A Remote User Authentication Scheme using Bilinear Pairings

Sanjeev Kumar
Department of Mathematics,
Institute of Basic Science,
Dr.B.R.Ambedkar University, Khandari,
Agra-282002 (U.P.), India.

K K Goyal
Department of Computer Science
RBS Management Technical Campus
Agra-282002 (U.P), India.

ABSTRACT

In 2006, Das et al. [6] proposed a novel remote user authentication scheme using bilinear pairings. In that scheme, the remote system receives user login request and allows login to the legitimate user. In this paper we present the cryptanalysis of the Das et al. scheme and propose an improved and more secure scheme that enables user to choose and change their password without the help of the remote server.

General Terms

Security, Algorithm, Verification, Reliability

Keywords

Network Security, Cryptography, Authentication; Bilinear pairings; Password; Smart card; Timestamp.

1. INTRODUCTION

Authentication is a one of the main goal of cryptography. There are many techniques used for authentication but password-based authentication is one of the most convenient technique to verify the legitimacy of a user. Classification of password based authentication can be done into two categories such as hash-based (Menezes et al. [19]) authentication and public-key based (IEEE P1363.2 Draft D12 [13]) authentication. The first well-known hash-based password authentication scheme was proposed by Lamport [16]. Later, Shimizu et al. [25] overcome the weakness of Lamport [16] that was suffering from high hash overhead and password resetting problems and proposed a modified scheme. Thereafter, many schemes and improvements (Lee et al. [17], Peyravian and Zunic [21], Ku et al. [14], Ku [15]) on hash-based remote user authentication, have been proposed. These schemes take low computation cost and are computationally viable for implementation in a handheld device like smart card; however, the schemes primarily suffer from password guessing, stolen-verifier and denial-of-service attacks (Ku et al. [14], Hsieh et al. [11]). And the first public key cryptosystem based on the complex discrete logarithm problem was proposed by Diffie and Hellman [5] and after that new area was opened for modern cryptography and several cryptosystems such as RSA [22], Digital Signature by Fiat and Shamir [7] and Authenticated Diffie-Hellman key agreement protocols by Wilson and Menezes [27] were proposed. Because of the higher security level, many researchers worked in this area for proposing the new schemes or improving some already existing applications. In the implementation of public-key cryptosystem computation cost is require very high, but meet higher security requirements.

Recently, the bilinear pairings (Boneh and Franklin [1]) such as Weil pairing or Tate pairing defined on elliptic curves have been found as important applications (Boneh and Franklin [1], Hess [12]) in cryptography and allowed us to fabricate identity (ID) based cryptographic schemes. The bilinear pairings reduce the complexity of the discrete log problem in a finite field (Frey and Ruck [8], Menezes et al [18]) and also provide a good setting for the bilinear Diffie-Hellman problem that has been used to design several cryptosystems. The benefit of a bilinear pairing cryptosystem is that it reduces the computation cost with the same security level. In 1984, Shamir [23] introduced the concept of ID-based cryptosystem; however, the practical ID-based schemes (Boneh and Franklin [1], Cocks [4]) were found in 2001. After that many protocols based on bilinear pairing such as short signature from the weil pairing (Boneh et al [2]), ID-based authentication key agreement protocol based on pairing (Smart [24]) and ID-based signature schemes (Paterson [20]) have been proposed.

In 2006, Das et al. [6] published a novel remote user authentication scheme using bilinear pairings. In that scheme, the remote system receives user login request and allows login to the legitimate user. Before publishing the paper, Chou et al. [3] and Thulasi et al. [26] pointed out some weakness in the Das et al's scheme. In the same year, Fang et al. [9] proposed an improvement to Das et al [6] scheme to prevent some weaknesses. Further, Recently, Goyal and Chahar [10], shown that the scheme is still insecure but they did not propose any solution. Therefore through this work we propose an improved scheme that is also secure against replay, forgery and impersonation attack and also enables user to choose and change their password without the help of the remote server.

The organization of this paper is as follows. In the next section, we present the preliminaries of bilinear pairings and computational problem. In the section following that, we review the Das et al.'s remote user authentication scheme. Cryptanalysis is done in section 4. In section 5, we propose our scheme. Finally, we conclude the paper in the last section.

2. PRELIMINARIES

The basic concepts of bilinear pairings are briefly reviewed.

2.1 Bilinear pairing

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 $\hat{e}: G_1 \times G_1 \rightarrow G_2$ is called a bilinear mapping if it satisfies the following properties:

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

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

Computability property: There is an efficient algorithm to compute $\hat{e}(Q, R)$ 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 $\hat{e}$ will be derived from either the Weil or the Tate pairing on an elliptic curve over a finite field.

