Pairing based Mutual Authentication Scheme Using Smart Cards

G. Shailaja, K. Phani Kumar, Ashutosh Saxena

Secure Technology Lab.,
Institute for Development and Research in Banking Technology
Castle Hills, Masab Tank, Hyderabad 500057, INDIA.
{gshailaja, kpkumar}@mtech.idrbt.ac.in, asaxena@idrbt.ac.in

Abstract

Bilinear pairings based mutual authentication scheme using smart card is presented. We propose a novel technique of using two different servers, one for registration and other for authentication. The scheme is resilient to replay, forgery, man-in-the-middle and insider attacks.

Keywords: Mutual authentication, Bilinear pairings, smart card

1 Introduction

Financial transactions and paid services over the web have grown tremendously in the recent times. With the significant increase in phising attacks, in addition to user authentication, it is also necessary that service provider authenticates themselves to the user to increase consumer confidence. Mutual authentication is required between the communicating parties, prior to any business transaction. Further, the recent developments in the smart card technology and the growing demand for secure applications has lead to lot of research work being done in the area of smart card based systems.

In this paper, we present a bilinear pairings based mutual authentication scheme using smart cards. Bilinear pairings [1] are special kinds of maps that pair two elements of an n-torsion subgroup of an elliptic curve and produce an element of a suitable finite field. Suitable maps are obtained from the Weil and the Tate pairing on special kinds of elliptic curves [2]. Elliptic curve cryptography is more efficient than integer factorization systems and discrete logarithm systems in terms of key sizes and bandwidth for schemes of relative security. These features make it especially attractive for secure applications where computational power is limited such as smart cards or any hand-held computation device.

We present a novel technique of using two different servers, a registration server to register new users and an authentication server to authenticate the registered users. Further two different secrets are used for the user authenticating to the server and the server authenticating to the user, to mitigate the risk of insider attacks. The scheme is resilient to replay, forgery, man-in-the-middle and insider attacks.

Related Work: Lamport [3] introduced the first well-known hash-based password authentication scheme, but the scheme suffers from high hash overhead and password resetting problems. Later, Shimizu et al. [4] overcame the weakness in [3] and proposed a modified scheme. Thereafter, many schemes and improvements [5] [6] [7] [8] [9] on hash-based remote user authentication have been proposed. These schemes take low computation cost and are computationally viable for implementation in a hand-held device like smart card; however, the schemes primarily suffer from password guessing, stolen-verifier, insider and denial-of-service attacks [7] [10] [11]. In contrast, public-key cryptography based authentica-

tion schemes require high computation cost for implementation, but meet higher security requirements. Several remote user authentication schemes [12] [13] [14] [15] which are based on public-key cryptography have been proposed. Recently a remote user authentication scheme using bilinear pairings is proposed by Das et al [16]. Chou et al. [17] pointed out the vulnerabilities in [16] and suggested an improvement. Very recently, Thulasi et al. [18] have cryptanalysed both the schemes [17] and [16].
Chien et al. [19] have proposed a mutual authentication scheme using smart cards. Unfortunately it was shown by Hsu [20] that this scheme is vulnerable to parallel session attack. Chien's scheme was later improved by Yeh [21]. Yoon et al. [22] have proposed a mutual authentication and key exchange scheme based on generalized ElGamal signature scheme. Wang et al. [23] have shown that the previous session keys will be compromised if the secret key of the system is leaked in Yoon et al.'s scheme. They have also proposed a new scheme which provides forward secrecy. Our scheme requires the bilinear pairing operations to be computed only at the server side. The computations done at the client side involve only hash based operations. This makes our scheme especially attractive for the applications with a powerful server and number of clients with low computational capabilities. Rest of the paper is organized as follows: Section 2 gives the background concepts on bilinear pairings and some related mathematical problems in brief. Section 3 presents our scheme and its correctness, performance and security are analyzed in Section 4. Section 5 concludes the paper.

2 Background Concepts

In this section, we first briefly review the basic concepts on bilinear pairings and some related mathematical problems.

