s.d_s.G$. Then server sends $W_i + W_s$ and $H(W_s, U_i^{'})$ to the user C.

Step 3. User C retrieves W_s^i by subtracting W_i from $W_i + W_s$. Check whether hash value of retrieved W_s^i is equal to the receive hash value of W_s . If so user C performs the hash operation $H(W_i, W_s, U_i^{'})$ and C sends it to the server S.

Step 4. The server S uses its own copies of W_s and $(W_i, U_i^{'})$ received from C .S compute $H(W_i, W_s, U_i^{'})$ and compares it with the received $H(W_i, W_s, U_i^{'})$. If it holds server S accept login request and replace old password verifier U_i with new password verifier $U_i^{'}$. Otherwise user's login request denied. Finally if all conditions are satisfied C's smart card relace r_i with $r_i^{'}$.





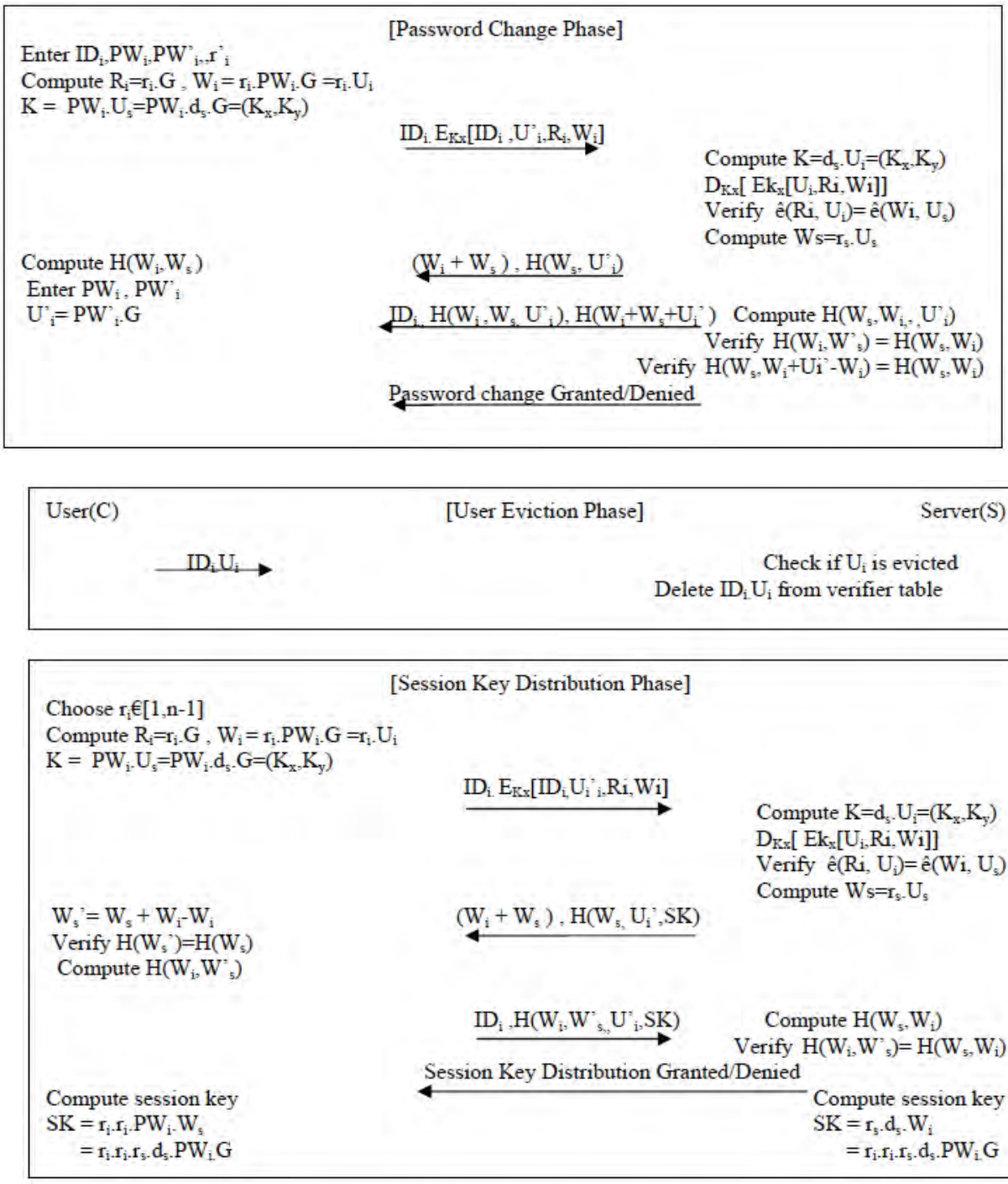


Fig. 1. Li et. al Scheme [5]

3.3 Password Change Phase

The user C wants to change his/her old password PW_i to a new password $PW_i^{'}$. The user C must notify the server to update the old password verifier $U_i = PW_i.G$ with new password verifier $U_i^{'} = PW_i^{'}.G$.

Step 1. User C enters his/her identity ID_i and password PW_i into the terminal. The client selects a random number r_i from $[1, n-1]$, computes $R_i = r_i.U_s$ and $W_i = r_i.r_i.PW_i.G$. User C calculates the encryption key K_x by $K = PW_i.U_s = PW_i.d_s.G = (K_x, K_y)$ where K_x is the symmetric key. Then encrypts $ID_i, R_i, W_i, U_i^{'}$ using K_x and sends to the server S.

Step 2. The server S computes the decryption key K_x by calculating $K = d_s.U_i = PW_i.d_s.G = (K_x, K_y)$ and then decrypts $E_{k_x}(ID_i, R_i, W_i, U_i^{'})$ using K_x. The server S compares decrypted ID_i with received ID_i and retrieves R_i from verifier table. Next compare $\hat{e}(\mathrm{R}_i, \mathrm{U}_i)$ with $\hat{e}(W_i, \mathrm{U}_s)$ where ê is bilinear paring operation. If all the conditions are satisfied the server S selects a random number r_s and computes $W_s = r_s.U_s = r_s.d_s.G$. Then server sends $W_i + W_s$ and $H(W_s, U_i^{'})$ to the user C.

