An Improved Remote User Authentication Scheme with Smart Cards using Bilinear Pairings

Debasis Giri and P. D. Srivastava

Department of Mathematics
Indian Institute of Technology, Kharagpur 721 302, India
(E-mail:{dgiri,pds}@maths.iitkgp.ernet.in)

Abstract

Recently, Fang et al [24] proposed an improvement to Das et al's scheme [6] to prevent some weaknesses. Further, Chou et al [19] and Thulasi et al [23] pointed out some weakness of Das et al's scheme. However, the improvd scheme is still insecure to off-line attack. In this paper, we propose an improvement of their schemes that provides the better security compared to the schemes previously published. Further, proposed scheme enables users to choose and change their password by their own choices without the help of a remote server.

Keywords: Authentication; Smart Card; Attacks; Password; Timestamp.

1 Introduction

User authentication is very important mechanism in computer network systems for preventing unauthorized network access. The password-based authentication schemes with smart cards are the important parts of security for accessing remote servers. Password-based authentication is one of the simpler and more convenient authentication mechanisms to deal with secret data over insecure networks. In 1981, Lamport [8] proposed a well-known hash-based password authentication scheme for insecure communication. His scheme resists replay attacks, but requires a verification table to verify the legitimacy of a login user. However, this approach introduces the risk and cost of managing and protecting the table. To avoid such problems, several authentication schemes without the verification table have been proposed [9, 10, 12]. Also, it is difficult for a user to memorize a long key or a server generated password. To overcome this problem, several schemes have been proposed [13, 12] so that the legitimate users can choose their passwords freely. Recently, some related schemes have been proposed [11, 6] for the authentication

using smart cards. In 2005, the Das et al. [6] proposed a scheme for smart card authentication using bilinear pairings that provides the users to choose and change their passwords by their own choices. But, their scheme has some security flaws, which are described in [19, 20]. In 2006, Fang et al [24] proposed an improvement of Das et al's scheme [6] to remedy their weakness. In this paper, we show that Fang et al scheme has still a security weakness. Further, we propose a scheme that can provides the better security compared to Das et al and Fang et al schemes. Besides, in our proposed scheme, users can freely choose their own passwords without any assistance from the remote server.

The remainder of this paper is organized as follows. Section 2 briefly introduces some preliminary mathematical concepts for introducing our proposed scheme. Section 3 briefly reviews the Fang et al's scheme. In Section 4, we describe possible attacks of the Fang et al's scheme. Section 5 introduces our proposed scheme. In Section 6, we discuss the security analysis for our proposed scheme. In Section 7, we compare our proposed scheme with previously published schemes. Finally, Section 8 concludes the paper.

2 Preliminaries

In this section, we briefly review the basic concepts on bilinear pairings and a related mathematical problem.

2.1 Bilinear pairing

The bilinear pairings [18] namely the Weil pairings or Tate pairings may be used in important applications of cryptography and allowed us to construct identity (ID)-based cryptographic schemes. Suppose $< G_1, + >$ be an additive cyclic group of order q generated by P, where q is prime and $< G_2, \times >$ a multiplicative cyclic group of same order as in G_1. A mapping $e : G_1^2 \rightarrow G_2$ is called a bilinear mapping if it satisfies the following properties:

1. *Bilinear property:* For all $Q, R, S \in G_1$, $e(Q + R, S) = e(Q, S) \times e(R, S)$ and $e(Q, R + S) = e(Q, R) \times e(Q, S)$. As a result $e(a * Q, b * R) = (Q, R)^{a \cdot b}$ for all $Q, R \in G_1$ and for all $a, b \in Z_q^*$, where $a * Q$ means a times additions of Q over the group $< G_1, + >$.

2. *Non-degeneracy property:* There exist $Q, R \in G_1$ such that $e(Q, R) \neq 1_{G_2}$, where 1_{G_2} is the identity element of G_2.

3. *Computability property:* There is an efficient algorithm to compute $e(Q, R)$ for all Q, R $\in G_1$.

For implementation point of view, G_1 will be the group of points on an elliptic curve and G_2 will denote a multiplicative subgroup of a finite field. Then there exists a mapping e will be

derived from either the Weil or the Tate pairing on an elliptic curve over a finite field. We refer to [7, 21, 22] for more comprehensive description on how these groups, pairings and other parameters are defined.