2.1 Bilinear Pairings

Let G_1 and G_2 be additive and multiplicative cyclic groups of same prime order q and let P be an arbitrary generator of G_1. A cryptographic bilinear map is defined as $e : G_1 \times G_1 \rightarrow G_2$ with the following properties:

Bilinear: For all $R, S, T \in G_1$, $e(R + S, T) = e(R, T)e(S, T)$ and $e(R, S + T) = e(R, S)e(R, T)$.

Non-degenerate: There exists $R, S \in G_1$ such that $e(R, S) \neq I_{G_2}$ where I_{G_2} denotes the identity element of G_2.

Computable: There exists an efficient algorithm to compute $e(R, S)$ $\forall R, S \in G_1$.

In general implementation, G_1 will be a group of points on an elliptic curve defined under elliptic curve point addition (+) and G_2 will be a multiplicative subgroup of a finite field.

2.2 Mathematical Problems

Here, we discuss some mathematical problems, which form the basis of security for our scheme. Let G be a group of prime order q and P, $Q \in G^*$.

Discrete Logarithm Problem (DLP): Given P, Q, find an integer $x \in Z_q^*$ such that $Q = xP$.
Computational Diffie-Hellman Problem (CDHP): For any $a, b \in Z_q^*$, given $\langle P, aP, bP \rangle$, compute abP.

Decisional Diffie-Hellman Problem(DDHP): For any $a, b, c \in Z_q^*$, given $\langle P, aP, bP, cP \rangle$, decide whether $c \equiv ab \bmod q$.

Gap Diffie-Hellman Group (GDH): We call a group G a GDH group iff DDHP can be solved in

polynomial time but no probabilistic algorithm can solve CDHP with non-negligible advantage within polynomial time [24].

3 Proposed Scheme

The proposed scheme involves four entities: user, user's smart card, registration server(RS) and authentication server(AS). The user has to first register with the registration server to access the services. The user is issued a personalized smart card during the registration. Mutual authentication is carried out between the authentication server and the user to offer and access the services respectively. The scheme consists of three phases - the setup phase, the registration phase and the mutual authentication phase. Password change option is also provided for the user.

3.1 Setup phase

Suppose G_1 is an additive cyclic group of prime order q, and G_2 is a multiplicative cyclic group of the same order. We assume that solving CDHP is hard in group G_1. Suppose P is a generator of G_1. $e : G_1 \times G_1 \rightarrow G_2$ is a bilinear mapping and $\mathcal{H}_1 : \{0,1\}^* \rightarrow G_1$ and $\mathcal{H}_2 : \{0,1\}^* \rightarrow Z_q^*$ are cryptographic hash functions. The RS selects a secret key s and computes the public key as $Pub_{RS} = sP$. The AS chooses a secret key s_0 and passes s_0P to the RS. The system public parameters are $\langle G_1, G_2, e, q, P, Pub_{RS}, \mathcal{H}_1, \mathcal{H}_2 \rangle$. This phase is executed only once.

3.2 Registration phase

When a new user registers with RS, the following steps are executed using secure off-line channel:
R1. The new user U_i submits his identity ID_i and password PW_i to the RS.
R2. On receiving the registration request, the RS computes $S_{ID_i} = s\mathcal{H}_1(ID_i)$, $P_{ID_i} = \mathcal{H}_1(PW_i)$ and $A_{ID_i} = e(\mathcal{H}_1(ID_i), s_0P)$
R3. The RS personalizes a smart card with the parameters ID_i, S_{ID_i}, P_{ID_i},A_{ID_i}, $\mathcal{H}_1(.)$,$\mathcal{H}_2(.)$ and issues the smart card to the user U_i in secure manner.

3.3 Mutual Authentication phase

This phase is executed whenever a user wants to log into the remote server to access the services. This phase is further divided into the login, user authentication and server authentication phases. In the login phase, user sends a login request to the AS. The AS first authenticates the user and then authenticates itself to the user.