Step 3. If the authenticated token $H(W_i, W_s, U_i^{'})$ and $H(W_s + W_i + U_i^{'})$ are valid, the server S subtracted W_i from $W_i + U_i^{'}$ to extract the new password verifier $U_i^{'}$. The server S computes the hash value of $(W_i, U_i^{'})$ and it compare it with the received $H(W_s^{'}, U_i^{'})$. If both the values are same server S replaces r_i with $r_i^{'}$.

3.4 Session key Distribution Phase

Step 1. User C enters his/her identity ID_i and password PW_i into the terminal. The client selects a random number r_i from $[1, n-1]$, computes $R_i = r_i.U_s$ and $W_i = r_i.r_i.PW_i.G$. User C calculates the encryption key K_x by $K = PW_i.U_s = PW_i.d_s.G = (K_x, K_y)$ where K_x is the symmetric key. Then encrypts $ID_i, R_i, W_i, U_i^{'}$ using K_x and sends to the server S.

Step 2. The server S computes the decryption key K_x by calculating $K = d_s.U_i = PW_i.d_s.G = (K_x, K_y)$ and then decrypts $E_{k_x}(ID_i, R_i, W_i, U_i^{'})$ using K_x. The server S compares decrypted ID_i with the received ID_i and retrieves R_i from verifier table. Next compare $\hat{e}(\mathrm{R}_i, \mathrm{U}_i)$ with $\hat{e}(W_i, \mathrm{U}_s)$ where ê is bilinear paring operation. If all the conditions are satisfied the server S selects a random number r_s and computes $W_s = r_s.U_s = r_s.d_s.G$. Then server sends $W_i + W_s$ and $H(W_s, U_i^{'}, SK)$ to the user C.

Step 3. User C retrieves $W_s^{'}$ by subtracting W_i from $W_s + W_i - W_i$. Check whether hash value of retrieved $W_s^{'}$ is equal to the receive hash value of W_s. If so user performs the hash operation $H(W_i, W_s, U_i^{'}, SK)$ and sends ID_i with it to the server S.

Step 4. The server S verifies the computed value of $H(W_i, W_s, U_i^{'}, SK)$ with the received value of $H(W_i, W_s, U_i^{'}, SK)$. If the hash values are same then two random numbers are chosen by the user C and server S from $[1, n-1]$. The user C computes the final session key as $SK = r_i.r_i.PW_i.W_s = r_i.r_i.r_s.PW_i.d_s.G$ and the server computes the session key as $SK = r_s.d_s.W_i = r_i.r_i.r_s.PW_i.d_s.G$.

3.5 User Eviction phase

In case of a client U_i is evicted by server S, U_i cannot use (ID_i, U_i) to login S because S can delete (ID_i, U_i) from its verifier table and ID_i cannot be found in the verifier table in Step 2 of the password authentication phase.

In this paper we research Li's scheme and show that this scheme cannot resist insiders attack and though eavesdropping the user's login request message in public network, the user U_i can be traced out. Furthermore a secure dynamic CID and password based remote user authentication scheme has been proposed in this paper using ECC. The proposed scheme is immune to various known types of attack and is more secure and practical for various clients.

IV. CRYPTANALYSIS OF LI ET AL.'S SCHEME

In this section, we will demonstrated that Li's scheme is vulnerable to insiders attack, the password guessing attack, stolen verifier attack and does not provide user anonymity. Li's scheme is inefficient in error password login and when the public key of the server is compromised, the adversary can obtain all the previous session keys between user U_i and the server S.

4.1 Offline password guessing attack

An adversary A eavesdropping the login request message in authentication , password change, session key distribution phases between user and server and launches an offline password guessing attack to get the users password PW_i. Generally user C chooses the low entropy and weak password for easy remembrance. In this attack adversary A known the public key of S and length of the password is small. It is not difficult to solve the elliptic curve discrete logarithm problem by a polynomial time algorithm. In Li et al. scheme the offline guessing attack may be done as follows.

Step 1. A intercepts the login request message $ID_i, E_{k_x}(ID_i, U_i^{'}, R_i, W_i)$ in step 1 of authentication phase.

Step 2. A computes $K^* = \left(K_x^{'}, K_y^{'}\right) = PW_i^*.U_s = PW_i^*.d_s.G$ as U_s is the public key of the server S.

Step 3. A decrypts the intercepted message $D_{k_x}[E_{k_x}(ID_i, U_i^{'}, R_i, W_i)]$ to obtain ID_i^*.

Step 4. A verify the guessing password PW_i^* by compare decrypted ID_i^* with the intercepted ID_i.

Step 5: Repeat Step 1, 2, 3 and 4 until the correct password PW_i is found.

After guessing the correct value of PW_i, A can compute the valid symmetric key $K = PW_i.U_s = (K_x, K_y)$. Attacker A can impersonate U_i to send a valid login request message to the server S. A can fabricate the values R_i, W_i from the guessed password PW_i. The attacker A can successfully masquerade as a legal user U_i to the server S. On the other hand the attacker my also impersonate the server S to user U_i successfully in a similar way.

4.2 Stolen verifier attack

An adversary A theft the password verifier from the verifier table in the database of the server S and create an offline guessing attack on it to obtain the user's password PW_i. S stores the password verifier $U_i = PW_i.G$ to the database and the adversary A can successfully find out C's password PW_i by perform the following steps

Step 1. A steals U_i from server S's database..

Step 2. A guesses a password PW_i^* and computes $U_i^{'} = PW_i.G$

Step 3. A compares $U_i^{'}$ with stolen U_i.

Step 4. Verify the correctness of PW_i^* by checking if the computed $U_i^{'}$ is equal to the obtained value U_i.

