Cryptanalysis of an Efficient Password Authentication Scheme

Prasanth Kumar Thandra, J. Rajan, and S. A. V. Satya Murty
(Corresponding author: Prasanth Kumar Thandra)

Computer Division, Indira Gandhi Centre for Atomic Research
Kalpakkam, Tamilnadu 603102, India
(Email: prasanth@igcar.gov.in)
(Received Aug. 22, 2013; revised and accepted July 15, 2014 & Nov. 3, 2014)

Abstract

Password authentication schemes are one of the most commonly used solution to protect resources in network environment from unauthorized access. Since, their first introduction in [9], many password authentications schemes have been proposed and analysed by crypto community. Contribution of the present paper is two-folded. At first it presents the cryptanalysis results of Ramasamy et al.'s RSA based password authentication scheme [11] and shows that it is vulnerable to privileged insider attack, password guessing attack and Impersonation attack. Secondly, modifications to the scheme were suggested to overcome the vulnerabilities. Formal security analysis of the proposed scheme was presented using BAN logic. In addition to being secure the modified scheme facilitate password update and mutual authentication. Efficiency comparison of the modified scheme is presented.

Keywords: Hash function, impersonation attack, mutual authentication, password guessing, RSA

1 Introduction

Remote authentication is a method to authenticate remote users over insecure communication channel. Password-based authentication schemes have been widely deployed to verify the legitimacy of remote users. In 1981, Lamport [9] proposed the first password-based authentication scheme using password tables to authenticate remote users over insecure network. Since then, many password authentication schemes [3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 15, 17] have been proposed and analyzed thoroughly by the cryptographic community. A password based remote user authentication scheme consists three components: remote user, remote server and an insecure channel to connect them. A typical smart card based remote user authentication scheme comprises three phases: registration phase, login phase and authentication phase. In the registration phase, a user sends a registration request and submits some necessary information to the server through a secure channel. The server uses the user's identity and password along with its long-term secret to generating user data. Some of this data is stored in a smart card, which then delivered to the user. In the login phase, a user uses the data in his smart card and his password to authenticate to the server.The smart card then uses the password and the values in the card to construct a login request and then sends it to the remote server. Successful authentication grants the user access rights to the protected resources. In the authentication phase, the server uses its long-term secret to check the validity of the login request. If mutual authentication is required, the server also uses its long-term secret to construct a message and sends it back to the user. The user then uses his password and the data in the smart card to check the validity of the message. In 1999, Yang et al. [17] proposed the first RSA-based remote user authentication scheme. Compared with Lamport's scheme [9], Yang et al.'s scheme needs no password tables or verification tables. Then Yang et al.'s is more practical than Lamport's scheme. However, many scholars have pointed that Yang et al.'s scheme was vulnerable to the forged login attacks [2, 3, 12]. Recently, quite a number of password authentication schemes with smart cards have been proposed [4, 8, 13]. Although, many remote user authentication schemes with smart cards have been proposed, none of them can solve all possible problems and withstand all possible attacks. In this line, in [11] Ramasamy et al., proposed an efficient password authentication scheme for smart cards using RSA algorithm [10]. This paper presents the cryptanalysis of the scheme in [11]. Password guessing attack and impersonation attack were demonstrated using proofs. To improve the security of Ramasamy et al.'s scheme, this paper proposes required changes to the scheme. The analysis shows that the new scheme not only overcomes the weaknesses in Ramasamy et al.'s scheme but also enables mutual authentication during login phase. The proposed scheme enables user to update his password without contacting

the server after registration phase. The rest of this paper is organized as follows. Section 2, reviews Ramasamy et al.'s scheme. Cryptanalysis of Ramasamy et al.'s scheme is presented in Section 3. In Section 4, the modified remote user authentication scheme is proposed. The security analysis is proposed in Section 5. Finally, Section 6 concludes the paper.

2 Review of Ramasamy et al.'s Scheme

Ramasamy et al.'s scheme has three phases, registration phase, login phase, and authentication phase. These phases are explained below.