2.2 Computational problem

Discrete Logarithm Problem (DLP): Given two elements $Q, R \in G_1$, find an element $x \in Z_q^*$, such that $Q = x * R$ whenever such an element exists.

3 Brief review of the Fang et al's authentication scheme

In this section, we review the Fang et al.'s authentication scheme with smart cards. Their scheme consists of the following important phases, namely, *the setup phase, the registration phase, login phase* and *the verification phase*.

3.1 Set-up phase

The set-up phase proceeds as follows by the remote server (RS, for short). The RS selects two groups: (i) G_1, an additive cyclic group of order prime, say, q, and (ii) G_2, a multiplicative cyclic group of the same order. They define $e : G_1^2 \rightarrow G_2$ is a bilinear mapping and $H : \{0,1\}^* \rightarrow G_1$ a cryptographic hash function. The RS chooses a secret key s and computes the public-key as $Pub_{RS} = s*P$, where P is a generator of the group G_1 . Finally, the RS publishes the following system parameters: G_1, G_2, q, P, Pub_{RS}, the functions e and H and keeps the parameter s as secret.

3.2 Registration phase

In this phase, if a new user U_i wants to register with the RS, he/she submits his/her own identity ID_i as well as his/her password PW_i to the RS. Once the RS receives the registration request, it computes the registration identifier as $Reg_{ID_i} = s * H(ID_i)$ and a point $H_{PW_i} = H(PW_i)$ on $< G_1, + >$ corresponding to the password PW_i Then, the RS issues a smart card with the parameters $ID_i, Reg_{ID_i}, H(\cdot)$ for the user U_i.

3.3 Login phase

In the login phase, the user U_i first inserts his smart card into a card reader and supplies his identifier ID_i and password PW_i. Firstly, smart card computes a dynamic coupon $DID_i = T * Reg_{ID_i}$ and $ET_i = E_{Pub_RS}(T)$, where T is the user system's timestamp. After that it sends the login request $\langle ID_i, DID_i, ET_i \rangle$ to the RS over a public channel.

3.4 Verification phase

Let the RS receive the login message $\langle ID_i, DID_i, ET_i \rangle$ at time $T^*(\geq T)$. In first step, the RS verifies the validity of the time interval between T^* and T. If $(T^* - T) \leq \Delta T$, the RS proceeds for the next step, where ΔT denotes the expected valid time interval for transmission delay. Otherwise, the RS rejects it. In next step, RS first computes $T = E_s(ET_i)$ and then checks whether the equation $e(DID_i, P) = e(H(ID_i), Pub_{RS})^T$ holds or not. In case, the above equation holds, the login request is accepted; otherwise the login request is rejected.

4 Attack on Fang et al's scheme

In this section, we will show that the Fang et al's authentication scheme with smart card is not secured. We have an attack on their scheme as follow:

Off-line attack: Let us assume that an user U_i sends the login request message $\langle ID_i, DID_i, ET_i \rangle$ to the RS and an adversary traps that message at timestamp, say, T_1. It is also known to the adversary that the maximum timestamp difference between the timestamp when legitimate smart card holder sent the login request to the RS and the timestamp when the adversary trapped that sent message, which is denoted by T_M. Now, the adversary can try to compute $\widetilde{ET} = E_{Pub_RS}(\widetilde{T})$ for $\widetilde{T}$ such that $T_1 - T_M \leq \widetilde{T} \leq T_1$ until $\widetilde{ET}$ equals ET_i. Hence, the adversary gets the correct timestamp which is encrypted by the smart card of the user U_i, which be denoted by $\widehat{T}$. As q is the order of G_1 which is a public parameter. As a result, the adversary computes $\widehat{T}^{-1}$ such that $\widehat{T}^{-1} \cdot \widehat{T} = 1 \bmod q$. Then adversary computes $\widehat{T}^{-1} * DID_i$ which is equal to Reg_{ID_i}. Hence, the adversary computes Reg_{ID_i}. After that adversary can create valid login request message in future without knowing password and smart card of the user U_i by the following techniques.

1. Adversary computes $DID'_i = T' * Reg_{ID_i}$, where T' is the current timestamp of its system.
2. It then computes $ET'_i = E_{Pub_RS}(T')$.
3. Next, it transmits the login request message as $M' = \langle ID_i, DID'_i, ED'_i \rangle$ to the RS.