4.3 Insider attack

We show the insiders attack on Li's password authentication scheme. If a privileged-insider of S can find an opportunity to derive client U_i's real password PW_i, he/she may use U_A's password PW_A to impersonate U_A to login other servers. After finishing the registration phase, the privileged-insider knows U_A's password-verifier $U_A = PW_A.r_A.G$. In addition, during the password authentication phase, client U_i sends a login request $ID_i, E_{k_x}(ID_i, R_i, W_i, U_i)$ to S. Then the privileged-insider reveals (ID_i, R_i, W_i, U_i) by using its secret key. Finally, the privileged-insider can derive client U_A's real password PW_A in off-line manner by using the following three steps:

Step 1. A selects a guessed password PW_A^*.

Step 2. Compute. $PW_A^*.G$.

Step 3. Compare $\hat{e}(R_A, PW_A^*.G)$ to $\hat{e}(U_S, U_A)$.

A match in Step 3 above indicates the correct guess of client U_i's password. The privileged-insider verifies the equation $\hat{e}(R_A, PW_A^*.G)$ to $\hat{e}(U_s, U_A)$ holds or not as follows:

$$\hat{e}(R_A, PW_A^*.G) = \hat{e}(r_A.U_S, PW_A^*.G) = \hat{e}(r_A.d_S.G, PW_A^*.G) = \hat{e}(d_S.G, r_A.PW_A^*.G) = \hat{e}(U_S, U_A).$$

As a result, the privileged-insider succeeds to guess the low-entropy password PW_A and Li's password authentication scheme is vulnerable to insider attack.

4.4 Lack of user anonymity

User identity ID_i is transmitted in plain text form; any adversary may eaves drop the login message $ID_i, E_{k_x}(ID_i, R_i, W_i)$. Static ID is used in login request message of registration, authentication and session key generation phase. An adversary easily trace out the different login request messages belonging to the same user and try to derive some related to the user U_i.

V. PROPOSED SCHEME

In this scheme we have mentioned an improved smart card based password authentication scheme which does not have password verifier table and bilinear operations. The password verifier table is vulnerable to security attacks and the bilinear pairings occupies more computation time, the proposed scheme avoids the usage of both. The proposed scheme comprised of 6 phases namely initialization, registration, login, password authentication, session key distribution, password change phase. Fig 2. show the entire proposed structure of a proposed scheme. Now each of the phases is discussed below.

5.1 Initializing Phase

Step 1. Server S chooses an elliptic curve E over a finite field F_p.

Step 2. Server S select a base point G has a large order n over E.

Step 3. S chooses its secret key x and compute public key $y = x.G$.

Step 4. S selects the one way hash function $H(.)$.

Step 5. S publishes the parameters $E, G, n, H(.), E_k(.) / D_k(.)$.

5.2 Registration Phase

Step 1. When a user C wants to register and become a valid user he/she freely choose identity ID_i and password PW_i. Select random number r_c The server S compute the password-verifier U_i as follows $U_i = PW_i.G$, Where G is the base point of the Elliptic curve chosen by the server S.

Step 2. User selects a random number r_i and collects the servers public key U_s submits ID_i, U_i to the server S via a secure communication channel.

Step 3. Server computes $A_i = H(ID_i) \oplus U_i \oplus H(d_s)$, $B_i = A_i \oplus H(r_s) \oplus H(d_s)$, where d_s is the server's secret key kept in secret place.

Step 4. Server stores $A_i, B_i, G, U_s, H(.), E_k(.) / D_k(.)$ in the smart card and submits the smart card to the user C via a secure channel.

Step 5. User C receives the smart card; user enters r_i into the smart card. Finally the smart card contains parameters $A_i, B_i, r_i, G, U_s, H(.), E_k(.) / D_k(.)$.

5.3 Login Phase

If user C wants to access the server he/she insert the smart card into the terminal and enter the identity ID_i with password PW_i. The smart card performs the verification process if it holds accept the login request or otherwise rejects it.

Step 1: User C inputs the identity ID_i and password PW_i and computes the password verifier $U_i = PW_i.G$.

Step 2: Insert the smart card into the smart card reader, the smart card compute the dynamic identity CID_i and verify whether $CID_i \oplus U_i$ equals $A_i \oplus B_i$. If it holds smart card accept the valid identity, password and compute $W_i = r_i.PW_i.G = r_i.U_i$.

Step 3: User C calculates the encryption key K_x by $K = PW_i.U_s = PW_i.d_s.G = (K_x, K_y)$ where K_x is the symmetric key.

Step 4: The smart card sends the login request message $CID_i, A_i, E_{K_x}(B_i, W_i)$ to the server S over a public channel.

5.4 Authentication Phase

Step 1. Server S received the login request message $CID_i, A_i, E_{K_x}(B_i, W_i)$ and computes the password verifier $U_i = A_i \oplus CID_i \oplus H(d_s)$. Server S calculates the symmetric key K_x by using the key $K = d_s.U_i = (K_x, K_y)$. Next server decrypts the encrypted message $E_{K_x}(B_i, W_i)$ by using the symmetric key K_x.

Step 2. Server S check the validity of $H(r_s) \oplus H(d_s) = A_i \oplus B_i$ if it equals server computes $W_s = r_s.U_s$. Then server sends $W_s + W_i$ and $H(W_s)$ to the user C.

Step 3. User C retrieves $W_s^{'}$ by subtracting W_i from $W_s + W_i$. Check whether hash value of retrieved $W_s^{'}$ is equal to the receive hash value of W_s. If so user C performs the hash operation $H(W_i, W_s)$ and C sends it to the server S.

Step 4. The server S uses its own copies of W_s and W_i and compares it with the received $H(W_i, W_s)$ to accept or denied the login request. If all of the conditions are meet out then the server S granted the user's login request, otherwise user's login request rejected.

5.5 Password Change Phase

The user C wants to change his/her old password PW_i to a new password PW_i^{new}. The user C must notify the server to update the values of A_i and B_i in the smart card.

Step 1. User C inserts his/her smart card enters his/her into the smart card reader and then inputs the identity ID_i, old password PW_i and new password PW_i^{new} into the terminal. The client C computes the old password verifier $U_i = PW_i.G$ and new password verifier $U_i^{'} = PW_i^{new}.G$.