2.1 Registration Phase

User U_i submits his identity ID_i and chosen password PW_i to Key Information Center. Key Information Center (S) issues a smart card to user U_i. Then S performs the registration steps:

1) Generates an RSA key pair, namely a private key d and public key (e, n). KIC publishes (e, n) and keeps d secret.

2) Determines an integer g, which is a primitive in both GF_p and GF_q.

3) Generates the smart card identifier CID_i of U_i and calculate the user's information as

$$W_i = ID_i \times CID_i \times d \,(\mathrm{mod}\, n).$$

4) Computes $V_i = g^{PW_i \times d \times T_r} \,(\mathrm{mod}\, n)$, here T_r is the time of registration of the user. This value is unique for every user, and maintained by the server.

5) Writes $(ID_i, CID_i, n, e, g, W_i, V_i)$ into the smart card of U_i, and send to the user securely.

2.2 Login Phase

When U_i wants to login to$S$, he inserts his smart card into a card reader and keys ID_i and PW_i. Then smart card reader will perform the following steps:

1) Generates a random number r and calculate X_i, Y_i as follows:

$$\begin{aligned} X_i &= g^{PW_i \times r} (\mathrm{mod}\, n) \\ Y_i &= W_i \times V_i^{r \times T} \,(\mathrm{mod}\, n). \end{aligned}$$

2) Sends the login request message (ID_i, CID_i, X_i, Y_i, n, e, g, T) to S.

2.3 Authentication Phase

Server receives the login request and performs the following steps:

1) Checks whether ID_i is a valid user identity and CID_i is a legal smart card identity, if not, then S rejects the login request.

2) Check, whether $T_c - \Delta T \leq T$, where T_c is the login request received time by server and ΔT is the legal time interval due to transmission delay, if not, then S rejects the login request.

3) Evaluate the following equation:

$$Y_i^e = ID_i^{CID_i} \times X_i^{T \times T_r} (\mathrm{mod}\, n) \quad (1)$$

where T is the login request time and T_r is the registration time of user.

4) If any one of the above result is negative, then login request is rejected. Otherwise, the login request is accepted. If the login request is rejected three times then automatically the user account is locked and he has to contact server to unlock the account.

3 Weaknesses of Ramasamy et al.'s Scheme

In [11] Ramasamy et al. claimed that their scheme is resistant to denial of service attack, parallel session attack, smart card lost attack, password guessing attack and impersonation attack. To evaluate the security of smart card based user authentication, we assume the capabilities that an adversary A may have as follows:

1) The adversary has total control over the communication channel between the users and the server in the login and authentication phases. That is, A may intercept, insert, delete, or modify any message in the channel.

2) A may (i) either steal a user's smart card and then extract the information from it, (ii) or obtain a user's password, (iii) but not both (i) and (ii).

In this section, we prove that Ramasamy et al.'s scheme is vulnerable to privileged insider attack, password guessing attacks and impersonation attack. A more detailed description of attacks is as follows.

3.1 Privileged Insider Attack

In a real environment, it is a common practice that many users use same passwords to access different applications or servers for their convenience of remembering long passwords and ease-of-use whenever required [4]. However, if the system manager or a privileged insider A of the server S knows the passwords of user U_i he may try to impersonate U_i by accessing other servers where U_i could be

a registered user. In the user registration phase of Ramasamy et al.'s scheme, sends his identity ID_i, the password PW_i to S directly. Then the privileged insider A could get U_i's password. Therefore, Ramasamy et al.'s scheme is vulnerable to the privileged insider attack.

3.2 Password Guessing Attack

In remote user authentication schemes that the user is allowed to choose his password, the client tends to choose a password that can be easily remembered for his convenience [4]. However, these easy-to-remember passwords are potentially vulnerable to password guessing attacks. In this type of attack an adversary tries to guess the client's password and then verifies his guess. Suppose an adversary A has stolen U_i's smart card and extracted the stored values $n; e; g; ID_i; V_i$ and then he got the value of T_r either from the server S or by observing the time during the registration process of the user. Then he guesses and checks for the user password as follows:

1) A computes,

$$\begin{aligned} L &= V_i^e = (g^{PW_i \times d \times T_r})^e (\mathrm{mod} n) \\ &= g^{PW_i \times T_r} (\mathrm{mod}\, n). \end{aligned}$$

2) A guesses a password PW_i' and computes N as $N = g^{T_r \times PW_i'} (\mathrm{mod} n)$.