Note that after receiving the message M', the RS can verify the validity of this message M'. Then the verification phase will be correct for this message sent by the adversary. Hence, without knowing password and stolen smart card, the adversary can create the valid login request message.

5 Our scheme

In this section, we present our authentication scheme with smart cards. We discussed four phases of our proposed scheme, namely, *setup*, *registration*, *authentication*, and *password change* phases.

5.1 Set-up phase

The system set-up has the following steps. The setup phase proceeds as follows by the RS. The RS selects two groups: (i) G_1, an additive cyclic group of order prime, say, q, and (ii) G_2, a multiplicative cyclic group of the same order. We define a function $e : G_1^2 \rightarrow G_2$ is a bilinear mapping and $H : \{0,1\}^* \rightarrow G_1$ is a cryptographic hash function. The RS chooses randomly a secret key (private key) s and computes the public-key as $Pub_{RS} = s * P$, where P is a generator of the group G_1. Again, the RS selects a public key cryptosystem, where $E_{Pub_{RS}}(\cdot)$ and $E_s(\cdot)$ are the encryption and decryption algorithms respectively. Finally, the RS publishes the following system parameters: G_1,G_2, q,Pub_{RS}, the functions $e(\cdot,\cdot)$, $H(\cdot)$ and $E_{Pub_{RS}}(\cdot)$. The RS keeps the parameter s as secret.

5.2 Registration

In this phase, an user U_i submits his/her identifier ID_i and password PW_i to the RS. These private data must be sent over a secure channel. Then the RS issues the smart card to the user U_i after performing the following steps:

1. It computes a secret parameter $SP_i = PW_i * Pub_{RS}$.
2. It computes registration identifier of the user U_i as $Reg_{ID_i} = s * H(ID_i) + SP_i$.
3. It loads $Pub_{RS}, ID_i, Reg_{ID_i}, SP_i$ and $H(\cdot)$ in the memory of the smart card and issues the card to U_i.

5.3 Authentication

In this subsection, authentication phase is divided in two phases: (1) *the login phase* and (2) *the verification phase*. These are described as follows:

5.3.1 Login

If the user U_i wants to log into the RS, he/she must insert his/her smart card into a card reader and keys in his identifier ID_i and password PW_i. Then the smart card performs the following steps:

1. The smart card computes $A = PW_i * Pub_{RS}$.
2. It computes $B = Reg_{ID_i} - A$.
3. It randomly selects a number r and computes $C_i = E_{Pub_{RS}}(r)$, where E is the encryption algorithm of public key cryptosystem with public key Pub_{RS}.
4. It computes $D_i = T * B + r * Pub_{RS}$, where $T$is the user system's current timestamp.
5. It sends the login request message $M = \langle ID_i, C_i, D_i, T \rangle$ to the RS over a public channel.

5.3.2 Verification

In this phase, assume that the RS receives the login request message $M = \langle ID_i, C_i, D_i, T \rangle$ at time T', the RS and the smart card will perform the following steps for mutual authentication between the user and the RS.

1. The RS verifies the validity of the time interval between T' and T. If $(T' - T) > \Delta T$, then the RS rejects the login request, where $\triangle T$ denotes the expected valid time interval for transmission delay. Otherwise, it goes for the next step.
2. It computes $X = E_s(C_i)$ and then $Y = X * Pub_{RS}$.
3. It Checks whether $e(D_i - Y, P) = e(H(ID_i), Pub_{RS})^T$. If it holds, the RS accepts the login request; otherwise, rejects it.

5.4 Password change

Our scheme also enables user to change their password freely and securely. If the user U_i wants to change his password from PW_i to PW_i' , he/she should insert his smart card into a card reader and keys in his identifier ID_i and password PW_i. Then the smart card performs the following steps:

1. The smart card computes $SP_i' = PW_i * Pub_{RS}$.
2. The smart card verifies whether SP_i^* and SP_i are equal. If yes, the smart card requests the user for new password and U_i then submits a new password PW_i', otherwise it rejects the password-change-request.
3. The smart card computes $Reg'_{ID_i} = Reg_{ID_i} - SP_i' + PW_i' * Pub_{RS} = s * H(ID_i) + PW_i' * Pub_{RS}$.
4. The password has been changed now with the new password PW_i' and the smart card stores new SP_i' and Reg'_{ID_i} in place of SP_i and Reg_{ID_i} respectively.