Step 2. User C sends the message $ID_i, U_i, U_i^{'}$ to the smart card. The smart card computes dynamic identity $CID_i = H(ID_i)$. Next smart card compares the computed value of $CID_i \oplus U_i$ with the stored values of $A_i \oplus B_i$. If the values are same smart card SC calculates the encryption key K_x by $K = PW_i.U_s = PW_i.d_s.G = (K_x, K_y)$ where K_x is the symmetric key. Then encrypts $B_i, U_i^{'}$ using K_x and sends the message $CID_i, A_i, E_{K_x}(B_i, U_i^{'})$ to the server S.

Step 3. The server S receive the computes the decryption key K_x by calculating $K = d_s.U_i = PW_i.d_s.G = (K_x, K_y)$ and then decrypts $E_{K_x}(B_i, U_i^{'})$ using K_x. The server S compares $H(r_s) \oplus H(d_s)$ with $A_i \oplus B_i$. If the values are same then the server S compute the new values $A_i^{'} = H(ID_i) \oplus U_i^{'} \oplus H(d_s)$ and $B_i^{'} = A_i^{'} \oplus H(r_s) \oplus H(d_s)$.

Step 4. Server S sends the encrypt the message $A_i^{'}, B_i^{'}$ using K_x to the smart card. The smart card replaces A_i with $A_i^{'}$ and B_i with $B_i^{'}$.

5.6 Session Key Distribution Phase

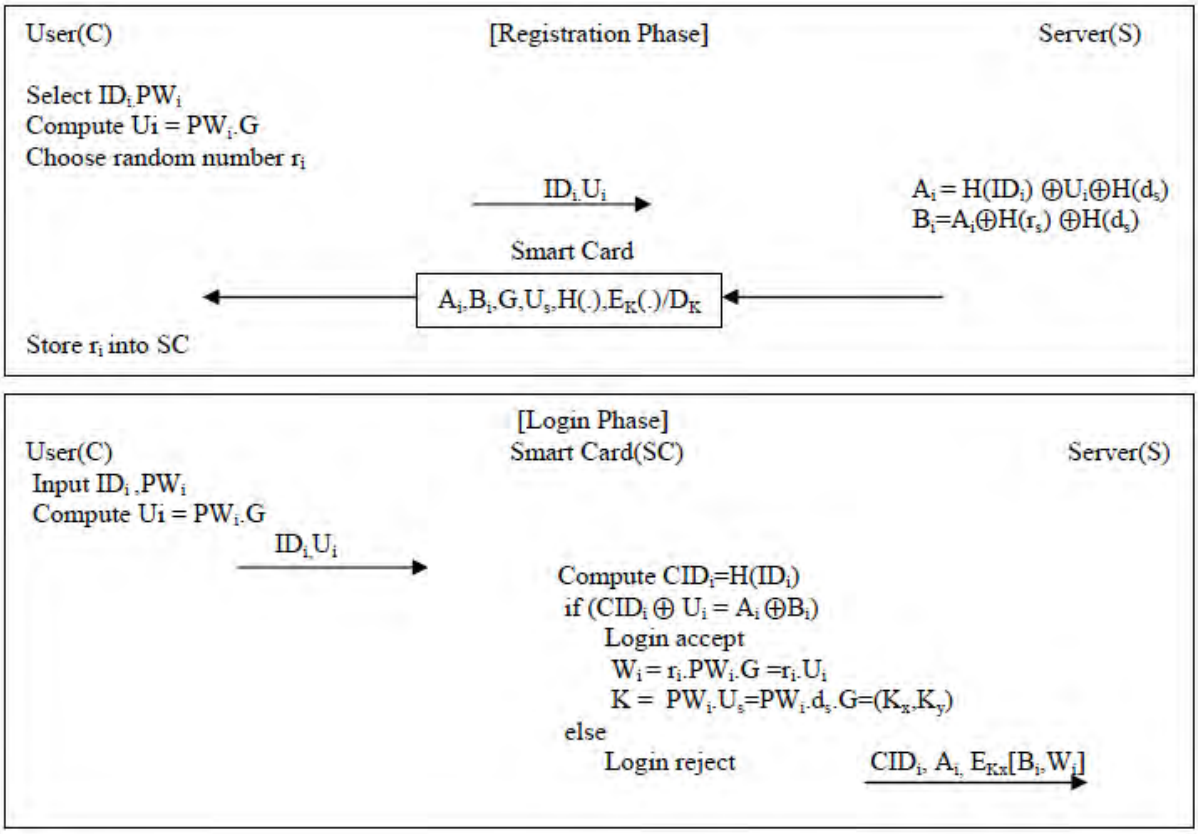
Step 1. User C enters his/her identity ID_i and password PW_i into the terminal. The client C computes the old password verifier $U_i = PW_i.G$. Insert the smart card into the smart card reader, the smart card compute the dynamic identity CID_i . and verify whether $CID_i \oplus U_i$ equals $A_i \oplus B_i$. If it holds smart card accept the valid identity, password and computes $W_i = r_i.PW_i.G = r_i.U_i$.

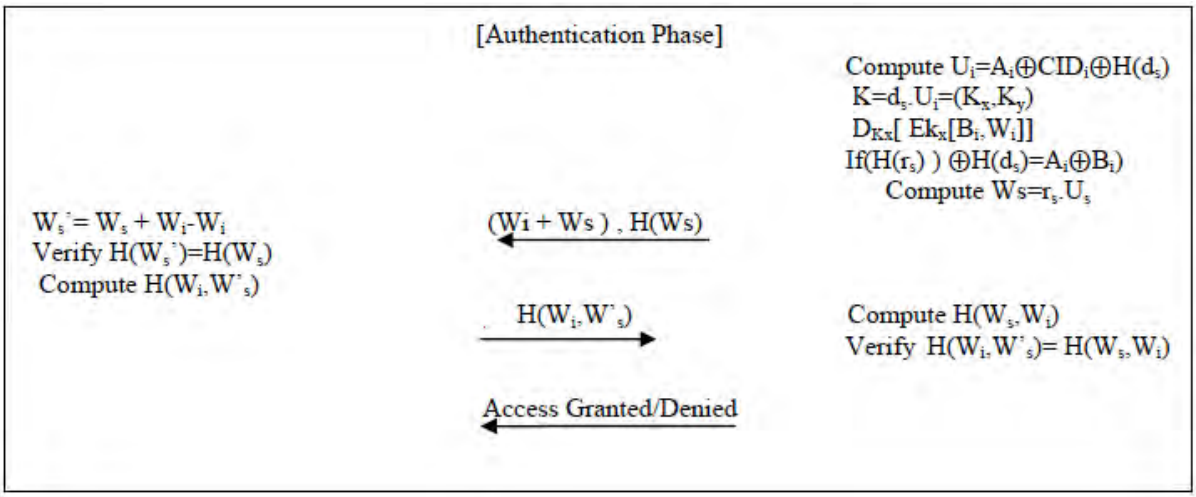
Step 2. User C calculates the encryption key K_x by $K = PW_i.U_s = PW_i.d_s.G = (K_x, K_y)$ where K_x is the symmetric key. Then encrypts B_i, W_i using K_x and sends the message $CID_i, A_i, E_{K_x}(B_i, W_i)$ to the server S.