3.3.1 Login Phase

The user U_i inserts the smart card in the reader connected to user computer system and enters the ID_i and PW_i. The smart card performs the following operations:
L1. Checks if ID_i is identical to the one in the smart card. If not, the login request fails. Otherwise, the smart card proceeds to the next step.
L2. Computes $\mathcal{H}_1(PW_i)$ and checks if it is equal to P_{ID_i}. If they are equal, then it proceeds to the next step. Otherwise, login request fails.
L3. Chooses a random number $r \in Z_q^*$
L4. Computes $V = rP$
L5. Computes $W = r^{-1}(S_{ID_i} + hP)$, where $h = \mathcal{H}_2(t||V_x||V_y)$, t is the user computer system's timestamp,

V_x and V_y are the x and y coordinates of the point V.
L6. Sends the login request $< ID_i, V, W, t >$ to the AS over a public channel.

3.3.2 User Authentication Phase

AS receives the login request $< ID_i, V, W, t >$ at time $t^* (\geq t)$. The AS performs the following operations to verify the login request:
C1. Verifies the validity of the time interval between t^* and t. If $(t^* - t) \leq \triangle t$, the AS proceeds to the next step. Otherwise, the login request is rejected. Here $\triangle t$ denotes the expected valid time interval for transmission delay. We note that at the time of registration, the user and the AS have agreed on the accepted value of the transmission delay $\triangle t$.
C2. Checks whether $e(W, V) == e(\mathcal{H}_1(ID_i), Pub_{RS}).e(P, P)^h$, where $h = \mathcal{H}_2(t||V_x||V_y)$. If it holds, the AS authenticates the user; Otherwise, rejects it.

3.3.3 Server Authentication Phase

The AS does the following to authenticate itself to the user:
S1. Computes $l = H_2(e(s_0\mathcal{H}_1(ID_i), V)||t)$ and sends $\langle l, t \rangle$ to the user.
S2. On receiving the tuple $\langle l, t \rangle$, the client computes $H_2({A_{ID_i}}^r||t)$ and compares it with l. If they are equal, then the user successfully authenticates the server.

3.4 Password Change Phase

This phase is invoked whenever a user U_i wants to change his password. This phase does not require any interaction with the servers and works as follows:
P1. U_i inserts the smart card into the terminal and enters the ID_i and PW_i. If ID_i is identical to the one stored in the smart card, it proceeds to the next step; Otherwise, terminates the operation.
P2. It then checks if the password entered i.e. PW_i is correct or not by checking the following equality: $P_{ID_i} == \mathcal{H}_1(PW_i)$. If this holds good, then it proceeds to next step; Otherwise terminates the operation.
P3. U_i submits a new password PW_i^*.
P4. The smart card computes $P_{ID_i}^* = \mathcal{H}_1(PW_i^*)$
P5. Smart card now replaces P_{ID_i} with $P_{ID_i}^*$ and this completes the password change from PW_i to PW_i^*.

4 Correctness, Performance and Security

4.1 Correctness

The correctness of the user authentication phase is verified by the following:

$$\begin{aligned} e(W, V) &= e(r^{-1}(S_{ID_i} + hP), rP) \\ &= e(s\mathcal{H}(ID_i) + hP, P) \\ &= e(s\mathcal{H}(ID_i), P).e(hP, P) \\ &= e(\mathcal{H}(ID_i), Pub_{RS}).e(P, P)^h \end{aligned}$$

The correctness of the server authentication phase is described as follows:

$$
\begin{aligned}
l &= H_2(e(s_0\mathcal{H}_1(ID_i), V)||t) \\
&= H_2(e(s_0\mathcal{H}_1(ID_i), rP)||t) \\
&= H_2(e(\mathcal{H}_1(ID_i), s_0P)^r||t) \\
&= H_2({A_{ID_i}}^r||t)
\end{aligned}
$$

4.2 Salient Features and Performance

Our scheme enjoys the following features:

