International Journal of Applied Mathematics and Computation
ISSN: 0974 - 4665 (Print) 0974 - 4673 (Online)

Journal homepage: www.darbose.in/ijamc
Volume 4(4) 2012 382–389

An improved remote user authentication scheme with smart cards using bilinear pairings

Amit K. Awasthi(✉)[a]

[a]Department of Applied Mathematics, Gautam Buddha University, Greater Noida, Uttar Pradesh, 201308, India

ABSTRACT

Manik et al. [3] proposed a novel remote user authentication scheme using bilinear pairings. Recently, Fang et al [15] analyzed this scheme and pointed out that the proposed scheme is insecure. They proposed an improvement over this scheme. Further, Giri and Srivastava [16] observed that the improved scheme of Fang et al is still insecure against off-line attack and they suggested a new improved scheme. However, the improved scheme is still insecure. In this paper, we show some attacks on this scheme and propose an improved protocol that provides the better security as compared to the schemes previously discussed.

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 the 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 [5] proposed a well-known password authentication scheme for insecure communication. His scheme 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 tables. To avoid such problems, several authentication schemes without the verification table have been proposed. 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 so that the legitimate users can choose their passwords freely. Recently, some related schemes have been proposed [11–15] for the authentication using smart cards. In 2005, the Das et al. [3] 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 [12–16]. In 2006, Fang et al [15] proposed an improvement over Das et al's scheme [3] to remedy their weakness. In continuation to that , in 2006, Giri and Srivastava [16] proposed an improvement over Fang et al scheme [15].

In this paper, we have shown that the proposed scheme of Giri and Srivastava is still insecure against the theft and on-line attack. In this paper, we propose an improvement over their schemes that provides the better security as 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.

This paper is organized as follows. Section 2 briefly introduces some preliminary mathematical concepts used in proposed scheme. Section 3 briefly reviews the Fang et al's scheme. In Section 4, Security flaws of the Fang et

✉ Corresponding author.
Email address: awasthi.amitk@gmail.com (Amit K. Awasthi)

al's scheme given by Giri et al are discussed. Section 5 briefly reviews Giri et al's scheme. In Section 6, we show possible attacks on the Giri et al's scheme. In Section 7, we introduce our scheme. 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 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 $< G_1,\ \times >$ a multiplicative cyclic group of same order as of G_1. We define a mapping $e : G_1 \times G_1 \rightarrow G_2$, called a bilinear mapping if it satisfies the following properties:

Bilinear property $e(aP, bQ) = e(P, Q)^{ab}$, for all $P, Q \in G_1$ and $a, b \in Z_q^*$

Non-degeneracy property There exist $P, Q \in G_1$ such that $e(P, Q) \neq 1_{G_2}$

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

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.

2.2 *Computational problems*

The security of schemes based on the hardness of following computational problems:

2.2.1 *Discrete Logarithm Problem*

For a given generator P of G_1 and $Q \in G_1$, find an element $a \in Z_q^*$ such that $aP = Q$.

2.2.2 *Computational Diffie-Hellman (CDH) Problem*

Let P be a generator of G_1. Given $\langle P, aP, bP \rangle \in G_1$ compute abP for $a, b \in Z_q^*$.

2.2.3 *Bilinear diffie-Hellman (BDH) Problem:*

Let P be a generator of G_1. Given $\langle P, aP, bP, cP \rangle \in G_1$ compute $e(P, P)^{abc}$, for $a, b, c \in Z_q^*$.

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, the login phase and the verification phase.

3.1 *Setup phase*

The setup phase proceeds as follows by the remote server (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.

It defines $e : G_1 \times G_1 \to G_2$ is a bilinear mapping and $H : \{0,1\}* \to G_1$ a cryptographic hash function. The RS chooses a secret key s and computes the public-key as $Pub_{RS} = sP$, 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}, e and H. It 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 $RegID_i = sH(ID_i)$ and a point $H(PW_i)$ on $< G_1, + >$ corresponding to the password PW_i. Then, the RS issues a smart card with the parameters ID_i , $RegID_i$, $H(.)$ 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 $DID_i = T.RegID_i$ and $ET_i = E_{Pub_{RS}}(T)$, where T is the user system's timestamp. After that it sends the login request $< ID_i; DID_i; ET_i >$ to the RS over a public channel.

3.4 *Verification phase*

Let the RS receive the login message $< ID_i; DID_i; ET_i >$ at time T' (=T). In first step, the RS verifies the validity of the time interval between T' and T. If $(T' - T) = \Delta T$ (acceptable duration), 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

$$e(DIDi, P) = e(H(ID_i), Pub_{RS})^T$$

holds or not. In case, the above equation holds, the login request is accepted; otherwise the login request is rejected.

4. Cryptanalysis of 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 $< ID_i; DID_i; ET_i >$ 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 $E_T = EPub_{RS}(T)$ for T such that $T_1 - T_M = T = T_1$, until E_T 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 T, as q is the order of G_1 which is a public parameter. As a result, the adversary computes T^{-1} such that $T^{-1}.T = 1 \mod q$. Then adversary computes $T^{-1}.DID_i$ which is equal to $RegID_i$. Hence, the adversary computes $RegID_i$, 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 $DID'_i = T'.RegID_i$, 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 $M' =< ID_i; DID'_i; ED'_i >$ 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. Brief review of the Giri et al's authentication 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 $e : G_1 \times G_1 \to G_2$ is a bilinear mapping and $H : \{0; 1\}* \to G_1$ is a cryptographic hash function. The RS chooses randomly a secret key (private key) s and computes the public-key as $Pub_{RS} = sP$, where P is a generator of the group G_1. Again, the RS selects a public key cryptosystem, where $E_{Pub_{RS}}(.)$ and $E_s(.)$ are the encryption and decryption algorithms respectively. Finally, the RS publishes the following system parameters: G_1, G_2, q, Pub_{RS}, e, H(.) and $E_{Pub_{RS}}(.)$. 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 *PWi* 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 $SPi = PWi.Pub_RS$.
2. It computes registration identifier of the user U_i as $RegIDi = s.H(ID_i) + SP_i$.
3. It loads Pub_{RS}, ID_i , $RegID_i$, SP_i and $H(.)$ 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: the login phase and 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 $B = RegID_i - A$.
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 $M =< IDi, Ci, Di, Ti >$ to the RS over a public channel.