5.5 Correctness

Step 3 in the verification phase is verified by the following:

$$\begin{aligned}
D_i - Y &= T * B + r * Pub_{RS} - Y \quad [\text{as } D_i = T * B + r * Pub_{RS}] \\
&= T * B + r * Pub_{RS} - r * Pub_{RS} \\
&\quad [\text{as } Y = X * Pub_{RS} \text{ and } X = E_s(C_i) = E_s E_{Pub_{RS}}(r) = r] \\
&= T * B \\
&= T * (Reg_{ID_i} - A) \quad [\text{as } B = Reg_{ID_i} - A] \\
&= T * (Reg_{ID_i} - PW_i * Pub_{RS}) \quad [\text{as } A = PW_i * Pub_{RS}] \\
&= T * (s * H(ID_i) + PW_i * Pub_{RS} - PW_i * Pub_{RS}) \\
&\quad [\text{as } Reg_{ID_i} = s * H(ID_i) + PW_i * Pub_{RS}] \\
&= T * (s * H(ID_i)) \\
&= (T \cdot s) * H(ID_i) \\
&= (s \cdot T) * H(ID_i) \\
&= s * (T * H(ID_i))
\end{aligned}$$

Therefore,

$$\begin{aligned}
e(D_i - Y, P) &= e(s * (T * H(ID_i)), P) \\
&= e(T * H(ID_i), s * P) \quad [\text{as } e(a * Q, R) = e(Q, a * R)] \\
&= e(T * H(ID_i), Pub_{RS}) \quad [\text{as } Pub_{RS} = s * P] \\
&= e(H(ID_i), Pub_{RS})^T \quad [\text{as } e(Q, b * R) = e(Q, R)^b]
\end{aligned}$$

6 Security analysis of our scheme

In this section, we analyze the security of our proposed scheme as follows:

1. In replay attack, an adversary can attempt to record an exchanged message. The replay of the old request message $M = \langle ID_i, C_i, D_i, T \rangle$ sent by user fails because the validity of these messages can be checked through the timestamp.

2. Let us assume that an adversary traps a valid message $M = \langle ID_i, C_i, D_i, T \rangle$ sent by the user U_i. If the adversary tries to forge the request message $M = \langle ID_i, C_i, D_i, T \rangle$, adversary has to compute s or decrypt to C_i. We consider the following two cases.

   Case-1: If r is known from $C_i = E_{Pub_RS}(r)$, the adversary can compute $r * Pub_{RS}$. Then, it can compute $D_i' = D_i - r * Pub_{RS} + r' * Pub_{RS}$ after choosing a number r' and then

computing $r' * Pub_{RS}$. As a result, a forge message can be $M' = \langle ID_i, C_i', D_i', T \rangle$, where $C_i' = E_{Pub_{RS}}(r')$.

Case-2: If s is known to the adversary, it can compute $r = E_s(C_i)$. Then it can forge the login request message as in Case-1. It can also try from another two ways by the adversary. First one is that the adversary can compute $D_i' = D_i - V_i + W_i$ after computing $V_i = (T \cdot s) * H(ID_i)$ and $W_i = (T' \cdot s) * H(ID_i)$, where T' is slightly different from T. As a result, $M' = \langle ID_i, C_i, D_i', T' \rangle$ can be a forge message. Second one is that the adversary can compute D_i' as $D_i' = (T' \cdot s) * H(ID_i) + r' * Pub_{RS}$, where T' is the current timestamp of the system of the adversary and r' is a random number. Then the adversary computes $C_i' = E_{Pub_{RS}}(r')$. Hence, the adversary can create a valid login request message as $M' = \langle ID_i, C_i', D_i', T' \rangle$.

But, in our scheme, it is computationally infeasible to compute s from given P and Pub_{RS} due to ***DLP***. Further, it is computationally hard to compute r from C_i where $C_i = E_{Pub_{RS}}(r)$, since it is as hard as to decrypt the encrypted message in public key cryptosystem without knowing the secret (private) key s of the RS. Hence, our scheme is secure against these type of attacks.