- The scheme prevents the scenario of many logged in users with the same login-ID: Even if user's password is leaked or the user has revealed his password, the adversary cannot login to AS without the smart card. It should be noted that the smart card has to be inside the reader connected to user computer system during the entire login session. If the user removes the smart card after successful login, the login session would immediately expire. Thus, we are able to prevent the scenario of many users simultaneously logging in with the same login-ID.
- The scheme provides a user-friendly password change option to the user without any assistance from remote servers: The user may wish to change password to avoid password guessing attacks. Our scheme provides a very convenient way of changing the user password which does not require any interaction with servers. This feature greatly reduces the network traffic and overhead on remote servers.
- The AS need not maintain any password or verifier table: The overhead on AS of maintaining password or verifier table does not exist in our scheme.
- The scheme uses two different servers for registration and authentication purposes: In some of the previous schemes [25] [26] , server secret is required during the authentication process. But this may lead to increased risk of insider attacks, since the secret has to be used very often for every user authentication. If the server secret is revealed, then an attacker can introduce fraudulent users into the system as well as he can provide fraudulent services to the genuine users.
  In our scheme, we use two different servers, one for registration and another for authentication. The secrets s and s_0P are stored securely in the registration server and are used only for registering new users. These secrets are not required during the authentication process. The secret s_0 stored in the authentication server will be used for the server authentication. Even if s_0 is revealed, an attacker cannot introduce fraudulent users into the system, thus protecting the interests of the service provider.

The computations required during the different phases of the proposed scheme are shown in the table 1. We note that $e(P, P)$ which is used during user authentication can be initially pre-computed.

4.3 Security Analysis

Some of the smart card manufacturers consider the risk of the side channel attacks, and provide counter measures to deter the reverse engineering attempt. The smart card is programmed in such a way that it is extremely difficult to extract the values from the smart card and we consider it as a secure device. Even if the smart card is lost/stolen, the impersonator is not capable of login/change the password since the scheme requires entering the correct password. AS maintains a database of tuples $< ID_i, V, W, t >$, where the tuple is stored in the database for $\triangle t$ time. We consider the following possible attack scenarios:

	H_1	H_2	E	PA	SM	BP
Registration Phase	2	-	-	-	1	1
Login Phase	1	1	-	1	3	-
User authentication Phase	1	1	1	-	-	2
Server authentication Phase	1	2	-	-	1	1

H_1: H_1 hash operation
H_2: H_2 hash operation
E: Exponentiation
PA: Elliptic curve point addition
SM: Elliptic curve scalar multiplication
BP: Bilinear pairing

Table 1. Computations required in the proposed scheme

4.3.1 Replay Attack

Suppose an adversary replays an intercepted valid login request, which is received by the AS at the time t_{new}.
RA1. AS computes $t_{diff} = (t_{new} - t)$
RA2. If $t_{diff} > \triangle t$, the login fails.
RA3. Otherwise, AS checks if the tuple already exists in the database. If so, AS identifies this as a replay attack and rejects the login request.

4.3.2 Forgery Attack

FA1. It is not possible for an adversary to construct a valid S_{ID_i}, as it requires the RS secret key s. So, an adversary cannot produce smart cards, thus preventing him from introducing fraudulent users into the system.
FA2. An adversary cannot calculate the term l as it requires the knowledge of AS secret key s_0, thus preventing him from providing fraudulent services to genuine users.

4.3.3 Man-in-the-Middle Attack

MA1. If an adversary changes the timestamp, he cannot construct a valid login request, since the term W in the login request tuple contains the hash of the time stamp.
MA2. Similarly, the term V cannot be changed as any modification in V can be later identified during the verification process when h is computed.
MA3. It is not possible for an adversary to construct a new W with the changed value of V, since this requires the knowledge of r. Computing r from V is equivalent to solving Elliptic Curve Discrete Logarithm problem(ECDLP).

4.3.4 Insider Attack

There exists an inherent risk of passwords being stolen if server maintains password or verifier table for login request verification. Our scheme does not require to store the user passwords in AS, thus eliminating such risks. Moreover, the authentication phase does not require the server secret s, which is used in the registration process. A different secret s_0 is used during server authentication, thus mitigating the risk of insider attacks.

