Cryptanalysis and Improvement of a Recently Proposed Remote User Authentication Scheme Using Smart Cards

S.Sharmila Deva Selvi, S.Sree Vivek

February 1, 2008

Abstract

Recently Debasis et al[1] proposed an improvement to prevent offline attack in Fang et al's[2] scheme, where [2] was an improvement of Das et al's[3] scheme. However the improved scheme is insecure against side channel attack. In this paper we propose an enhancement for [1]. The enhanced scheme is secure against substitution, impersonation, spoofing, replay, side-channel and password guessing attacks.
Keywords : Cryptanalysis, Authentication, Smart cards, Side-channel attack.

1 Introduction

Remote user authentication scheme allows an user to login into a remote server in computer network systems. In 1981, Lamport [4] proposed a remote user authentication scheme using smart cards. Based on the concept proposed by Lamport, several authentication schemes were proposed. Most of them suffered from attacks which were cryptanalysed by various authors. In 2005, Das et al[3] proposed a smart card based remote user authentication scheme using bilinear pairing. In 2006, Fang et al [2] pointed out weakness in Das et al's scheme and proposed enhancement for the weaknesses in their scheme. Recently Debasis et al [1] showed that [2] suffered from offline attack. In this paper we show that [1] is vulnerable to side channel attack and we propose an enhancement for the security pitfall in it.
The structure of our paper is organized as follows. In the following section we give a brief note on the preliminaries and following that we present a

brief review of Debasis et al's scheme. Next, we show the weakness of [1] and in the sections following that we propose our scheme, analyze its security against various attacks and compare it with the parent schemes. Finally we conclude the paper in the last section.

2 Preliminaries

2.1 Bilinear Pairing

Let G_1 be an additive cyclic group generated by P, whose order is a prime q, and G_2 be the multiplicative cyclic group of the same order q. A bilinear pairing is a map $e : G_1 \times G_1 \longrightarrow G_2$ with the following properties:

1. Bilinearity: For all $P, Q, R \in G_1$,
$$e(P + Q, R) = e(P, R)e(Q, R),$$
$$e(P, Q + R) = e(P, Q)e(P, R),$$
$$e(aP, bQ) = e(P, Q)^{ab}.$$

2. Non-degeneracy: There exists $P, Q \in G_1$ such that $e(P, Q) \neq I_{G_2}$ where I_{G_2} is the identity element of G_2

3. Computable: There exists an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in G_1$.

2.2 Computational Assumptions

In this subsection we discuss some of the hard problems that act as backbone for our scheme. Let G be a group of prime order q and $P, Q \in G$.

1. Elliptic Curve Discrete Logarithm Problem (ECDLP): Given P and Q find $x \in Z_q^*$ such that $Q = xP$.

2. Elliptic Curve Computational Diffie Hellman Problem (ECCDHP): For any $a, b \in Z_q^*$, given the values of $< P, aP, bP >$, finding out the value of abP is called the CDH problem in elliptic curves.

3. Elliptic Curve Decisional Diffie Hellman Problem (ECDDHP): For any $a, b, c \in Z_q^*$ given the values of $< P, aP, bP, cP >$, deciding whether $cP = abP$ is called the DDH problem in elliptic curves.

2.3 Definitions

The security requirements for an authentication scheme are.

1. **Unforgeablity under replay attack:** An authentication scheme should be secure, in the sense that it should not authenticate some one else as a legitimate user, even if an adversary uses a valid login message of a legitimate user sent in the past.

2. **Security against impersonation attack:** An authentication scheme should be secure in the sense that no adversary should be able to login as a legitimate user even if he gets access to any number of login messages of any user and any data from the smart card of any user and passwords of any user other than the person he is trying to impersonate.

3. **Security against substitution attack:** Even if the adversary intercepts a valid login message and changes some of the values he should not be able to succeed in the impersonation attack.

4. **Security against spoofing attack:** Even if the adversary observes 'r' different valid login message send by an user he should not be able to form a forged login message and get authenticated.

5. **Security against side-channel attack:** A side-channel attack is any attack based on information gained from the physical implementation of a cryptosystem, rather than theoretical weaknesses in the algorithms. For example, timing information, power consumption, electromagnetic leaks or even sound can provide an extra source of information which can be exploited to break the system.

3 Brief review of Debasis et al's authentication scheme

In this section we review Debasis et al's [1] scheme. The following are the phases of the scheme.