3) A Compare if L and$N$ are equal or not. If equal the password guess is correct else repeat Steps 2) and 3).

From the above description, we know that with enough number of guesses, an adversary can get the password. Therefore, Ramasamy et al.'s scheme is vulnerable to the offline password guessing attack.

3.3 Impersonation Attack

A more serious attack on the scheme is impersonate attack in which attacker tries to masquerade as a valid user using some of his credentials. Suppose an adversary A has stolen U_i's smart card and extracted the stored values $n; e; g; ID_i; CID_i; W_i$. Then the attacker A can impersonate U_i to login in the server by performing the following procedure.

A computes and sends the login request message (ID_i, CID_i, X_i, Y_i, n, e, g, T) to S as follows:

1) $X_i = n - 1$, $Y_i = W_i$.

2) Chooses the current time when T is even. i.e, T (mod 2) $= 0$ or $T = 2t$.

Then, A sends the login request as $(ID_i, CID_i,\ n-1,\ W_i, n, e, g, T)$. Server receives the login request and checks, whether ID_i and CID_i are valid or not and whether $T_c - \Delta T \leq T$. If the verification in successful then S evaluate Equation (1) as follows:
Server computes L.H.S

$$\begin{aligned} Y_i^e &= W_i^e \\ &= ID_i^{CID_i \times d \times e} \\ &= ID_i^{CID_i} (\mathrm{mod}\, n). \end{aligned}$$

Then computes R.H.S

$$\begin{aligned} ID_i^{CID_i} \times X_i^{T \times T_r} &= ID_i^{CID_i} \times (n-1)^{T \times T_r} \\ &= ID_i^{CID_i} \times (n-1)^{2t \times T_r} \\ &= ID_i^{CID_i} \times 1^{t \times T_r} \\ &= ID_i^{CID_i} (\mathrm{mod}\, n). \end{aligned}$$

Finally, as L.H.S and R.H.S are equal server allows the attacker to login to the server. A similar message tuple that satisfies the above attack is (ID_i, CID_i, 1, W_i, n, e, g, T). Therefore, A could impersonate U_i successfully and Ramasamy et al.'s scheme is vulnerable to the impersonation attack.

3.4 Other Weaknesses of the Scheme

In addition to the weaknesses mentioned in Sections 3.1, 3.2 and 3.3 Ramasamy et al.'s scheme is lacking password update and mutual authentication. These features are desired by a good password authentication scheme. The password update feature, after registration of the user, enables a user to change his password at his will without contacting the server. This way he can update his password periodically. Mutual authentication enables a user to authenticate the server during the login process. This prevents a category of attacks called server masquerading attack. In this attack an attacker tries to act as legitimate server and allows user to login to his server and tries to get user information.

4 Securing Ramasamy et al.'s Scheme

In this section we are going to modify Ramasamy et al.'s scheme to make the scheme resist the attacks mentioned. The new modifications, also, enable the scheme to have password update and mutual authentication features. Here, our aim is not to propose a new efficient password authentication mechanism, instead to show how the scheme can be made secure. Our modification requires a 128/256 bit secure one-way hash function H.

4.1 Registration Phase

User U_i, with a user identity ID_i, chooses a password PW_i. He compute the hash of PW_i as $h_p = H(PW_i)$. Then submits the two tuple (U_i, h_p) to S securely. S performs the registration as follows and issues a smart card to user U_i.

1) Generates a RSA key pair, namely a private key$d$ and public key (e, n). S publishes (e, n) and keeps d secret.

2) Determines an integer g, which is primitive in both GF_p and GF_q where p and q are RSA primes.

