Commun. Korean Math. Soc. **20** (2005), No. 4, pp. 849–859

IDENTITY BASED AUTHENTICATED KEY AGREEMENT FROM PAIRINGS

Hyang-Sook Lee* and Young-Ran Lee

Abstract. We present a new identity based authenticated key agreement protocol from pairings satisfying the required security attributes. The security of our protocol is based on the bilinear Diffie-Hellman assumption.

1. Introduction

Diffie-Hellman protocol [7] is well known two party key exchange protocol. However the basic Diffie-Hellman protocol does not authenticate the two communication parties, hence it suffers from the man-in-the-middle attack. A simple solution would be to combine a key agreement protocol with a digital signature scheme to obtain an authenticated key agreement protocol. Over the years, different approaches have been developed to solve the problem [5, 11, 17, 18]. For instance, the MTI[12] and MQV[11] family of protocols are based on using Certificate Authority to bind an entity's identity with his static (long-term) keys. The final session key is generated by both ephemeral (short-term) keys and static (long-term) keys. The authenticity of the static keys assures that only the entities who possess the static keys are able to compute the session keys. However, in a certification system, before using the public key of a user, the participants must first verify the certificate of the user. As a consequence, this system requires a large amount of computing time and storage.

In 1984, Shamir[16] suggested the concept of identity (ID) based cryptography to simplify key management procedures in certificate-based public key infrastructure. Since then there have been many identity

Received February 23, 2005.

2000 Mathematics Subject Classification: 94A60, 11T71.

Key words and phrases: ECC, pairings, authenticated key agreement protocol, identity based scheme.

*The author was supported by KOSEF grant No. R06-2002-012-01001-0.

based cryptography like encryption schemes, signature schemes and key agreement schemes [2, 4, 9, 15] etc. Especially many ID based cryptographic schemes are based on the pairings. Many applications using pairings [2, 3, 4, 9, 14, 15] have been proposed in cryptography. Smart [18] proposed an ID based authenticated key (AK) agreement protocol based on Weil pairing, Boneh-Franklin ID based encryption scheme and tripartite Diffie-Hellman protocol. However this scheme is weak if two long term private keys are compromised. Chen and Kudla[5] presented a two-pass AK protocol that overcame the flaw of Smart's protocol. As pointed out in [6], their scheme is inefficient in terms of performance attributes of AK protocols. Recently Choie *et al.*[6] suggested an ID based AK protocol that reduced the communication overhead. However, this scheme is less efficient on computational cost than the previous schemes. This paper propose an ID based authenticated key agreement protocol using Weil or Tate pairings which provides the TA forward secrecy, and hence perfect forward secrecy. We also give the security properties of our protocol.

This paper is organized as follows. Section 2 discusses the security attributes of the key agreement protocols, ID based schemes and the bilinear Diffie-Hellman assumption. In section 3, we review some known authenticated key agreement protocols and provide an efficient new identity-based authenticated key agreement scheme.

2. Preliminaries

2.1. Identity based schemes

In 1984, Shamir[16] suggested a public key encryption scheme in which the public key can be an arbitrary string. Shamir's original motivation for identity (ID) based encryption was to simplify certificate management in e-mail systems. When Alice sends mail to Bob at *bob@ewha.ac.kr*, she simply encrypts her message using the public key string *bob@ewha.ac.kr*. There is no need for Alice to obtain Bob's public key certificate. When Bob receives the encrypted mail he contacts a third party with trusted authority (TA), which we call Private Key Generater (PKG). Bob authenticates himself to the PKG in the same way he would authenticate himself to a CA and obtains his private key from the PKG. Bob can then read his e-mail. Note that unlike the existing public key infrastructure, Alice can send encrypted mail to Bob even if

Bob has not yet setup his public key certificate. Also note that key escrow is inherent in the identity based system since the PKG knows Bob's private key. In such an identity scheme, PKG generates global system parameters and a *master-key*, generate the private key corresponding to an arbitrary public key string $ID \in \{0,1\}^*$ using the master key, and then send the private keys to the users.