2.2 Computational problem

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

Computational Diffiee-Hellman Problem (CDHP): Given (P, aP, bP) for $a, b \in Z_q^*$, compute abP.

3. REVIEW OF DAS ET AL.'S REMOTE USER AUTHENTICATION SCHEME

The scheme consists of mainly three phases: the setup phase, the registration phase and the authentication phase.

3.1 Setup phase

Let G_1 be an additive cyclic group of a prime order q, G_2 be a multiplicative cyclic group of the same order, and Suppose P be a generator of G_1. Define $\hat{e} : (G_1 \times G_1 \rightarrow G_2)$ to be a bilinear mapping and $H: \{0, 1\}^* \rightarrow G_1$ be a cryptographic hash function. Suppose the remote system *(RS)* selects a secret key s and computes his public key as $Pub_{RS} = sP$. Then, the *RS* publishes the system parameters $(G_1, G_2, \hat{e}, q, P, Pub_{RS}, H)$ and keeps s secret.

3.2 Registration phase

This phase is executed by the following steps when a new user U_i wants to register with the *RS*.

Step1. U_i submits his identity ID_i and password PW_i to the *RS*.

Step2. On receiving the registration request, the *RS* computes $Reg_{ID_i} = s.H(ID_i) + H(PW_i)$.

Step3. The *RS* personalizes a smart card with the parameters ID_i, Reg_{ID_i}, $H(.)$ and sends the smart card to U_i over a secure channel.

3.3 Authentication phase

This phase will be executed whenever a user wants to log into the *RS*. We describe it as follows:

a. Login phase

The user U_i inserts the smart card in the terminal and keys ID_i and PW_i. If ID_i is same as stored in the smart card, then smart card perform the following operations:

L1. Compute $DID_i = T * Reg_{ID_i}$ and $V_i = T * H(PW_i)$, where T is the user system's timestamp.

L2. After that, terminal will send the login request $<ID_i, DID_i, V_i, T>$ to the *RS* over the public channel.

b. Verification phase

After receiving the login message $<ID_i, DID_i, V_i, T>$ at time T^*, *RS* will perform the following operations to verify it.

V1. Verify the validity of the time interval between T^* and T. If $(T^* - T) \leq \Delta T$, then *RS* goes to step (V2) else rejects. Here ΔT denotes the time delay which is in the tolerable range by both the user and *RS*.

V2. Checks to see whether $\hat{e}(DID_i - V_i , P) = \hat{e}(H(ID_i), Pub_{RS})^T$ holds, if it holds, *RS* accepts the login request; otherwise, it rejects. The deduction process is as follows:

$\hat{e}(DID_i - V_i , P) = \hat{e}(T * Reg_{ID_i} - V_i , P)$

$= \hat{e}((T(s.H(ID_i) + H(PW_i) - T.H(PW_i), P)$

$= \hat{e}(T(s.H(ID_i)), P)$

$= \hat{e}(s.H(ID_i), P)^T$ [as $\hat{e}(aP, Q) = \hat{e}(P, Q)^a$, bilinearity of $\hat{e}$]

$= \hat{e}(H(ID_i), s.P)^T$ [as $\hat{e}(bP, Q) = \hat{e}(P, bQ)$]

$= \hat{e}(H(ID_i), Pub_{RS})^T$ [$\because Pub_{RS} = s.P$]

3.4 Password change phase

This phase allows U_i to change his password freely. He can easily change his password without taking any assistance from the *RS*. This phase can be described as follows:

Step1. U_i first inputs his correct ID_i and PW_i, and then he submits a newly selected password PW_i^* to the smart card.

Step2. The smart card then does the computation as follows:

$Reg_{ID_i} = Reg_{ID_i} - H(PW_i) + H(PW_i^*) = sH(ID_i) + H(PW_i^*)$.

Thus, the password can be changed to PW_i^* and the smart card will replace the previously stored Reg_{ID_i} by $Reg_{ID_i}^*$.

4. CRYPTANALYSIS OF DAS ET AL.'S REMOTE USER AUTHENTICATION SCHEME

In Das et al.'s [6] scheme they have shown that the scheme can withstand the Replay, Forgery and Insider attack. But in this section we analyze the scheme with different attacks and seen that the scheme can't withstand the following attacks.