5.3.2 *Verification*

In this phase, assume that the RS receives the login request message $M =< IDi, Ci, Di, Ti >$ 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 ΔT 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 $RegID_i' = RegID_i - SP_i'$ + PW'$_i.Pub_{RS} = sH(ID_i) + PW_i'.Pub_{RS}$.

4. The password has been changed now with the new password PW_i' and the smartcard stores new SP_i' and $RegID_i'$ in place of SP_i and $RegID_i$ respectively.

6. Attack on Giri et al's scheme

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

Theft attack: As user U_i submits his/her identifier *IDi* and password *PWi* to the RS. Then 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 *RegIDi = s.H*(ID_i) + *SPi.*
3. It loads *PubRS , IDi , RegIDi , SPi* and *H*(.) in the memory of the smart card and issues
the card to U_i.
Because the RS loads *PubRS , IDi , RegIDi , SPi* and *H*(.) in the memory of the smart card & suppose the smart card has been stolen then the adversary can create valid login request massage in future as follows :
1. *RegIDi* & *SPi* are stored on smart card adversary can find the *s*H*(ID_i)
*s*H*(ID_i) *=RegIDi − SPi*
2. Now adversary will create the new *SP'i* by new password as
SP'I =PW'i *PubRS
3. New *Reg'ID = s*H(IDi)+SP'i*

Now adversary loads PubRS, ID_i , Reg_{ID_i} , SP_i and H(.) in the memory of the smart card and uses the card as it is used by *Ui.,* until card theft is not detected and it is not blocked.

7. Our Modification

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.

7.1 *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 $e : G_1^2 \to G_2$ is a bilinear mapping and *H*:{0, 1}* ? G_1 is a cryptographic hash function. The RS chooses randomly a secret key (private key) s and computes the public-key as PubRS = *s.P*, where P is a generator of the group G_1. Again, the RS selects a public key cryptosystem, where $EPub_{RS}(.)$ and *Es*(.) are the encryption and decryption algorithms respectively. Finally, the RS publishes the following system parameters: G_1, G_2, q, Pub_{RS}, *e*, *H*(.) and EPubRS(.). The RS keeps the parameter s as secret.

7.2 *Registration*

In this phase, an user U_i submits his/her identifier ID_i and password *PWi* 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 $Reg_{ID_i} = sH(ID_i) + SP_i$.
3. It loads $Pub_{RS}, ID_i, Reg_{ID_i}$. and H(.) in the memory of the smart card and issues the card to U_i.

7.3 *Authentication*

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

7.3.1 *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 *PWi.* Then the smart card performs the following steps:
1. The smart card computes $A = PW_i * Pub_{RS}$.
2. It computes *B = RegIDi - A*.
3. It randomly selects a number r and computes $C_i = EPub_{RS}(r)$, where *E* is the encryption algorithm of public key cryptosystem with public key PubRS.
4. It computes *Di = T.B + r.PubRS*, where T is the user system's current timestamp.
5. It sends the login request message *M = <IDi, Ci, Di, Ti>* to the RS over a public channel.

7.3.2 *Verification*

In this phase, assume that the RS receives the login request message *M = <IDi, Ci, Di, Ti>* at time $T\prime$, 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 $T\prime$ and T. If ($T\prime$− T) >$?T$, then the RS rejects the login request, where $?T$ denotes the expected valid time interval for transmission delay. Otherwise, it goes for the next step.

2. It computes $X = Es(Ci)$ and then $Y = X.PubRS$.
3. It Checks whether $e(Di - Y, P) = e(H(ID_i), Pub_{RS})T$. If it holds, the RS accepts the login request; otherwise, rejects it.

7.4 *Password change*

Our scheme enables user to change their password, but online, not offline. If the user U_i wants to change his password from *PWi* to *PWi* , he/she should insert his smart card into a card reader and keys in his identifier ID_i , old password *PWi*., new password *PWn*., Then the smart card performs the following steps:
1.The server computes $SPi = PWi .PubRS$.
2.Checks the validity $RegIDi = s.H(IDi) + SPi.$, *if* valid it computes $Reg_{ID_n} = sH(ID_i) + SP_n.$, where $SPn = PWn .PubRS$
3.It loads $Pub_{RS}, ID_i, Reg_{ID_i}$.n and H(.) in the memory of the smart card and issues the card to U_i.

8. Conclusions

In this paper, we analyzed both Feng et al.'s scheme and Giri et al.'s scheme. We propose an improvement for the flaws we found. Now this scheme can stand with theft attack as well as all the vulnerability proposed in previous schemes. We also modified the password change protocol, which is in our scheme is unlike from other schemes is online line. Offline secure password change protocol is still to be a open problem.

References

[1] 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.

[2] 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.

[3] 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, 2005.

[4] 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.

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

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

[7] M. S. 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.

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

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

[10] 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.

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

[12] 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.

[13] 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.

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

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

[16] D. Giri and P. D. Srivastava, “An Improved Remote User Authentication Scheme with Smart Card using Billinear Pairings.” http://eprint.iacr.org/2006/274.pdf.