2.2. Security attributes

A *key agreement protocol* is a key establishment technique in which a shared secret is derived by two (or more) parties as a function of information contributed by, or associated with, each of these, such that no party can predetermine the resulting value. A key agreement protocol is said to provide *implicit key authentication* (of B to A) if A is assured that no other entity besides B can possibly ascertain the value of the secret key. *Authenticated key agreement protocol* (AK protocol) is a key agreement protocol which provides mutual implicit key authentication. A key agreement protocol provides *key confirmation* (of B to A) if A is assured that B possesses the secret key. *Authenticated key agreement with key confirmation protocol* (AKC protocol) provides mutual key authentication and mutual key confirmation.

Now we give the desirable security attributes of the key agreement protocols.

Known-key security : Each run of the protocol should result in a unique secret session key. The compromise of one session key should not compromise other session keys.

Forward secrecy : If long-term private keys of one or more of entities are compromised, the secrecy of previously established session keys should not be affected.

- *Partial forward secrecy* : If the long-term key of one or more but not all are compromised, the secrecy of previously established session keys should not be affected.
- *Perfect forward secrecy* : If the long-term key of all entities are compromised, the secrecy of previously established session keys should not be affected.
- *TA forward secrecy* : If the TA's long-term key is compromised(and hence all users' long-term private keys), the secrecy of previously established session keys should not be affected.

Key compromise impersonation : The compromise of an entity A's long-term private key will allow an adversary to impersonate A, but it should not be able the adversary to impersonate other entities to A.

Unknown-key share : If the adversary convinces the user B that he shares some session key with the adversary, while in fact he shares the key with another entity (user A), we call the protocol suffering from unknown-key share attack.

2.3. Bilinear Diffie-Hellman assumption

Let G_1 be an additive group of prime order l and G_2 be a multiplicative group of the same order l. We assume discrete log problems in G_1 and G_2 are hard. We consider a pairing map $e : G_1 \times G_1 \longrightarrow G_2$ satisfying the following properties.

(i) Bilinearity : $e(aP, bQ) = e(P, Q)^{ab}$ for all $P, Q \in G_1$, and for all $a, b \in \mathbb{Z}_l^*$.

(ii) Non-degeneracy : The map does not send all pairs in $G_1 \times G_1$ to the identity in G_2. Observe that since G_1 and G_2 are groups of prime order this implies that if P is a generator of G_1, then $e(P, P)$ is a generator of G_2.

(iii) Computability : Given $P, Q \in G_1$, $e(P, Q)$ can be efficiently computable.

A bilinear map satisfying the three properties above is said to be an *admissible bilinear map.* We note that the Weil and Tate pairings associated with supersingular elliptic curves or abelian varieties can be modified to create such bilinear maps. We consider G_1 to be an additive abelian group defined on elliptic curves. We refer to [2, 13] for more details.

We consider an admissible bilinear map $e : G_1 \times G_1 \longrightarrow G_2$ defined as above. Let P be a generator of G_1.

Bilinear Diffie-Hellman Problem (BDHP) : The BDH problem in $< G_1, G_2, e >$ is as follows. Given $P, aP, bP, cP \in G_1$, compute $e(P, P)^{abc} \in G_2$ where a, b, c are randomly chosen from $\mathbb{Z}_l^*$. An algorithm is said to solve the BDH problem with an advantage of ϵ if

$$\Pr[\mathcal{A}(P, aP, bP, cP) = e(P, P)^{abc}] \geq \epsilon,$$

where the probability is over the random choice of a, b, c in $\mathbb{Z}_l^*$, the random choice of $P \in G_1^*$, and the random bits of $\mathcal{A}$. We assume that BDHP is hard, in other words, there is no polynomial time algorithm to solve BDHP with non-negligible probability.

3. An authenticated key agreement protocol

In this section, we first review the following previous ID based key agreement protocols and then propose a new ID based authenticated key agreement protocol using the pairings.