Replay attack

Suppose that if user U_i sends the login request message $<ID_i, DID_i, V_i, T>$ to the *RS* and an adversary traps that message at timestamp T_M. 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. Now, the adversary can try to compute $\tilde{T}$ such that $T_M - T \leq \tilde{T} \leq T$ until $\tilde{T}$ equals T. Hence, the adversary gets the correct timestamp which is sent by a legitimate user U_i, which be denoted by $\tilde{T}$. Now, the adversary can computes $\tilde{T}^{-1}$ such that $\tilde{T}^{-1} . \tilde{T} = 1 \bmod q$ where q is the order of G_1 which is a public parameter. Then adversary computes $\tilde{T}^{-1} . DID_i$ which

is equal to Reg_{ID_i}. Hence, the adversary computes Reg_{ID_i}. Using Reg_{ID_i} adversary can create valid login request message in future without knowing password and smart card of the user U_i by the following steps: -

R1. Computes $DID_i' = T'.Reg_{ID_i}$, where T' is the current timestamp of its system.

R2. Computes $V_i'=T'.H(PW')$, where PW' is the adversary's password

R3. Then, transmits the login request message as $<ID_i, DID_i', V_i', T'>$ to the *RS*.

Note that after receiving the message $<ID_i, DID_i', V_i', T'>$, the *RS* can verify the validity of this message. 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.

Forgery attack

Step1. Adversary *X* can record any login message $< ID_i, DID_i, V_i, T >$, sent by U_i who had ever logged into *RS*. And then computes as follows:

$DID_i\text{-}V_i = T.Reg_{ID_i} – T.H(PW_i)$

$= T.[s.H(ID_i)+H(PW_i)] – T.H(PW_i)$

$= T.s.H(ID_i)$

Step2. *X* can pick a random timestamp T^* and computes $T^*.H(PW_j)$, where PW_j is *X's* randomly selected password not confirmed by *RS*.

Step3. *X* computes his DID_j and V_j as follows.

$DID_j = T^*.(DID_i\text{-}V_i) + T^*.T.H(PW_j)$

$= T^*.T.s.H(ID_i) + T^*.T.H(PW_j)$, and

$= T_{NEW}.s.H(ID_i) + T_{NEW}.H(PW_j)$
$[Let\ T.T^* = T_{NEW}]$

$V_j = T^*.T.H(PW_j)$,

$= T_{NEW}.H(PW_j)$ respectively.

Then *X* computes: (Let)

$DID_j\text{-}V_j = T_{NEW}.Reg_{ID_i}\text{-}T_{NEW}.H(PW_i)$

$=\{T_{NEW}.[s.H(ID_i)+ H(PW_i)]\}- T_{NEW}.H(PW_i)$

$= T_{NEW}.s.H(ID_i)$.

Step4. At a later time T_{NEW}, when *X* wants to launch an attack, he can use this forged message $<ID_i, DID_j, V_j, T_{NEW}>$ to masquerade as U_i to *RS*.

RS does not store ID_i and PW_i of any user and it's verification depends only on checking whether $\hat{e}(DID_i\text{-}V_i , P) = \hat{e}(H(ID_i), Pub_{RS})^T$ [Das et al.'s proposed verification equation] holds. If this equation holds, *RS* will accept the forged login message. Clearly, it can be seen that this verification equation holds. Since we already deduce $(DID_j\text{-}V_j)$ to be $T_{NEW}.s.H(ID_i)$ in Step3. As a result, *X* can easily impersonate any valid user. Now *RS* verify the user by the Das et al.'s suggested verification equation after receiving the forged message $<ID_i, DID_j, V_j, T_{NEW}>$.

To overcome this problem Chou et al. suggested the solution to modify the verification equation from $\hat{e}(DID_i\text{-}V_i , P) = \hat{e}(H(ID_i), Pub_{RS})^T$ to $\hat{e}(DID_i, P) = \hat{e}(T.s.H(ID_i)+V_i, P)$.

Impersonation attack

Goyal and Chahar [10] shown that the Chou et al. [3] scheme is still insecure. Steps discussed in section Forgery attack are same but Goyal and Chahar verifies the verification equation using the forged message $<ID_i, DID_j, V_j, T_{NEW}>$ and they found the proposed equation can easily be verified by the forged message $<ID_i, DID_j, V_j, T_{NEW}>$.

$\hat{e}\ (DID_j, P)= \hat{e}\ (T_{NEW}.s.H(ID_i) + T_{NEW}.H(PW_j), P)$ (From Forgery attack, step 3)

$= \hat{e}\ (T_{NEW}.s.H(ID_i)+V_j, P) \quad \because V_j == T_{NEW}.H(PW_j)$

Here we see that the verification is done successfully with that forged message. So the scheme is still insecure after resolvable solution.