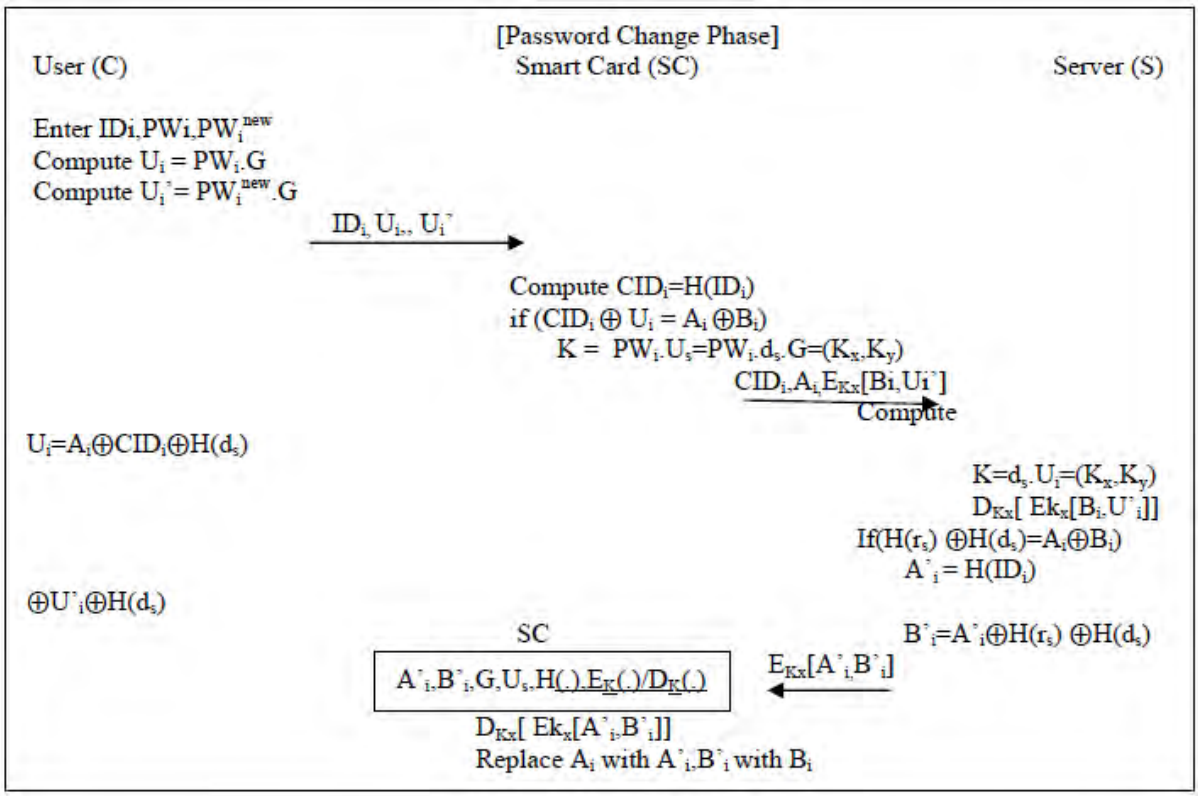
Step 3. The server S receives the message and computes the password verifier $U_i = A_i \oplus CID_i \oplus H(d_s)$. Next S computes the decryption key K_x by calculating $K = d_s.U_i = PW_i.d_s.G = (K_x, K_y)$ and then decrypts $E_{K_x}(B_i, W_i)$ using K_x. The server S compares $H(r_s) \oplus H(d_s)$ with $A_i \oplus B_i$. If it holds the server S selects a random number r_s and computes $W_s = r_s.U_s = r_s.d_s.G$. Then server sends $W_i + W_s$ and $H(W_s)$ to the user C.

Step 4. User C retrieves $W_s^{'}$ by subtracting W_i from $W_i + W_s$. Check whether hash value of retrieved $W_s^{'}$ is equal to the received hash value of W_s . If so user performs the hash operation $H(W_i, W_s)$ and C sends it to the server S.

Step 5. The server S verify the computed value of $H(W_i, W_s)$ with the received value of $H(W_i, W_s)$. If the hash values are same then two random numbers are chosen by the user C and server S from $[1, n-1]$ .The user C computes the final session key as $SK = r_i.PW_i.W_s = r_i.r_s.PW_i.d_s.G$ and the server computes $SK = r_s.d_s.W_i = r_i.r_s.PW_i.G$.