3.1 Setup Phase

Remote server (RS) does the following

1. RS selects an additive cyclic group G_1 and a multiplicative cyclic group G_2 of same order q, where q is a prime.

2. Let $e : G_1 \times G_1 \longrightarrow G_2$ define a bilinear map.

3. Let $H : \{0,1\}^* \longrightarrow G_1$ be a cryptographic hash function.

4. RS chooses a value s at random and keeps it as private key and computes $Pub_{RS} = sP$ where P is a generator of the group G_1.

5. The RS selects a public key cryptosystem, where $Enc_{Pub_{RS}}(.)$ and Dec_s are the encryption and decryption algorithms respectively.

6. Finally RS publishes $< G_1, G_2, e(.,.), H(.), Pub_{RS}, q$ and $Enc_{Pub_{RS}}(.) >$ as system parameters .

3.2 Registration Phase

The user U_i submits his identity ID_i and password PW_i to the RS through a secure channel. The RS issues the smart card after performing the following steps:

1. Compute a secret parameter $SP_i = PW_i Pub_{RS}$.

2. Then the RS computes $Reg_{ID_i} = sH(ID_i) + SP_i$ which is the registration identifier of the user U_i.

3. It loads $< ID_i, Pub_{RS}, Reg_{ID_i}, SP_i, H(.) >$ in the memory of the smart card and issues it to user U_i

3.3 Authentication Phase

Authentication phase is divided into *login phase* and *verification phase* which are described as follows.

3.3.1 Login Phase

User U_i inserts his smart card into the card reader and enters his identity ID_i and password PW_i. Then the following steps are performed in the user machine.

1. Compute $A = PW_i Pub_{RS}$, $B = Reg_{ID_i} - A$.

2. Select a number r at random and compute $C_i = Enc_{Pub_{RS}}(r)$.

3. Compute $D_i = TB + rPub_{RS}$, where T is the current timestamp.

4. It sends the login request message $M =< ID_i, C_i, D_i, T >$ to RS over a public channel.

3.3.2 Verification Phase

Let RS receive the login message at time T'. The following verifications are done by the RS to authenticate the user U_i.

1. Verifies the validity of the time interval between T and T'. If $(T'-T) > \Delta T$, then the RS rejects the login request else proceed to the next step.
2. Computes $X = Dec_s(C_i)$ and then $Y = XPub_{RS}$
3. Checks whether $e(D_i - Y, P) = e(H(ID_i), Pub_{RS})^T$, If it holds accept else reject.

3.4 Password Change

To change the old password PW_i to a new password PW_i' the following steps are performed.

1. Compute $SP_i^* = PW_i Pub_{RS}$ after accepting PW_i from the user.
2. Verify whether SP_i in the smart card is equal to SP_i^*. If so accept the new password PW_i' else reject.
3. Compute $SP_i' = PW_i' Pub_{RS}$ and $Reg'_{ID_i} = Reg_{ID_i} - SP_i^* + SP_i'$.
4. Replace SP_i' and Reg'_{ID_i} in place of SP_i and Reg_{ID_i} in the smart card.

4 Attack on Debasis et al's scheme

In this section we show that Debasis et al's scheme is inherent to side-channel attack.

Side-channel attack: In this scheme, U_1's smart card contains SP_i and Reg_{ID_i} after registration. The adversary could have extracted the secret information stored in the smart card by monitoring the power consumption [5] or by analyzing the leaked information [6]. If the smart card is not assumed to be tamper proof the adversary can access the secret details stored in a smart card after stealing it or from a lost smart card of U_i. Thus the adversary obtains SP_i and Reg_{ID_i}. Knowing these values the adversary launches the following attack to impersonate U_i :

1. Using SP_i and Reg_{ID_i} the adversary computes $Reg_{ID_i} - SP_i = sH(ID_i)$.
2. Choose a random number k and compute $C_i' = Enc_{Pub_{RS}}(k)$, this can be done because $Enc_{Pub_{RS}}$ is a publicly known encryption algorithm.

3. Compute $D_i' = T''sH(ID_i) + kPub_{RS}$, where T'' is the timestamp at which the adversary launches the impersonation attack and Pub_{RS} is a public parameter.

4. Now $< ID_i, C_i', D_i', T'' >$ is a valid login message computed by the adversary which is send to the RS to impersonate U_i.