5. PROPOSED SCHEME

As we have seen that Das et al. scheme is still insecure against the replay, forgery and impersonation attack. In this section we propose an improved scheme that is also secure against replay, forgery and impersonation attack and also enables user to choose and change their password without the help of the remote server. The all phases of our scheme are as follows:

5.1 Setup phase

Let G_1 and G_2 are the additive and multiplicative cyclic group of a prime order *q* respectively and *P* be a generator of G_1. Bilinear mapping is defined as $e :(G_1 \times G_1 \rightarrow G_2)$ and $H: \{0, 1\}^* \rightarrow G_1$ be a cryptographic hash function. Suppose the remote system (*RS*) selects a secret key *s* and computes his public key as $Pub_{RS} = sP$. Then, the *RS* publishes the system parameters $(G_1, G_2, \hat{e}, q, P, Pub_{RS}, H)$ and keeps *s* secret.

5.2 Registration phase

This phase is executed by the following steps when a new user U_i wants to register with the *RS*.

Step1.U_i submits his identity ID_i and password PW_i to the *RS*.

Step2.On receiving the registration request, the *RS* computes $Reg_{ID_i} = (s + H(PW_i)).H(ID_i))$.

Step3.The *RS* personalizes a smart card with the parameters ID_i, Reg_{ID_i}, $H(.)$ and sends the smart card to U_i over a secure channel.

5.3Authentication phase

This phase will be executed whenever a user wants to log into the *RS*. We describe it as follows:

a. login

Suppose the ID_i of user U_i is stored in the smartcard and U_i wants to login to the remote system, and then the smart card will process the login operation after U_i has inserted the smart card and inputted the ID_i and PW_i to the terminal. For example, the smart card will compute $DID_i = T.Reg_{ID_i}$ and $V_i = T.H(PW_i)$, where *T* is the user system's timestamp. After

that, terminal will send the login request $<ID_i, DID_i, V_i, T>$ to the *RS* over the public channel.

b. verification

After receiving the login message $<ID_i, DID_i, V_i, T>$, *RS* will perform the following operations to verify it.

Step1. Verify the validity of the time interval between T^* and *T*. If $(T^* - T) \leq \Delta T$, then *RS* goes to step2 else rejects. Here ΔT denotes the time delay which is in the tolerable range by both the user and *RS*.

Step2. Checks to see whether $e(DID_i, P) = e(H(ID_i), Pub_{RS})^T \times (H(ID_i).V_i, P))$ holds, if it holds, *RS* accepts the login request; otherwise, it rejects. The deduction process is as follows:

$$\hat{e}(DID_i, P) = \hat{e}(T.Reg_{ID_i}, P)$$

$$= \hat{e}((T.(s + H(PW_i)).H(ID_i)), P)$$

$$= \hat{e}(T.s.H(ID_i) + T.H(ID_i).H(PW_i)), P)$$

$$= \hat{e}(T.s.H(ID_i), P) \times \hat{e}(T.H(ID_i) * H(PW_i), P)$$

$$= \hat{e}(H(ID_i), P)^{T.s} \times \hat{e}(V_i * H(ID_i), P)$$

$$= \hat{e}(H(ID_i), sP)^T \times \hat{e}(V_i * H(ID_i), P)$$

$$= \hat{e}(H(ID_i), Pub_{RS})^T \times \hat{e}(V_i * H(ID_i), P)$$

$$\because Pub_{RS} = sP$$

5.4Password change phase

This phase allows U_i to change his password freely. He can easily change his password without taking any assistance from the *RS*. This phase can be described as follows:

Step1. U_i first inputs his correct ID_i and PW_i, and then he submits a newly selected password PW_i^* to the smart card.

Step2. The smart card then does the computation as follows:

$$Reg_{ID_i} = Reg_{ID_i} - H(PW_i).H(ID_i) + HPW_i^*).H(ID_i)$$

$$= (s + H(PW_i)).H(ID_i) - H(PW_i).H(ID_i) + H(PW_i^*).H(ID_i)$$

$$= s.H(ID_i) + H(ID_i).H(PW_i) - H(PW_i).H(ID_i) + H(PW_i^*).H(ID_i)$$

$$= s.H(ID_i) + H(PW_i^*).H(ID_i)$$

$$= (s + H(PW_i^*)).H(ID_i)$$

Thus, the password can be changed to PW_i^* and the smart card will replace the previously stored Reg_{ID_i} by $Reg_{ID_i}^*$.