3) Generates the smart card identifier $CID_i$of U_i as $CID_i = H(ID_i||d)$ and calculate the user's secret information as

$$w_i = CID_i \times g^{h_p} (\mathrm{mod}\, n).$$

4) Computes $V_i = g^{CID_i \times h_p \times T_r} (\mathrm{mod}\, n)$, here T_r is the time of registration of the user. The tuple (ID_i, T_r) is maintained by server.

5) Writes $(ID_i, w_i, n, e, g, V_i, T_r)$ into the smart card of U_i, and delivers it to the user securely.

4.2 Login Phase

When U_i wants to login to S, he inserts his smart card into a card reader and keys ID_i and PW_i. Then smart card reader will perform the following steps:

1) Computes hash of PW_i as $h_p = H(PW_i)$.

2) Unlocks $CID_i = w_i g^{-h_p} (\mathrm{mod}\, n)$.

3) Generates two k bit random numbers r_1, r_2 and calculates W_i, X_i and Y_i as $W_i = (CID_i||r_1)$, $X_i = g^{CID_i \times h_p \times r_2} (\mathrm{mod}\, n)$ and $Y_i = (W_i \times V_i^{r_2 \times T})^e (\mathrm{mod}\, n)$. Here the size of k is equivalent to bit security the RSA algorithm used by S provides.

4) Sends the login request message (ID_i, X_i, Y_i, T) to S and keeps r_1 and CID_i.

4.3 Authentication Phase

When Server receives a login request, it first checks whether ID_i is valid and $T_{sc} - \Delta T \leq T$ or not, where T_{sc} is the current time on S. If not, then S rejects the login request. Else, S does the following to evaluate the login request.

1) Computes $CID_i = H(ID_i||d)$.

2) Computes $L = X_i^{T \times T_r} = g^{CID_i \times h_p \times r_2 \times T \times T_r} (\mathrm{mod}\, n)$.

3) Computes $M = Y_i^d \times L^{-1} (\mathrm{mod} n)$.

4) Computes $R = M \,(\mathrm{mod}\, 2^{k+1})$ and $O = CID_i||R$.

5) Compares If $M = O$.

If the above result is negative, then login request is rejected. Otherwise, the login request is accepted. Correctness of the authentication is due to the following:

$$\begin{aligned} M &= Y_i^d \times L^{-1} (\mathrm{mod} n) \\ &= (W_i \times V_i^{r_2 \times T})^{ed} \times L^{-1} (\mathrm{mod} n) \\ &= W_i \times g^{CID_i \times h_p \times T_r \times r_2 \times T} \times L^{-1} (\mathrm{mod} n) \\ &= W_i (\mathrm{mod} n) \\ &= (CID_i||r_1). \\ O &= CID_i||R \\ &= CID_i||(M \,(\mathrm{mod}\, 2^{k+1})) \\ &= CID_i||((CID_i||r_1) \,(\mathrm{mod}\, 2^{k+1})) \\ &= (CID_i||r_1). \end{aligned}$$

Hence, M is equal to O.

To support the mutual authentication S computes $P = H(CID_i||R||T_m)$ where, T_m is the current time on S. Then S sends the tuple (P, T_m) to the user U_i. Upon receiving (P, T_m), user U_i verifies the server as follows.

1) Checks, whether $T_{cc} - \Delta T \leq T_m$, if not, then U_i rejects the login request. Here, T_{cc} is the current time of U_i.

2) Using the credentials of Step 5 of Section 4.2 user computes $P^{'}$ as $P^{'} = H(CID_i||r_1||T_m)$.

3) Compares if P and $P^{'}$are equal. If equal U_i accepts the server otherwise reject the server and disconnect it.

4.4 Password Update

The modified scheme supports password modification/ update without contacting the server. When a user U_i wants to update his password, he performs the following:

1) Computes $h_{p,cur} = H(PW_{current})$ and $h_{p,new} = H(PW_{new})$.