It can be easily verified that the C_i' and D_i' values computed above can easily satisfy the verification done in authentification phase as demondtrated below:

The server computes $k = Dec_s(C_i')$ and $Y' = kPub_{RS}$ and does the verification with these values.

$$\begin{aligned} e(D_i' - Y', P) &= e(T''sH(ID_i + kPub_{RS} - kPub_{RS}, P) \\ &= e(T''sH(ID_i), P) \\ &= e(H(ID_i), sP)^{T''} \\ &= e(H(ID_i), Pub_{RS})^{T''} \end{aligned}$$

Therefore it is clear that $< ID_i, C_i', D_i', T'' >$ is a valid login message.

5 Our scheme

In this section we present the enhanced remote user authentication scheme based on smart cards. There are five phases in our scheme. The phases of our scheme are explained below.

5.1 Setup Phase

The Remote Server (RS) which is responsible for setting up the system does the following.

1. RS selects an additive cyclic group G_1 and a multiplicative cyclic group G_2 of same order q, where q is a prime.

2. Let $e : G_1 \times G_1 \longrightarrow G_2$ define a bilinear map.

3. Let $H : \{0,1\}^* \longrightarrow G_1$ be a cryptographic hash function.

4. RS chooses a value $s \in {}_RZ_q^*$ at random and keeps it as private key and computes $Pub_{RS} = sP$ where P is a generator of the group G_1.

5. The RS selects a public key cryptosystem, where $Enc_{Pub_{RS}}(.)$ and Dec_s are the encryption and decryption algorithms respectively.

6. Finally RS publishes $< G_1, G_2, e(.,.), H(.), Pub_{RS}, q$ and $Enc_{Pub_{RS}}(.) >$ as system parameters .

5.2 Registration Phase

The user U_i submits his identity ID_i and password PW_i to the RS through a secure channel. The RS issues the smart card after performing the following steps:

1. The RS computes $Reg_{ID_i} = sPW_iH(ID_i)$ which is the registration identifier of the user U_i.

2. It loads $< ID_i, Pub_{RS}, Reg_{ID_i}, H(.) >$ in the memory of the smart card and issues it to user U_i

5.3 Authentication Phase

Authentication phase is divided into *login phase* and *verification phase* which are described as follows.

5.3.1 Login Phase

User U_i inserts his smart card into the card reader and enters his identity ID_i and password PW_i. Then the following steps are performed in the user machine.

1. Computes $B_i = PW_i^{-1}Reg_{ID_i}$.

2. Selects a number r at random and compute $C_i = Enc_{Pub_{RS}}(r)$.

3. Computes $D_i = TB_i + rPub_{RS}$, where T is the current timestamp.

4. It sends the login request message $M =< ID_i, C_i, D_i, T >$ to RS over a public channel.

5.3.2 Verification Phase

Let RS receive the login message at time T'. The following verifications are done by the RS to authenticate the user U_i.

1. Verifies the validity of the time interval between T and T'. If $(T'-T) > \Delta T$, then the RS rejects the login request else proceed to the next step.
2. Computes $X = Dec_s(C_i)$ and then $Y = XPub_{RS}$
3. Checks whether $e(D_i - Y, P) = e(H(ID_i), Pub_{RS})^T$, If it holds accept else reject.

5.4 Password Change

To change the old password PW_i^* to a new password PW_i' the following steps are performed.

1. Accept the old password, let it be denoted as PW_i^* and the new password PW_i' from the user.
2. Verify whether $e(Reg_{ID}, P) = e(H(ID), sP)^{PW_i^*}$. If so accept the new password PW_i' else reject.
3. Compute $Reg'_{ID_i} = PW_i^{*-1} PW_i' Reg_{ID_i}$.
4. Replace Reg'_{ID_i} in place of Reg_{ID_i} in the smart card.

6 Proof of Correctness

In this section we show the proof of correctness of our scheme.

6.1 Verification Phase

$D_i - Y = TB_i + rPub_{RS}$
$= TPW_i^{-1} Reg_{ID_i} + rPub_{RS}$
$= TPW_i^{-1} PW_i H(ID_i) + rPub_{RS}$
$= sTH(ID_i) + rPub_{RS}$

$e(D_i - Y, P) = e(sTH(ID_i) + rPub_{RS} - rPub_{RS}, P)$
$= e(sTH(ID_i), P)$
$= e(sH(ID_i), P)^T$

6.2 Password Change Phase

$e(Reg_{ID}, P) = e(sPW_i H(ID_i), P)$
$= e(PW_i H(ID), sP)$
$= e(H(ID), sP)^{PW_i}$

7 Security Analysis

In this section we show that our scheme is secure against replay, impersonation, spoofing, substitution, side-channel and password guessing attacks.