3.1. Previous ID based key agreement protocols

The following schemes own the same system initialization but different public key/private key issuing processes and different key agreement protocols.

(1) Smart's ID based AK protocol [18]
(2) Chen-Kudla's ID based AK protocol [5]
(3) Modification of Nalla's ID based AK protocol
(4) Choie-Jeong-Lee's ID based AK protocol [6]

Initialization : Let G_1 and G_2 be two groups of prime order l, where G_1 is an additive group and G_2 is a multiplicative group. The discrete logarithm problems (DLP) in both G_1 and G_2 are assumed to be hard. Let P be a generator of G_1, and $H : \{0,1\}^* \to \mathbb{Z}_l^*$ be a cryptographic hash function. The key generation center (KGC) chooses a random number $s \in \mathbb{Z}_l^*$ and set $P_{pub} = sP$. The center publishes system parameters $\texttt{Params} =< G_1, G_2, l, e, P, P_{pub}, H >$, and keep s as the master key, which is known only by itself.

In addition to the system initialization, KGC performs the following private key issuing process.

Private key extraction: Let A and B be the two entities who are going to agree to some session keys. The identities of A and B are ID_A and ID_B, respectively. Their public keys and private keys are as follows: A's public key is $P_A = H(ID_A)$, and the private key is $S_A = sP_A$. B's public key is $P_B = H(ID_B)$, and the private key is $S_B = sP_B$. The pairs (P_ID, S_ID) for A and B serve as their static public/private key pairs.

(1) Smart's ID based AK protocol

1. A selects $a \underset{R}{\in} [1, l-1]$ and sends $T_A = aP$ to B.
2. B selects $b \underset{R}{\in} [1, l-1]$ and sends $T_B = bP$ to A.
3. A computes $k_{AB} = e(aP_B, P_{pub})e(S_A, T_B)$.
4. B computes $k_{BA} = e(bP_A, P_{pub})e(S_B, T_A)$.

The session key is then $K = kdf(k_{AB}) = kdf(k_{BA})$, where $kdf : G_2 \rightarrow \{0,1\}^*$ is a key derivation function. As described in [17], this protocol does not provide full forward secrecy since an adversary who learns S_A and S_B can compute all session keys established by A and B.

(2) Chen-Kudla's ID based AK protocol

In [5], Chen-Kudla suggested a two-pass AK protocol satisfying the KGC forward secrecy property.

1. A selects $a \underset{R}{\in} [1, l-1]$ and sends $W_A = aP_A$ and $U_A = aP$ to B.
2. B selects $b \underset{R}{\in} [1, l-1]$ and sends $W_B = bP_B$ and $U_B = bP$ to A.
3. A computes $k_{AB} = e(S_A, W_B + aP_B)$.
4. B computes $k_{BA} = e(W_A + bP_A, S_B)$.

The session key is then $K = kdf(k_{AB}, aU_B) = kdf(k_{BA}, bU_A)$, where $kdf : G_2 \times G_1 \rightarrow \{0,1\}^*$ is a key derivation function. This protocol satisfies desirable security attributes, but is less efficient on the message bandwidth since two points need to be distributed by each user.

(3) Modification of Nalla's ID based AK protocol

In [14], Nalla proposed a tripartite key agreement protocol for ID based systems. The proposed scheme utilized the signature method to overcome the vulnerability of man-in-the-middle attack. If we consider one entity as KGC in the tripartite ID based protocol, it can be converted to the following AK protocol between two entities A and B. Apart from the initial system setting, a hash function $H_2 : G_1 \rightarrow \mathbb{Z}_l^*$ is required.