2) Unlocks the Secret $CID_i = w_i \times g^{-h_{p,cur}} (\mathrm{mod}\, n)$.

3) Compute and writes the new value of w_i, V_i as $w_{i,new} = CID_i \times g^{h_{p,new}} (\mathrm{mod} n)$ and $V_{i,new} = g^{CID_i \times h_{p,new} \times T_r} (\mathrm{mod}\, n)$, into the smart card.

5 Security and Efficiency Analysis

5.1 Security Analysis

This section analyzes the security of the new scheme. A formal analysis using BAN logic [1] is presented besides evaluating the security for various attack scenarios.

5.1.1 Formal Security Proof Using BAN Logic

BAN logic introduced by Barrow et al. in [1] is a formal analysis method to reason about security properties of information exchange protocols. Specifically, BAN

logic helps to determine whether exchanged information is trustworthy, secured against eavesdropping, or both. BAN logic starts with the assumption that all information exchanges happen on media vulnerable to tampering and public monitoring. Hence, it is a good method to analyse the security of remote user authentication protocols.The BAN logic has the advantages of clear concept, simple, easy to understand and use and it can effectively find the secure vulnerability difficult to detect in the protocol. For full details and notion of BAN logic readers are encouraged to go through [1].

The goal of this analysis is to prove that a user U_i and server S can come to a common session key $r_1 = R = K_s$ in a secure way using the proposed protocol. In BAN logic this can be represented as

"U_i believes $U_i \xleftrightarrow{K_s} S$" & "$S$ believes $U_i \xleftrightarrow{K_s} S$"
or ideally

"U_i believes S believes $U_i \xleftrightarrow{K_s} S$" & "$S$ believes U_i believes$U_i \xleftrightarrow{K_s} S$"

Basically, we want U_i and S to establish their session key $K_s$and believe that each has the key that is valid. Also, we would like each to believe that the other believes that they have established the same valid key. To prove the above goal we assume that both U_i and S have believe that registration phase. This can be represented using the following assumptions.

"U_i believes $\xrightarrow{K_{(n,e)}} S$"
"U_i believes $(ID_i, w_i, n, e, g, V_i, T_r)$"
"S believes (ID_i, T_r)"
"S believes $\xrightarrow{PW_i} U_i$"
"U_i controls K_s"

Let us start the proof from authentication phase, where, "S sees (ID_i, X_i, Y_i, T)". Now, since, "S believes (ID_i, T_r)" and "S believes $\xrightarrow{PW_i} U_i$" S verifies and then "S believes U_i said (ID_i, X_i, Y_i, T)". Also, S verifies the fresh (T) and then finally "S believes U_i believes $U_i \xleftrightarrow{K_s} S$" for the present session. In the second part, "U_i sees (P, T_m)". U_i verifies the fresh(T_m) and compares the hash. Since "U_i believes $\xrightarrow{K_{(n,e)}} S$" and "$U_i$ controls K_s" U_i believes that only S can decrypt Y_i and send the tuple (P, T_m) that matches the hash comparison. Therefore,"U_i believes S believes $U_i \xleftrightarrow{K_s} S$".

The above analysis clearly shows that the proposed scheme is secure in establishing a session key between a user U_i and server S over an insecure channel.

5.1.2 Security Analysis for Various Attack Scenarios

We now show that the new scheme can withstand various types of attacks.Here, the adversary A, is assumed to have the capabilities same as that of in Section 3. The details are described as follows.

Privileged insider attack: In the registration phase, user sends $h_p = H(PW_i)$ to the server. The privileged insider of the server could get h_p. However, he cannot get the password PW_i, since, it is protected by a secure hash function. Therefore the modified scheme can withstand the privileged insider attack. Also, password update feature of the scheme provides privileged insider no clue about the value of h_p once the user updates his password. Therefore, modified scheme is can withstand the privileged insider attack.

Password guessing attack: When an attacker gets the user smart card he can extract the values of $(ID_i, CID_i, n, e, g, w_i, V_i, T_r)$ from the card. He can then guess password and compute h_p. But, he cannot verify it using any of the above values without knowing the server secret key d. Therefore, modified scheme is could withstand the password guessing attacks.