5 Conclusion

Bilinear pairings based mutual authentication scheme is presented in this paper. We present a novel approach of using two different servers, one for registration and other for authentication. Using this approach, we minimize the risk of compromising the registration server secret key, which can be prone to attack in cases where same server is being used for both registration and authentication. The scheme is resilient to replay, forgery, man-in-the-middle and insider attacks.

References

[1] Boneh D, Franklin M, Identity-based Encryption from the Weil pairing, 2001, Crypto'01, LNCS vol. 2139, pp.213-229.

[2] Barreto PSLM, Kim HY, Lynn B, Scott M, Efficient algorithms for pairing-based cryptosystems, 2002, LNCS vol. 2442, pp.354-368.

[3] Lamport L, Password authentication with insecure communication, 1981, Communications of ACM, vol. 24(11), pp. 770-772.

[4] Shimizu A, Horioka T, Inagaki H, A password authentication method for contents communication on the Internet, 1998, IEICE Transactions on Communications, vol. E31-B(8), pp. 1666-1673.

[5] Lee CC, Li LH, Hwang MS, A remote user authentication scheme using hash functions, 2002, ACM Operating Systems Review, vol. 36(4), pp.23-29.

[6] Peyravian M, Zunic N, Methods for protecting password transmission, 2000, Computers and Security, vol. 19(5), pp. 466-469.

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

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

[9] Das ML, Ashutosh Saxena, Gulati VP, A Dynamic ID-based Remote User Authentication Scheme, 2004, IEEE Transactions on Consumer Electronics, vol. 50(2).

[10] Kim HS, Lee SW, Yoo KY, ID-based Password Authentication Scheme using smart cards and fingerprints, 2003, ACM SIGOPS Operating Systems Review archives, vol. 37, pp.32-41.

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

[12] Chang CC, Wu TC, Remote Password Authentication with smart cards, 1993, IEE proceedings, vol. 138(3), pp. 165-168.

[13] Chang CC, Liao WY, A remote password authentication scheme based upon ElGamal's signature scheme, 1994, Computers and Security, vol. 13(2), pp. 137-144.

[14] Hwang JJ, Yeh TC, Improvement on Peyravian-Zunic's password authentication schemes, 2002, IEICE transactions on Communcations, vol. E85-B(4), pp. 823-825.

[15] Shen JJ, Lin CW, Hwang MS, A modified remote user authentication scheme using smart cards, 2003, IEEE Transactions on Consumer Electronics, vol. 49(2), pp. 414-416.

[16] Das ML, Ashutosh Saxena, Gulati VP, Phatak DB, A novel remote user authentication scheme using bilinear pairings, 2005, (in press): Computers & Security.

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

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

[19] Chien HY, Jan JK, Tseng YM, An efficient and practical solution to remote authentication: Smart Card, 2002, Computers and Security, vol. 121(4), pp. 372-373.

[20] Hsu CL, Security of Chien et al.s remote user authentication scheme using smart cards, 2004, Computer Standards & Interfaces, Elsevier, vol. 26, pp. 167169.

[21] Yeh HT, Improvement of an Efficient and Practical Solution to Remote Authentication: Smart Card, 2006, IEICE Transactions on Communications, vol. E89B(1), pp. 210-211.

[22] Yoon EJ, Ryu EK, Yoo KY, Efficient remote user authentication scheme based on generalized ElGamal signature scheme, 2004, IEEE Transactions on Consumer Electronics, vol. 50(2), pp. 568-570.

[23] Wang B, Li ZQ, A Forward-Secure User Authentication Scheme with Smart Cards, 2006, International Journal of Network Security, vol.3(2), PP. 108111.

[24] Boneh D, Lynn B, Shacham H, Short Signatures from the Weil Pairing, 2001, Asiacrypt'01, LNCS vol. 2248, pp. 514-532.

[25] Hwang MS, Li LH, A new remote user authentication scheme using smart cards, 2000, IEEE Transactions on Consumer Electronics, vol. 46(1).

[26] Awasthi AK, Sunder Lal, A remote user authentication scheme using smart cards with forward secrecy, 2003, IEEE Transactions on Consumer Electronics, vol. 49(4).