1. A selects $a \underset{R}{\in} [1, l-1]$, and sends $U_A = aP$ and $V_A = a^{-1}H_2(U_A)S_A$.
2. B selects $b \underset{R}{\in} [1, l-1]$, and sends $U_B = bP$ and $V_B = b^{-1}H_2(U_B)S_B$.
3. A verifies that $e(U_B, V_B) = e(P_{pub}, H_2(U_B)P_B)$. If this check fails, then A terminates the protocol run with failure. Otherwise, A compute $k_{AB} = e(U_B, aP_{pub}) = e(P, P)^{abs}$.
4. B verifies that $e(U_A, V_A) = e(P_{pub}, H_2(U_A)P_A)$. If this check fails, then A terminates the protocol run with failure. Otherwise, B computes $k_{BA} = e(U_A, bP_{pub}) = e(P, P)^{abs}$.

The session key is then $K = kdf(k_{AB}) = kdf(k_{BA})$, where $kdf : G_2 \rightarrow \{0,1\}^*$ is a key derivation function. The verification process ensures the authenticity of the received messages. Comparing with Chen-Kudla's, this protocol is less efficient in term of computations.

(4) Choie-Jeong-Lee's ID based AK protocol

In [6], Choie *et al.* proposed two ID based AK protocols. The one is a modification of variation of Nalla's and the other one is a modification of Smart's protocol. We now review their protocols.

- CJL protocol 1

1. A selects $a \underset{R}{\in} [1, l-1]$, and sends $U_A = aP_{pub}$ and $V_A = aS_A$ to B.
2. B selects $b \underset{R}{\in} [1, l-1]$, and sends $U_B = bP_{pub}$ and $V_B = bS_B$ to A.
3. A verifies that $e(V_B, P) = e(P_B, U_B)$ and computes $k_{AB} = aU_B = abP_{pub}$.
4. B verifies that $e(V_A, P) = e(P_A, U_A)$ and computes $k_{BA} = bU_A = abP_{pub}$.

The session key is $K = kdf(k_{AB}, P_A, P_B) = kdf(k_{BA}, P_A, P_B)$, where $kdf : G_2 \times G_1 \times G_1 \to \{0,1\}^*$ is a key derivation function.

- CJL protocol 2

1. A selects $a \underset{R}{\in} [1, l-1]$, and sends $T_A = aP$ to B.
2. B selects $b \underset{R}{\in} [1, l-1]$, and sends $T_B = bP$ to A.
3. A computes $h = H_2(aT_B)$ and $k_{AB} = e(haP_B, P_{pub})e(S_A, hT_B)$.
4. B computes $h = H_2(bT_A)$ and $k_{BA} = e(hbP_A, P_{pub})e(S_B, hT_A)$.

The session key is $K = kdf(k_{AB}, P_A, P_B) = kdf(k_{BA}, P_A, P_B)$. Comparing with CJL protocol 1, CJL protocol 2 has the minimal message bandwidth.

3.2. A new ID based key agreement protocol

Suppose two users A and B want to share a common secret. A and B have static private keys $S_A = sP_A$ and $S_B = sP_B$ obtained from KGC. Since the initialization and the private key extraction step is same as that of Section 3.1, our scheme can be simplified as follows.

Let $kdf : G_2 \times G_1 \times G_1 \to \{0,1\}^*$ be a key derivation function which can be readily found in a number of standard documents. A and B generate ephemeral private keys a and b, respectively. The corresponding ephemeral public keys are (V_A, W_A) and (V_B, W_B) where $V_A = aP_B$, $W_A = aS_A$, $V_B = bP_A$, $W_B = bS_B$. These are the data flow between A and B.

$$\begin{array}{rcl} A & \Longrightarrow & B : (V_A, W_A) \\ B & \Longrightarrow & A : (V_B, W_B) \end{array}$$

User A computes $k_A = e(aP_A + V_B, W_B)^a$. User B computes $k_B = e(bP_B + V_A, W_A)^b$. Then the shared common secret between A and B is $K = kdf(k_A, P_A, P_B) = kdf(k_B, P_A, P_B) = kdf(e(P_A, P_B)^{(a+b)abs}, P_A, P_B)$

3.3. Efficiency and security

Each party has three elliptic curve point multiplications, one exponentiation and one evaluation of the pairing map. We compare the number of computations of our protocol with the previous key agreement protocols [18, 5, 14, 6] (Refer to the table 1).