User impersonation attack: To impersonate a legal user to login to the server, the adversary could generate a message (ID_i, X_i, Y_i, T). But he cannot have control over or guess the value of Y_i^d which is computed during the verification process. This prevents him the choice of the value of X_i for a give valid time T so that the verification is successful. On the other hand, if attacker chooses the value of X_i then he has to find a value for Y_i such that the verification is successful. This method, also, fails as the decrypted value of Y_i by the server will not contain proper $CID_i^{'}$ for ID_i. Hence Step 5 of Section 4.3 fails. Therefore, the modified scheme could withstand the user impersonation attack.

Server masquerading attack: To impersonate the server to a legal user attacker should face the mutual authentication challenge by the user. As the attacker does not know the value of server secret key$d$, he cannot decrypt Y_i sent by the user. Hence he cannot extract the random value sent by the user and fails to send the reply tuple (P, T_m). Therefore, the modified scheme could withstand the Server masquerading attack.

Reply attack: Suppose that an adversary intercept the login message and replay it to the server. However, the server could find the attack easily by checking the freshness of T. Similarly, a legitimate user can, also, find the replay attack by checking the freshness of T_m. Therefore, the proposed scheme can withstand the replay attack.

5.2 Efficiency Analysis

This section, presents the cost comparison of our scheme with Ramasamy et al.'s scheme along with other smart card based authentication schemes mentioned in [11]. Comparison of computation cost between the schemes is presented using the number of various computation expensive operation involved in Registration Phase, Login Phase and Authentication Phase. Let $E1$, $E2$ and $E3$

Table 1: Comparison of computation cost of various authentication schemes

Scheme	$E1$	$E2$	$E3$
Yang-Shieh [17]	(2,1,0,0)	(2,3,1,0)	(2,1,1,0)
Fan-Li-Zhu [3]	(2,1,0,0)	(2,3,1,0)	(2,1,1,0)
Yang-Wang-Chang [16]	(2,2,0,0)	(2,3,0,0)	(3,1,0,0)
Kumar [6]	(1,0,0,1)	(3,0,2,0)	(2,0,1,1)
Kumar [7]	(1,0,0,1)	(2,0,1,0)	(1,0,1,1)
Ramasamy [11]	(2,3,0,0)	(2,3,0,0)	(3,2,0,0)
Modified Scheme	(2,3,2,0)	(4,5,1,0)	(2,2,3,0)

represents computation cost for Registration Phase, Login Phase and Authentication Phase respectively. $T_{m,exe}$, $T_{m,mul}$, $T_h$and T_{Ck} are the time taken for executing a modular exponentiation, modular multiplication, one-way hash function and to generate check digit for the registered identity. Table 1 presents comparison of computation cost of proposed scheme with other schemes in [11] using the 4 tuple $(T_{m,exe}, T_{m,mul}, T_h, T_{Ck})$ notation.

6 Conclusions

This paper reviewed Ramasamy et al.'s RSA-based remote authentication scheme and analyze its security. Proofs were presented to show that their scheme is vulnerable to privileged insider attack, password guessing attack and impersonation attacks.The impersonation attack proposed is easy to implement and the computation requirements are negligible. Formal security analysis using the BAN logic showed that the proposed new scheme is secure over an insecure channel. Security analysis of the new scheme against various types of attacks showed that it could overcome weaknesses that are in the original scheme. In addition the new scheme lets the users to update their password and authenticate the server during login phase.

References

[1] M. Burrows, M. Abadi, and R. M Needham, "A logic of authentication," in *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, pp. 233–271, 1989.

[2] C. K. Chan and L. M. Cheng, "Cryptanalysis of a timestamp-based password authentication scheme," *Computers & Security*, vol. 21, no. 1, pp. 74–76, 2001.

[3] L. Fan, J. H. Li, and H. W. Zhu, "An enhancement of timestamp-based password authentication scheme," *Computers & Security*, vol. 21, no. 7, pp. 665–667, 2002.