[Authentication Phase]
Compute $U_i=A_i \oplus CID_i \oplus H(d_s)$
$K=d_s.U_i=(K_x,K_y)$
$D_{Kx}[\ Ek_x[B_i,W_i]]$
If$(H(r_s)\ ) \oplus H(d_s)=A_i \oplus B_i)$
Compute $Ws=r_s.U_s$
$W_s' = W_s + W_i - W_i$
Verify $H(W_s')=H(W_s)$
Compute $H(W_i,W'_s)$
$(Wi + Ws\ )$, $H(Ws)$
$H(W_i,W'_s)$
Compute $H(W_s,W_i)$
Verify $H(W_i,W'_s)= H(W_s,W_i)$
Access Granted/Denied


[Password Change Phase]
User (C)
Smart Card (SC)
Server (S)
Enter IDi,PWi,PW_i^{new}
Compute $U_i = PW_i.G$
Compute $U_i' = PW_i^{new}.G$
ID_i, U_i, U_i'
Compute $CID_i=H(ID_i)$
if $(CID_i \oplus U_i = A_i \oplus B_i)$
$K = PW_i.U_s=PW_i.d_s.G=(K_x,K_y)$
$CID_i,A_i,E_{Kx}[Bi,Ui']$
Compute
$U_i=A_i \oplus CID_i \oplus H(d_s)$
$K=d_s.U_i=(K_x,K_y)$
$D_{Kx}[\ Ek_x[B_i,U'_i]]$
If$(H(r_s) \oplus H(d_s)=A_i \oplus B_i)$
$A'_i = H(ID_i)$
$\oplus U'_i \oplus H(d_s)$
$B'_i=A'_i \oplus H(r_s) \oplus H(d_s)$
SC
$A'_i,B'_i,G,U_s,H(.),E_K(.)/D_K(.)$
$E_{Kx}[A'_i,B'_i]$
$D_{Kx}[\ Ek_x[A'_i,B'_i]]$
Replace A_i with A'_i,B'_i with B_i

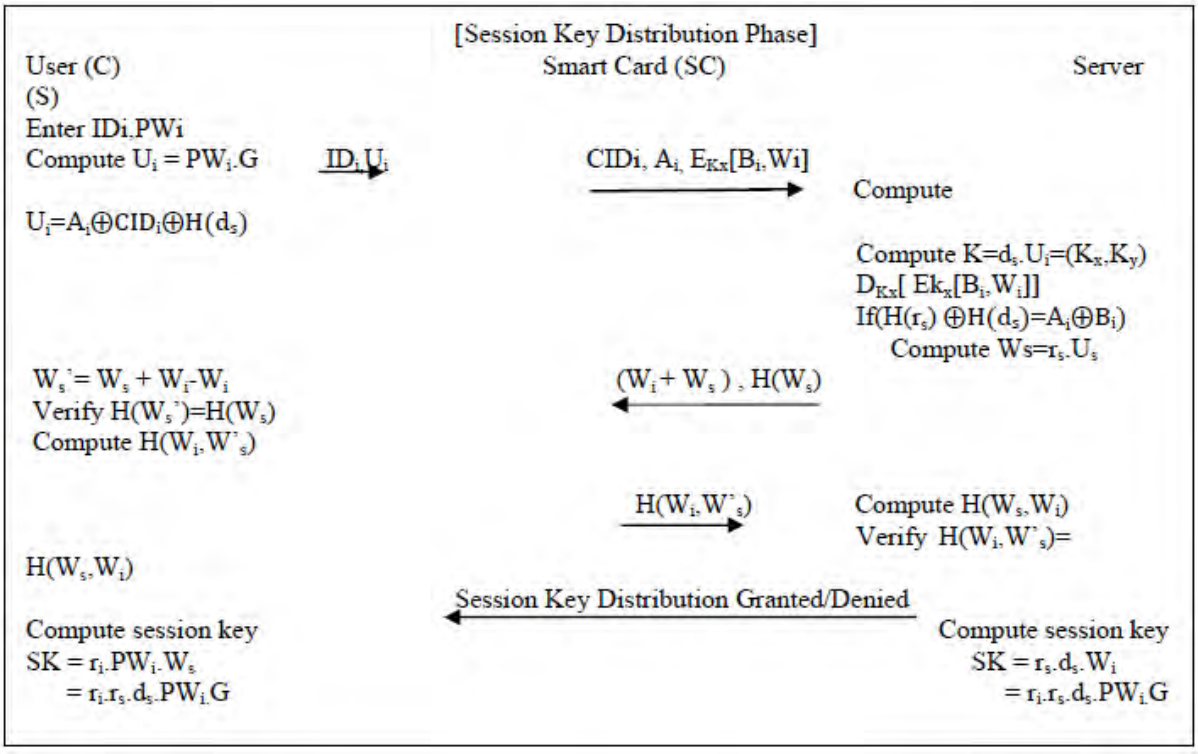


Fig. 2. Proposed Scheme

VI. SECURITY ANALYSIS

In this section we analyze the security of the proposed scheme and discuss the security features involved in it. The proposed scheme provides several security characteristics and resist against various known attacks. Table III describe the security characteristics involved in our scheme and other related schemes.

6.1 Resist stolen verifier attack

In our scheme the remote server does not store any password verifier table and the attacker cannot successfully launches an offline password guessing attack to obtain the client's original password PW_i. The client and server choose a long random numbers and it is hard to solve ECDLP by a polynomial time algorithm.

6.2 Resist insiders attack

In the proposed scheme an adversary cannot steal user's identity and password because the user sends the dynamic identity computed by hash function. It is computationally impossible to derive the password PW_i from the password verifier because of the difficulties of ECDLP and hence the adversary who steals U_i cannot generate the encryption key K_x without knowing the password PW_i and random number r_i. Hence the privileged insider of the server S cannot compute user identity CID_i and password PW_i.

6.3 Resist impersonation attack

If an attacker tries to modify the intercepted message $CID_i, A_i, E_{K_x}(B_i, W_i)$ or forge a valid login message to masquerade the user and login to the server to force impersonation as user. Hence it is difficult to the adversary to figure out W_i because (B_i, W_i) is encrypted by a symmetric secret key K_x only known to the user and the server. Moreover it is not possible to forge a valid login request $CID_i, A_i, E_{K_x}(B_i, W_i)$ without the knowledge of the secret key K_x and W_i. Hence there is infeasible of impersonation attack in the proposed scheme.

6.4 Resist server spoofing attack

An attacker cheats the server as valid user by forging the valid response message in the authentication phase. But in our scheme it is not feasible as the adversary cannot get the W_i by decrypting $E_{K_x}(B_i, W_i)$ without knowing symmetric secret key K_x. Hence the propose scheme can resist server spoofing attack.