7.1 Replay attack

It is impossible for an adversary to replay a login message sent at past by a legitimate user without altering any values in it because of the use of timestamp. If the login message does not reach the server within the stipulated network delay ΔT the message will be considered to be an invalid message.

7.2 Impersonation Attack

It is not possible for an adversary to compute a valid login message of a legitimate user because computing a value D_i in the login message requires the value of $sH(ID_i)$. $H(ID_i)$ can be computed easily because H(.) is a public hash function but s is the secret key of the server. Finding s from the public parameters as well as the values stored in the smart card will lead to ECDLP. Also computing $sH(ID_i)$ from Reg_{ID} stored in the smart card is not possible because it requires the password PW_i. Security against impersonation attack implies security against spoofing and substitution because changing any value in a login message or accessing any value of the smart card of the user does not help in the construction of a valid login message.

7.3 Side-channel Attack

A side-channel attack is one in which the user gets access to the entries of the smart card. In our scheme even if the adversary gets the value Reg_{ID} from the smart card he will not be able to get the value $sH(ID_i)$ which can be used to find out the value of D_i. Without knowing the value of PW_i it is impossible to find B_i. Thus it is never possible to find a valid D_i for a chosen timestamp.

7.4 Password Guessing Attack

Even if the adversary tries to guess the password, he will not be able to succeed in launching an attack because there are two unknown quantities in the Reg_{ID} value in the smart card. namely s and PW_i. So to find the

password PWi using Reg_{ID}, he should know s, solving for s will lead to ECDLP.

8 Performance Comparision

In this section we present the comparision of our scheme with the parent schemes [1],[2] and [3].

Scheme	*HO*	*PA*	*SM*	*PC*	*ED*	*EX*
[3]	2	-	1	-	-	-
[2]	2	-	1	-	-	-
[1]	1	1	2	-	-	-
Our	1	-	1	-	-	-

Table 1: Registration Phase

Scheme	*HO*	*PA*	*SM*	*PC*	*ED*	*EX*
[3]	1	-	2	-	-	-
[2]	-	-	1	-	1	-
[1]	-	2	3	-	1	-
Our	-	1	3	-	1	-

Table 2: Login Phase

Scheme	*HO*	*PA*	*SM*	*PC*	*ED*	*EX*
[3]	1	1	-	2	-	1
[2]	1	-	2	-	1	1
[1]	1	1	-	2	1	1
Our	1	-	1	2	1	1

Table 3: Verification Phase

9 Conclusion

In this paper we point out the security weakness in [1] and propose an improvement to the scheme. Our scheme is resilient to replay, impersonation,

Scheme	*HO*	*PA*	*SM*	*PC*	*ED*	*EX*
[3]	2	1	1	-	-	-
[2]	2	-	-	-	-	-
[1]	-	2	2	-	-	-
Our	1	-	1	2	-	1

Table 4: Password Change Phase
HO - Hash Operation
PA - Point Addition
SM - Scalar Point Multiplication
PC - Pairing Computation
ED - Encryption/Decryption
EX - Pairing Exponentiation

spoofing, substitution, offline and password attacks. In this scheme it is possible for an user to use passwords with out being guessed.

References

[1] Debasis Giri and P. D. Srivastava: *An Improved Remote User Authentication Scheme with Smart Cards using Bilinear Pairings,* http://eprint.iacr.org/2006/274.pdf.

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

[3] Das ML, Ashutosh Saxena, Gulati VP, Phatak DB: *A novel remote user authentication scheme using bilinear pairings,* 2005, Computers Security.

[4] L. Lamport: *Password Authentication with Insecure Communication,* Communications of the ACM, vol. 24, no. 11, pp 770-772, 1981.

[5] P. Kocher, J. Jaffe, B. Jun : *Differential power analysis,* Proc. Advances in Cryptology (CRYPTO'99), 1999, pp. 388397.

[6] T.S. Messerges, E.A. Dabbish, R.H. Sloan: *Examining smart card security under the threat of power analysis attacks,* IEEE Transactions on Computers 51 (5) (2002) 541552.

Contents