[4] S. K. H. Islam and G. P. Biswas, "Design of improved password authentication and update scheme based on elliptic curve cryptography," *Mathematical and Computer Modelling*, vol. 57, no. 11, pp. 2703–2717, 2013.

[5] H. F. Huang, H. W. Chang, and Po K. Yu, "Enhancement of timestamp-based user authentication scheme with smart card," *International Journal of Network Security*, vol. 16, no. 4, pp. 385–389, 2014.

[6] M. Kumar, "An enhanced remote user authentication scheme with smart card," *International Journal of Network Security*, vol. 10, no. 3, pp. 175–184, 2010.

[7] 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.

[8] M. Kumar, M. K. Gupta, and S. Kumari, "An improved efficient remote password authentication scheme with smart card over insecure networks," *International Journal of Network Security*, vol. 13, no. 3, pp. 167–177, 2011.

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

[10] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC press, 2010.

[11] R. Ramasamy and A. P. Muniyandi, "An efficient password authentication scheme for smart card," *International Journal of Network Security*, vol. 14, no. 3, pp. 180–186, 2012.

[12] J. Ji Shen, C. W. Lin, and M. S. Hwang, "Security enhancement for the timestamp-based password authentication scheme using smart cards," *Computers & Security*, vol. 22, no. 7, pp. 591–595, 2003.

[13] H. Tang, X. Liu, and L. Jiang, "A robust and efficient timestamp-based remote user authentication scheme with smart card lost attack resistance," *International Journal of Network Security*, vol. 15, no. 6, pp. 446–454, 2013.

[14] X. Tian, R. W. Zhu, and D. S. Wong, "Improved efficient remote user authentication schemes," *International Journal of Network Security*, vol. 4, no. 2, pp. 149–154, 2007.

[15] S. T. Wu and B. C. Chieu, "A user friendly remote authentication scheme with smart cards," *Computers & Security*, vol. 22, no. 6, pp. 547–550, 2003.

[16] C. C. Yang, R. C. Wang, and T. Yi Chang, "An improvement of the yang-shieh password authentication schemes," *Applied Mathematics and Computation*, vol. 162, no. 3, pp. 1391–1396, 2005.

[17] W. H. Yang and S. P. Shieh, "Password authentication schemes with smart cards," *Computers & Security*, vol. 18, no. 8, pp. 727–733, 1999.

Prasanth Kumar Thandra has received his M.Sc., Degree in physics in 2002 from KGRL college, Bhimavaram (Andhra University). He is currently working as Scientific Officer (D) in Indira Gandhi Centre for Atomic Research, Department of Atomic Energy, India. His research interests are public key cryptography,cryptanalysis of digital signature and hash functions.

J. Rajan has received his B.E., Degree in E.C.E from University of Madras in 1992. He received his M.S

from BITS Pilani in 1998. He is currently working as Scientific Officer (F) and heading Networking Section, Computer Division in Indira Gandhi Centre for Atomic Research, Department of Atomic Energy, India. His research interests are Network Security, Cryptography and Visualization.

S. A. V. Satya Murty has received his B.Tech Degree from Jawaharlal Nehru Technological University, A.P in 1977. Later, he joined one year orientation course in Nuclear Science & Engineering (21st Batch) at Bhaba Atomic Research Centre (BARC)-Mumbai and then he joined in Indira Gandhi Centre for Atomic Research in 1978. He received his Ph.D., degree from Homi Bhaba National Institute in 2014. He is currently an Outstanding Scientist, Director of Electronics Instrumentation and Radiological Safety Group, IGCAR. He has more than 100 Journal Publications / Conference Papers, 40 Internal Design Reports and edited two International Conference Proceedings. He had written two chapters in important books. His research interest includes Cryptography, High Performance Computer Systems, Grid Computing, Network Security, Wireless Sensor Networks, WSN for Nuclear Reactor Applications, Computational Intelligence, Virtual Reality, and Knowledge Management.