3. Assume that an adversary stores some valid login request messages $M^{(j)} = \langle ID_i, C_i^{(j)}, C_i^{(j)}, T^{(j)} \rangle$ for $j = 1, 2, \ldots, n$, the adversary has no way to derive another valid message $M' = \langle ID_i, C_i', D_i', T' \rangle$ because of the fact that it is infeasible to compute the secret key s of the RS or decrypt any $C_i^{(j)}$ for $j = 1, 2, \ldots, n$, which are illustrated in the previous attacks.

4. Assume that a user U_i is an adversary. Let us see whether he/she can compute the secret key s of the RS. Even if $s * H(ID_i)$ is known to the user U_i from the stored parameter Reg_{ID_i} in his/her smart card memory after changing the password PW_i to q using the password-change-phase, it is computationally hard for the user U_i being an adversary to derive the server's secret key s from the given $s * H(ID_i)$ and $H(ID_i)$ due to ***DLP***.

7 Comparison

In this section, we compare our scheme with the previous schemes with respect to time complexity required by different phases.

We use the following notations to analyze the computational complexity for our scheme and some existing previous schemes, which are based on public-key cryptosystems for remote user authentication with smart cards.

- t_+ is the time for addition of two elements in the additive group $< G_1, + >$.
- t_{A_G} is the time for $x \in Z_q^*$ times additions in the additive group $< G_1, + >$.
- t_{M_G} is the time for $x \in Z_q^*$ times multiplication in the multiplicative group $< G_2, \times >$.
- t_e is the time for bilinear pairing operation.
- t_H is the time for executing the one-way hash function.
- t_E is the time for encrypting/decrypting a message.

Items $\Rightarrow$ Schemes $\Downarrow$	*registration*	*login*	*verification*	*password change*
Das et al	$2t_H + t_{A_G}$	$2t_{A_G} + t_H$	$t_H + 2t_e + t_{M_G} + t_+$	$2t_H + 2t_+$
Fang et al	$2t_H + t_{A_G}$	$t_{A_G} + t_E$	$t_H + t_E + 2t_e + t_{M_G}$	$2t_H$
Our	$t_H + 2t_{A_G} + t_+$	$3t_{A_G} + 2t_+ + t_E$	$t_H + t_{A_G} + t_+$ + $t_E + 2t_e + t_{M_G}$	$2t_{A_G} + 2t_+$

Table 1: Time complexity for different phases

The computational time of different schemes in registration, login, verification and password change phases are described in Table-1. In Das et al scheme, the computational cost in registration, login, verification and password change phases require $2t_H + t_{A_G}$, $2t_{A_G} + t_H$, $t_H + 2t_e + t_{M_G} + t_+$ and $2t_H + 2t_+$ respectively. In Fang et al scheme, the computational cost in registration, login, verification and password change phases require $2t_H + t_{A_G}$, $t_{A_G} + t_E$, $t_H + t_E + 2t_e + t_{M_G}$ and $2t_H$ respectively. On the other hand, in our scheme, the computational cost in registration, login, verification and password change phases require $t_H + 2t_{A_G} + t_+$, $3t_{A_G} + 2t_+ + t_E$, $t_H + t_{A_G} + t_+ + t_E + 2t_e + t_{M_G}$ and $2t_{A_G} + 2t_+$ respectively. It is observed that each phase of our scheme takes more computational time compared to the Fang et al's and Das et al schemes. But, Das et al's scheme has security flaws which are shown in both Chou et al's scheme [19] and Thulasi et al scheme [23]. Further, we show that Fang et at scheme is insecure to off-line attack. But, our scheme withstands forgery attack, insider attack, off-line attack etc, which are described in Section 6. Hence, our scheme is more secure compared to the Das et al's and Fang et al's schemes.

8 Conclusion

In this paper, we point out a security leak of the Fang et al's scheme with smart card and propose a remote user authentication scheme with smart cards using bilinear pairings that enhances their security by withstanding the weaknesses. The proposed scheme supports the password change phase so that users can choose and change their password freely by their own choices. We also show that the our scheme can resist the different types of possible attacks such as forgery attack, off-line attack, insider attack. Moreover, the scheme provides a flexibility in password change option, where users can choose and their passwords freely and securely without any help of the remote server.