6.5 Resist many logged in users attack

Any adversary tries to leak the login–id ID_i and password PW_i to other adversaries in login phase and password change phase. We provide a two additional security measure to restrict the adversaries not more than once to login to the server. Users who hold the smart card that contains r_i that integrated with A_i and B_i only login to the server.

6.6 Resist password disclosure attack

The proposed scheme should provide a mechanism to resist the password disclosure to any others. Immoral server may not able to find user password. Server S does not store any user's password in database. The client and server send and receive the message using the password verifier U_i. So it is difficult to figure out the user's password because U_i is protected by the random number r_i.

TABLE III
Security Comparison

Security Factors	Song et. al [2010]	Islam-Biswas et. al [2011]	Li et.al [2012]	Proposed Scheme
Stolen verifier attack	No	No	No	Yes
Insider attack	Yes	No	Yes	Yes
Impersonation attack	Yes	Yes	No	Yes
Password disclosure attack	No	No	No	Yes
Many logged in users attack	No	No	No	Yes
Server spoofing attack	Yes	Yes	Yes	Yes

VII. PERFORMANCE AND FUNCTIONAL ANALYSIS

In this section we compare our proposed scheme with other related schemes, our scheme achieves more functionality features that are required to implement the real time password authentication using smart cards is described in Table IV.

7.1 Mutual authentication

In our scheme an user C sends login request message to the server S, the server verify the login request by checking whether $H(r_s) \oplus H(d_s) = A_i \oplus B_i$ holds or not. The legal client who have the values of W_i, r_i and PW_i. On the other end client also check server is a legitimate one by checking the value of $W_i + W_s$ and $H(W_s)$. Because server possesses the private key d_s and password verifier U_i are used to compute the symmetric key K_x to decrypt the values of W_i and B_i. The server reply with legal response message $W_i + W_s$ and $H(W_s)$. The client checks the validity by compare $H(W_i + W_s - W_i)$ with $H(W_s)$. If it is valid client confirm that it communicated with the legal server.

7.2 User anonymity

Suppose the adversary intercepts the login message $CID_i, A_i, E_{K_x}(B_i, W_i)$ in the login phase of our scheme, he/she no way of guessing ID_i, because of the hardness of inverting of hash functions. Moreover due to the random number r_i the user cannot be traced out from the login request message. Therefore our scheme is able to preserve the user anonymity.

7.3 Session key agreement

Both client and server compute a common session key $SK = r_i.r_s.d_s.PW_i.G$ that establish a secure channel in between them. An attacker guess the random secrets r_i and r_s it is difficult to derive the session key SK without knowing the value of PW_i. Even if an adversary know the values of W_i and W_s, it is impossible to find the SK , because of the difficulties of elliptic curve computational Diffie-Hellman problem.

7.4 Time clock synchronization

Our proposed scheme does not associate with time for replay attack. Remote server and user are in different geographical locations there may be exists time clock synchronization problem. The time clock synchronization problem is eliminated by our scheme.

7.5 Forward secrecy

User's password PW_i and server's secret key d_s are compromised so that our proposed scheme is said to be forward secrecy. An attacker cannot obtain all past session keys. In the session key computation $SK = r_i.r_s.d_s.PW_i.G$ cannot trace the values of r_i and r_s which is hard to find due to the computational Diffie-Hellman problem.

7.6 No verification table

Our proposed scheme does not store the user's password in the password verifier table. The server maintains the table in a safer place. The adversary or inside user's of the server leak the confidential password to outsiders. Hence our scheme resist against password snooping attack.

TABLE IV
Functional Analysis

Functional Factors	Song et. al [2010]	Islam-Biswas et. al [2011]	Li et.al [2012]	Proposed Scheme
Mutual authentication	No	No	No	Yes
User anonymity	Yes	No	Yes	Yes
Session key agreement	Yes	Yes	No	Yes
No Time synchronization	No	Yes	Yes	Yes
Forward secrecy	No	No	No	Yes
No verification table	Yes	No	No	Yes
Without Bilinear Paring	Yes	No	No	Yes

In order to evaluate the performance of the proposed scheme, we compare it with other schemes. Table V gives a brief review of their performance, where the symmetric key encryption is denoted as T_S is similar to the hash operation T_H. The modulus exponentiation operation is T_{ME} have much higher computational complexity than T_E and T_H. T_{EM} denoted as elliptic curve multiplication and T_A denoted as elliptic curve addition and subtraction. Compared with other schemes our scheme's total computational cost is less and without using modular exponentiation involved in bilinear parings.

TABLE V
Performance Analysis

Phases	Song et. al [2010]		Islam-Biswas et. al [2011]		Li et.al [2012]		Proposed Scheme	
	User	Server	User	Server	User	Server	User	Server
Registration Phase		$T_{ME}+2T_H$	T_{EM}		$2T_{EM}$		T_{EM}	$3T_H$
Login & Authentication Phase	$T_S+3\ T_H$	$T_{ME}+2T_H+T_S$	$2T_H+T_S+5T_{EM}+T_A$	$2T_H+T_S+3T_{EM}+T_A+2\ T_{ME}$	$2T_H+T_S+8T_{EM}+T_A$	$2T_H+T_S+3T_{EM}+T_A+2\ T_{ME}$	$T_H+T_S+4T_{EM}+T_A$	$3T_H+T_S+2T_{EM}+T_A$
Session Key generation Phase	T_H	T_H	$2T_H+T_S+5T_{EM}+T_A$	$2T_H+T_S+3T_{EM}+T_A+2\ T_{ME}$	$2T_H+T_S+8T_{EM}+T_A$	$2T_H+T_S+3T_{EM}+T_A+2\ T_M$	$2T_H+T_S+3T_{EM}+T_A$	$3T_H+T_S+5T_{EM}+T_A$
Password Change Phase	$T_{ME}+2T_H$	T_H	$2T_H+T_S+5T_{EM}+3\ T_A$	$2T_H+T_S+3T_{EM}+T_A+2\ T_{ME}$	$2T_H+T_S+5T_{EM}+3\ T_A$	$2T_H+T_S+3T_{EM}+T_A+2\ T_{ME}$	$T_H+T_S+3T_{EM}$	$2T_H+T_S+T_{EM}$

VIII. CONCLUSION

In this paper an existing method of password based remote user authentication scheme's security weakness has been analysed. The password table maintenance is found to be vulnerable to various attacks. Hence this paper proposed an improved scheme which utilizes the advantages of smartcard usage and overcomes the usage of password verifier table and bilinear paring. The proposed scheme has been implemented and analysed with various key inputs and it proved that has much better security features and functional features when compared to other existing schemes.