Our scheme is mutual implicit key authenticated and also role symmetric since each party has the same operation to obtain the common session key. We heuristically discuss the security of our protocol.

(i) SECURITY FROM THE MAN-IN-THE-MIDDLE ATTACK : Let E be an adversary who wants to share a secret key with an entity A as an entity B without the knowledge of S_B. E generates $V_E = a'P_A$ and $W_E = a''P_B$ for random numbers a', a'' and send them to A. Then A computes $k_{AE} = e(aP_A + V_E, W_E)^a = e(P_A, P_B)^{(a+a')aa''}$. But E computes $k_{EA} = e(a'P_B + V_A, W_A)^{a'} = e(P_B, P_A)^{(a+a')aa's}$. Therefore E is not able to share the common key with A. Similarly for B. This implies the proposed scheme is secure from the man in the middle attack.

(ii) KNOWN KEY SECURITY : From the randomness of a and b in our protocol, the session keys in different key agreements are independent of each other. The knowledge of the previous session keys does not help an adversary to derive any future session key. Hence, our protocol provides known-key security.

(iii) FORWARD SECRECY

(a) Even if a long term private key S_A(or S_B) of our protocol is compromised, the data protected with a previous session key ${k_A}'$ (or ${k_B}'$) is still secure because the derivation of ${k_A}'$(or ${k_B}'$) requires the knowledge of previous random values a'(or b'). Therefore, our protocol has the property of (perfect) forward secrecy.

(b) Compromising of the master key s of KGC(TA) does not allow an adversary to compute the session key if the elliptic curve discrete logarithm problem (For given $W_A = aS_A, S_A$, find a.) is intractable. Thus our protocol also provides the TA's forward secrecy.

(iv) KEY-COMPROMISE IMPERSONATION : Suppose that an adversary who knows A's long-term private key S_A wants to masquerade B to A. First, she chooses a random value b', computes ${V_E}' = b'P_A$, ${W_E}' = b'{S_B}'$ and sends it to A. Here ${S_B}' = b''P_B$ for a random b'' since she does not know S_B. However she can not compute the session key since she does not know A's ephemeral private key a.

(v) UNKNOWN KEY SHARE : According to [1], including the identities of the sender and the intended receiver in the key derivation function can prevent potential unknown key-share attacks.

Table 1. Comparison of protocols

Protocol	Weak attributes	# Key components	# Pairings	# Scalar multiplications
Smart [18]	Forward secrecy	1	2	2
Chen-Kudla [5]	none	2	1	4
Modification of Nalla	none	2	3	4
CJL 1 [6]	none	2	2	3
CJL 2 [6]	none	1	2	4
Our protocol*	none	2	1	3

* One multiplication over a finite field is required additionally for our protocol.

3.4. Key confirmation process

We describe the AKC protocol which can be derived from an AK protocol by adding the MACs (Message Authentication Code) of the flow number, identities and the ephemeral public keys. Here, MACs are used to provide key confirmation, and d_1, d_2 are two independent key derivation functions. These two functions could be different since they are often with different inputs from different groups and may be required to produce outputs of different forms. Let $FK = d_1(K)$, $FK' = d_2(K)$.

$$\begin{array}{lll}
 & \text{User A} & \qquad\qquad \text{User B} \\
 & V_A, W_A \qquad \longrightarrow & \\
 & \qquad\qquad \longleftarrow V_B, W_B, MAC_{FK'}(2, A, B, V_A, W_A, V_B, W_B) & \\
MAC_{FK'}(3, A, B, V_A, W_A, V_B, W_B) \longrightarrow & &
\end{array}$$

If the protocol succeeds, A and B share the session key, $FK = d_1(K)$. The method used here is well known in [11].