References

[1] C. C. Chang and W. Y. Liao, "A remote password authentication scheme based upon ElGamals signature scheme," Computers & Security, vol. 13, no. 2, pp. 137-144, 1994.

[2] 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.

[3] M. S. Hwang, "A remote password authentication scheme based on the digital signature method," International Journal of Computer Mathematics, vol. 70, pp. 657-666, 1999.

[4] C. C. Lee, M. S. Hwang, and W. P. Yang, "A flexible remote user authentication scheme using smart cards," ACM Operating Systems Review, vol. 36, no. 3, pp. 46-52, 2002.

[5] E. J. Yoon, E. K. Ryu and K. Y. Yoo, "Efficient remote user authentication scheme based on generalized ElGamal signature scheme," IEEE Transactions on Consumer Electronics, vol. 50, no. 2, pp. 568-570, 2004.

[6] M. L. Das, A. Saxena, V. P. Gulati and D. B. Phatak, "A novel remote user authentication scheme using bilinear pairings," Computers and Security, vol. 25, no. 3, pp. 184-189, 2006.

[7] D. Boneh, M. Franklin, "Identity-based Encryption from the Weil pairing," In J. Kilian, editor, Advances in Cryptology-CRYPTO 2001, Springer-Verlag, LNCS, vol. 2139, pp. 213- 229, 2001.

[8] L. Lamport, "Password authentication with insecure communication," Commun ACM, vol. 24, pp.770-772, 1981.

[9] H. Sun, "An efficient remote user authentication scheme using smart cards," IEEE Trans Consumer Electron, vol. 46, no. 4, pp. 958-961, November 2000.

[10] M. Hwang and L. Li, "A new remote user authentication scheme using smart cards," IEEE Trans Consumer Electron, vol. 46, no. 1, pp. 28-30, February 2000.

[11] H. Chien, J. Jan and Y.Tseng, "An efficient and practical solution to remote authentication: smart card," Computers and Security, vol. 21,no. 4, 372-375, 2002.

[12] W. Yang and S. Shieh, "Password authentication schemes with smart cards," Computers and Security, vol.18,no. 8, pp. 727-733, 1999.

[13] K. Tan and H. Zhu, "Remote password authentication scheme with smart cards," Comput Commun, vol. 18, pp. 390-393, 1999.

[14] B. Schneier, "Applied Cryptography," John Wiley & Sons Inc., 1996.

[15] T. T. May, J. W. James, P. H. Bosma, and J. D. Veatch, "Requirements Driven Methodology for accessing the security and business use of smart cards," IEEE International Camahan Conference on Security Technology, pp. 72-88, 1996.

[16] M. L. Gemplus, "Smart-cards: a cost-effective solution against electronic fraude," European Conference on Security and Detection, 28-30 April 1997, Conference Publication No. 437, pp. 81-85, IEE, 1997.

[17] C. C. Chang, and T.C. Wu, "Remote password authentication with smart cards," IEE Proceedings-E, 138 (3), pp. 165-168, 1991.

[18] D. Boneh, B. Lynn and H. Shacham, "Short signatures from the Weil pairing," Advances in Cryptology - Asiacrypt2001, LNCS 2248, Springer-Verlag, pp. 514-532, 2002.

[19] J. S. Chou, Y. Chen, and J. Y. Lin, "Improvement of Manik et al.s remote user authentication scheme," http://eprint.iacr.org/2005/450.pdf, 2005.

[20] T. Goriparthi, M. L. Das, A. Negi, and A. Saxena, "Cryptanalysis of recently proposed Remote User Authentication Schemes ,"http://eprint.iacr.org/2006/028.pdf, 2005.

[21] A. J. Menezes, T. Okamoto and S. A. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field," IEEE Trans. Inf. Theory, vol.39, no. 5, pp. 1639-1646, 1993.

[22] G. Frey and H. G. Rück, "A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves," Math. Comp., vol. 62, no. 206, pp. 865-874, 1994.

[23] G. Thulasi, Manik Lal Das and Ashutosh Saxena," Cryptanalysis of recently proposed Remote User Authentication Schemes," http://eprint.iacr.org/2006/028.pdf

[24] G. Fang and G. Huang, "Improvement of recently proposed Remote User Authentication Schemes," http://eprint.iacr.org/2006/200.pdf.