REFERENCES

[1] L. Lamport, "Password authentication with insecure communication," Communication of the ACM, vol. 24, No 11, pp. 770-772, 1981.

[2] J. Qu, and Li-min Zou, "An improved dynamic ID-based remote user authentication with key agreement scheme," Elecrtical and Computer Engineering, Hindwai, vol. 13, No. 5, pp. 1-5, 2013.

[3] SK. Hafizul Islam, and G. P. Biswas, "Design of improved password authentication and update scheme based on elliptic curve cryptography," Mathematical and Computer Modeling , Elsevier., vol. 57, pp. 2703–2717, 2011.

[4] R. Song, "Advanced smear card based password authentication protocol," Computer Standards & Interfaces, Elsevier vol. 32, No. 4, pp. 321-325, 2010.

[5] C.T. Li, "A new password authentication and user anonymity scheme based on elliptic curve cryptography and smart card," IET Information Security, vol. 7, No. 1, pp. 3-10, 2012.

[6] P. Jiang, Q. Wen, W. Li, Z. Jin and H. Zhang, "Ana anonymous user authentication with key agreement scheme without pairings for multiserver architecture using SCPKs," The Scienctific world Journal, Hindwai, vol. 13, No. 1, pp. 1-8, 2013.

[7] M.L. Das, A. Saxena, V.P. Gulati and D.B.Pathak, "A novel remote client authentication protocol using bilinear pairings," Computer and Security, Elsevier, vol. 25, No. 3, pp. 184-189, 2006.

[8] M.S. Hwang and L.H. Li, "A new remote user authentication scheme using smart cards," IEEE Transactions on consumer electronics, vol. 46, No.1, pp. 28-30, 2000.

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

[10] W. H. Yang, and S.P. Sheih, "Password authentication schemes with smart cards," Computers & Security, Elsevier, vol. 18, No. 8, pp. 727-733, 1999.

[11] C. C.Chang, and T. C. Wu, "Remote password authentication with smart cards," IEE proceedings-E, vol. 138, No. 3, pp. 165-168, 1991.

[12] L. Fan, J. H. Li, and H. W. Zhu, "An enhancement of timestamp based password authentication scheme," Computers & Security, Elsevier, vol. 21, No. 7, pp. 137-144 , 2002.

[13] M.Kumar, "New remote user authentication scheme using smart cards," IEEE Transaction on Consumer Electronics, vol. 50, No. 2, pp. 497-500, 2004.

[14] J. Chao, "An improved remote password authentication scheme with smart card", Journal of Electronics, Springer, vol. 29, No. 6, pp. 550-555, 2012.

[15] Y. Lee, and D. Won, "Enhancing of a password based authentication scheme using smart cards," LNCS Springer, vol. 5871, No. 2, pp. 879-886, 2009.

[16] A. K. Awasthi, "An improved remote user authentication scheme with smart cards using bilinear pairings," International journal of applied mathematics and computations, vol. 4, No. 4, pp. 382-389, 2012.

[17] A. A. Yassin, H. J. A. Ibrahim, and D. Zou, "Encrypted remote user authentication scheme using smart card," LNCS Springer, vol. 7529, No. 4, pp. 314-323, 2012.

[18] K. Chatterjee, A. De, and D.Gupta, "Timestamp based authentication protocol for smart card using ECC," LNCS Springer, vol. 6987, No.6, pp. 368-375, 2011.

[19] H. Debiao, C.Jianhua, and H.Jin, "An ID based client authentication with key agreement protocol for mobile client server environment on ECC with provable security," Information Fusion, Elsevier, vol. 13, No. 3, pp. 223-230, 2012.

[20] H. L. Yeh, T. H. Chen and W.K. Shih, "Robust smart card secured authentication scheme on SIP using Elliptic Curve Cryptography," Computer Standards & Interfaces, Elsevier, In press, 2013.

[21] H. Debiao, J.Chen, and R.Zhang, "An efficient identity based blind signature scheme without using bilinear parings, " Computers and Electrical Engineering, Elsevier, vol. 37, No. 4, pp. 444-450 , 2011.

[22] S. K. Kim, and M. G. Chung, "More secure remote user authentication scheme," Computers Communications", Elsevier , vol. 32, No. 6, pp. 1018-1021, 2009.

[23] S. K. Sood , A. K. Sarje AK, K. Singh,." A secure dynamic identity based authentication protocol for multi-server architecture," Journal of Network and Computer Applications , Elsevier, vol. 34, No. 2, pp. 609–618 , 2011.

AUTHORS PROFILE

Dr. V. Murali Bhaskaran received the B.E. degree in Computer Science and Engineering from the Bharathidasan University, in 1990, M.S. degree in Computer Science from BITS, PILANI in 1995 and M,E. degree in 2000 from Bharathiyar University. He received the Ph.D. degree in Computer Science and Engineering from the Bharathiyar University. Currently, he is a professor in Computer Science and Engineering at Dhirajlal Gandhi College of Technology. His research interests include Network Security, Data Mining, Image Processing and Grid Computing.

S. Ramesh received the B.E. degree in Computer Science and Engineering from Madras University in 1992, M.S. degree in Software Systems from BITS, PILANI in 1997 and M.E. degree in 2006 from Anna University,Chennai. He is a research student of Anna University, Chennai. Currently, he is an Assistant Professor at Paavai College of Engineering in Computer Science department. His interests are in Cryptography and Network security.