6. CONCLUSION

In this paper, we have reviewed the Das et al.'s [6] scheme and also shown that the scheme is still insecure against replay, forgery and impersonation attack by the cryptanalysis. Finally, we have proposed improved scheme that is also secure against replay, forgery and impersonation attack and also enables user to choose and change their password without the help of the remote server.

7. REFERENCES

[1] Boneh D, Franklin M. Identity-based encryption from the Weil pairing. In: Advances in Cryptology-Proceeding of CRYPTO 2001, LNCS, vol. 2139. Springer-Verlag; 2001. p. 213–29.

[2] Boneh D, Lynn B, Shacham H. Short signatures from the Weil pairing. In: Advances in Cryptology-Asiacrypt2001, LNCS, vol. 2248, Springer-Verlag; 2002. p. 514-32.

[3] Chou JS, Chen Y, Lin JY. Improvement of Malik et al.'s remote user authentication scheme. http://eprint.iacr.org/2005/450.

[4] Cocks C. An identity based encryption scheme based on quadratic residues. In: Cryptography and coding, LNCS, vol. 2260. Springer-Verlag; 2001. p. 360-63.

[5] Diffie W, Hellman M. New directions in cryptography, IEEE Transactions on Information Theory 1976; 22(6): p. 644–54.

[6] Das ML, Saxena A, Gulati VP, Phatak DB. A novel remote user authentication scheme using bilinear pairings. Computers & Security 2006; vol. 25(3), p.184-89.

[7] Fiat A, Shamir A. How to prove yourself: practical solutions to identification and signature problems. in: Advances in cryptology-Crypto'86, LNCS, vol. 263. Springer-verlag; 1987. p 186-94.

[8] Frey G, Ruck H. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. Mathematics of Computation 1994; vol. 62(206): p.865–74.

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

[10] Goyal KK, Chahar MS. Cryptanalysis of a Novel Remote User Authentication Scheme. International Journal of Computer Science, System Engineering and Information Technology (IJCSSEIT) 2011; vol 4(1): p. 99-102.

[11] Hsieh BT, Sun HM, Hwang T. On the security of some password authentication protocols. Informatica 2003;14(2):195-204.

[12] Hess F. Efficient identity based signature schemes based on pairings. In: Selected areas in cryptography'02, LNCS, vol. 2595. Springer-Verlag; 2003. p. 310-24.

[13] IEEE P1363.2 draft D12: standard specifications for password based public key cryptographic techniques. IEEE P1363 working group; 2003.

[14] Ku WC, Chen CM, Lee HL. Weaknesses of Lee-Li-Hwang's hash-based password authentication scheme. ACM Operating Systems Review 2003; 37(4):9-25.

[15] Ku WC. A hash-based strong-password authentication scheme without using smart cards. ACM Operating Systems Review 2004; 38(1):29-34.

[16] Lamport L. Password authentication with insecure communication. Communications of the ACM 1981; 24(11):770-2.

[17] Lee CC, Li LH, Hwang MS. A remote user authentication scheme using hash functions. ACM Operating Systems Review 2002; 36(4):23-9.

[18] Menezes A, Okamoto T, Vanston S. Reducing elliptic curve logarithms to logarithms in a finite field. IEEE

Transaction on Information Theory 1993; vol. 39(5), p.1639–46.

[19] Menezes A, van Oorschot PC, Vanstone S. Handbook of applied cryptography. CRC Press; 1996.

[20] Paterson KG. ID-based signature from pairings on elliptic curves. Electronics Letters 2002; vol. 38(18), p.1025–26.

[21] Peyravian M, Zunic N. Methods for protecting password transmission. Computers & Security 2000;19(5):466-9.

[22] Rivest RL, Shamir A, Adelman L. A method for obtaining digital signature and public key cryptosystem. Communications of ACM 1978; vol. 21(2), p.120–126.

[23] Shamir A. Identity-Based cryptosystems and signature schemes. Advances in Cryptology, LNCS, 1984; p. 47–53.

[24] Smart NP. Identity based authenticated key agreement protocol based on Weil pairing. Electronics Letters 2002; vol. 38(13), p.630–32.

[25] Shimizu A, Horioka T, Inagaki H. A password authentication methods for contents communication on the Internet. IEICE Transactions on Communications 1998; E81-B(8):1666-73.

[26] Thulasi G, Das ML, Saxena A. Cryptanalysis of recently proposed Remote User Authentication Schemes. http://eprint.iacr.org/2006/028.pdf

[27] Wilson SB, Menezes A. Authenticated Diffie-Hellman key agreement protocols. In: Proceedings of the 5th Annual Workshop on Selected Areas in Cryptography (SAC ’98), LNCS 1999; p.339-61.