4. Conclusions

We have presented an ID based authenticated key agreement protocol based on pairings and compared the efficiency and security with the two pass authenticated key agreement given by [18, 5, 14, 6]. We discussed the security properties of our protocol. It is also easy to add a key confirmation property to our protocol using message authentication code just as with MQV and the three pass AKC protocol in [18].

References

[1] S. Blake-Wilson and A. Menezes, *Unknown key-share attacks on the station-to-station (STS) protocol*, in Proceedings of Second International Workshop on Practice and Theory in Public Key Cryptography, PKC'99, LNCS 1560, pp. 154–170, 1999.

[2] D. Boneh and M. Franklin, *Identity-based encryption from the Weil pairing*, Advances in Cryptology, Crypto 2001, Springer-Verlag, 2001.

[3] D. Boneh, B. Lynn, and H. Shacham, *Short signatures from the Weil pairing,* Advances in cryptology-Asiacrypt 2001, LNCS 2248, pp. 514-532, Springer-Verlag 2001.

[4] J. Cha and J. Cheon, *An ID-based signature from Gap-Diffie-Hellman Groups*, Proc. of PKC 2003, Lecture Notes in Computer Science, Vol. 2567, pp. 18–30, 2003.

[5] L. Chen, C. Kudla, *Identity based authenticated key agreement protocols from pairings*, Hewlett-Packard Laboratories technical reports HPL-2003-25, 2003.

[6] Y. Choie, E. Jeong, and E. Lee, *Efficient identity-based authenticated key agreement protocol from pairings*, Appl. Math. Comput. **162** (2005), no. 1, 179–188.

[7] W. Diffie and M. Hellman, *New directions in cryptography*, IEEE Trans. Inform. Theory **IT-2** (1976), no. 6, 644–654.

[8] G. Frey and H. Ruck, *A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves*, Math. Comp. **62** (1994), 865–874.

[9] F. Hess, *Efficient identity based signature scheme based on pairings*, Proceedings of the Workshop Selected Areas in Cryptology, SAC, Aug. 2002.

[10] A. Joux, *A one round protocol for tripartite Diffie-Hellman*, In W. Bosma, editor, Proceedings of Algorithmic Number Theory Symposium-ANTS IV, volume 1838 of Lecture Notes in Computer Science, pp. 385-394. Springer Verlag, 2000.

[11] L. Law, A. Menezes, M. Qu, J. Solinas, and S. Vanstone, *An efficient protocol for authenticated key agreement*, Design, Codes and Cryptography **28** (2003), 119–134.

[12] T. Matsumoto, Y. Takashima, and H. Imai, *On seeking smart public-key distribution systems,* The Transactions of the IECE of Japan, E69, pp. 99-106, 1986.

[13] A. Menezes, T. Okamoto, and S. Vanstone, *Reducing elliptic curve logarithms to logarithms in a finite field*, IEEE Trans. Inform. Theory **139** (1993), 1639–1646.

[14] D. Nalla, *ID-based tripartite key agreement with signatures*, Cryptology ePrint Archive, Report 2003/144, available at http://eprint.iacr.org/2003/144.

[15] K. G. Paterson, *ID-based signature from pairings on elliptic curves*, Electronics Letters **38** (2002), no. 18, 1025–1026.

[16] A. Shamir, *Identity-based cryptosystems and signature schemes*, Advances in Cryptology-Crypto 84, LNCS 196, pp.47-53, Springer-Verlag, 1984.

[17] K. Shim, *Efficient ID-based authenticated key agreement protocol based on Weil pairing*, Electronics Letters **39** (2003), no. 8, 653–654.

[18] N. P. Smart, *An identity based authentication key agreement protocol based on the Weil pairing*, Electronic letters **38** (2002), no. 13, 630–632.

Hyang-Sook Lee and Young-Ran Lee
Department of Mathematics
Ewha Womans University
Seoul 120-750, Korea
E-mail: hsl@ewha.ac.kr
iris92@ewha.ac.kr