ID-based tripartite key agreement with signatures

[1]Divya Nalla
AILab, Dept of Computer/Info. Sciences, University of Hyderabad,
Gachibowli, Hyderabad, 500046, India
divyanalla@yahoo.com

Abstract : This paper proposes a new identity based tripartite key agreement protocol which is more efficient than the existing ID-based tripartite protocol. This protocol is based on the Joux's protocol for key agreement, and introduces signature along with key agreement to overcome man-in-the-middle attacks and to provide authentication. The new protocol resists existential forgeries against adaptively chosen message attacks under the random oracle model.

Key Words: Key Agreement, tripartite, elliptic curves, Weil pairing, cryptography, Identity based, ID-based, Diffie-Hellman, Key Agreement Protocols, Signature, ID-based signature.

1. Introduction

The first modern protocol for key agreement was the Diffie-Hellman protocol given in the seminal paper in 1976 [DH76]. Diffie-Hellman key agreement provided the first practical solution to the key agreement problem, allowing two parties never having met in advance or shared keying material, to establish a shared secret key by exchanging messages over an open channel. The security rests in the intractability of the Diffie-Hellman problem and the related problem of computing discrete logarithms.

The concept of Identity-based (ID-based) systems was first proposed by Shamir in 1984 [AS84]. According to him, in an identity-based system, the public keys of the users are their identities itself. Joux [AJ00] proposed a tripartite generalisation of the Diffie-Hellman protocol using bilinear pairings. But this protocol suffered the man-in-the-middle attack just like the basic DH protocol. Joux's protocol has been modified by Paterson et al [SAK02] to provide authentication by including certificates. Paterson [KGP02] proposed Identity-based signatures which can be used for this purpose. This paper proposes a one-round ID-based tripartite key agreement protocol with signature to provide authentication.

Section 2 discusses the mathematical definitions and preliminaries required for the new protocol. Joux's tripartite protocol and its improvements are explained in section 3. Section 4 discusses the signature scheme by Paterson. The new protocol for identity-based tripartite key agreement is proposed and analysed in section 5. Section 6 concludes the paper.

2. Preliminaries

This section discusses the Weil pairing definition, and the bilinear Diffie-Hellman problem which forms the basis for the new protocol. This section also describes the initial settings for an Identity-based (ID-based) system.

2.1. The Weil Pairing

Let G_1 be an additive group of order prime q and G_2 be a multiplicative group of the same order. The modified Weil pairing [NPS01] is a map $\hat{e}: G_1 \times G_1 \rightarrow G_2$ which satisfies the following properties:

---

[1] The author is a Research Scholar in the Dept. of CIS, University of Hyderabad.

1. Bilinear

   $\hat{e}(P_1 + P_2, Q) = \hat{e}(P_1, Q).\hat{e}(P_2, Q)$

   $\hat{e}(P, Q_1 + Q_2) = \hat{e}(P, Q_1).\hat{e}(P, Q_2)$

   i.e., $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ where $a, b \in \mathbb{Z}_q^*$
2. Non-Degenerate

   There exists a $P \in G_1$ such that $\hat{e}(P, P) \neq 1$
3. Computable:

   One can compute $\hat{e}(P, Q)$ in polynomial time.

The existence of such a pairing is assumed. Typically, G_1 will be a subgroup of the group of points on a elliptic curve over a finite field, G_2 will be a subgroup of the multiplicative group of a related finite field. The non-degeneracy defined here does not hold for the standard Weil Pairing $e(P, Q)$. A more comprehensive description is provided in [BF01].

2.2. Diffie-Hellman Problem

The security of the identity-based system in this paper depends on a variant of Computational Diffie-Hellman assumption called the Bilinear Diffie-Hellman assumption (also called the Weil Diffie-Hellman Assumption) [BF01]. Variations of the Diffie-Hellman problem can be listed as follows:

Let G_1, G_2 be two groups of prime order q (G_1 is an additive group and G_2 is a multiplicative group). Let P be a generator of G_1.

A *Diffie-Hellman tuple* in G_1 is $(P, xP, yP, zP) \in G_1$ for some $x, y, z \in \mathbb{Z}_q^*$, satisfying $z \equiv xy \bmod q$.

Given any three elements from the four elements in the diffie-hellman tuple, computing the fourth element is the *computational diffie-hellman problem* (CDHP).

Given $P, xP, yP, zP \in G_1$, deciding if it is a valid diffie-hellman tuple is the *decisional diffie-hellman problem* (DDHP).

The *bilinear diffie-hellman problem* (BDHP) in $\langle G_1, G_2, \hat{e} \rangle$ is that given $(P, xP, yP, zP) \in G_1$ for some $x, y, z \in \mathbb{Z}_q^*$, compute $\hat{e}(P, P)^{xyz} \in G_2$

On a finite field, all these classes of problems are known to be difficult. But on certain elliptic curves, DDHP is easy though CDHP is intractable. The assumption that the DDHP is easy and the CDHP is intractable, is called the Gap diffie-hellman assumption [CC02].

2.3 System Settings

The key generation centre (KGC) chooses a secret key $s \in \mathbb{Z}_q^*$, produces a random $P \in G_1$ and computes $P_{KGC} = [s]\, P$. Then the KGC publishes (P, P_{KGC}).

When a user with identity *ID* wishes to obtain a public/private key pair, the public key is given by

$$Q_{ID} = H_1(ID) \text{ where } H_1 : \{0,1\}^* \rightarrow G_1 \text{ is a hash function.}$$

And the KGC computes the private key

$$S_{ID} = [s]\, Q_{ID}$$

This calculation can be performed using multiple key generation centres, each generating a part of the private key. These are combined by the user to obtain his / her private key.

Let $V : G_2 \rightarrow \{0,1\}^*$ be the key derivation function [NPS01].

3. Joux's protocol and it's improvements

Joux [AJ01] introduced a simple one round tripartite key agreement protocol using Weil pairing. In Joux's protocol, a, b, $c \in \mathbb{Z}_q^*$ are selected uniformly at random by A, B, and C respectively.

Protocol messages:

$$A \to B, C : aP$$
$$B \to A, C : bP$$
$$C \to A, B : cP$$

In this protocol, the ordering of the protocol messages is unimportant and any of the three entities can initiate the protocol. Once the communication is over, A, B, and C compute their keys k_A, k_B, k_C.

$k_A = \hat{e}\ (bP, cP)^a$; $k_B = \hat{e}\ (aP, cP)^b$; $k_C = \hat{e}\ (aP, bP)^c$

$$k_A = k_B = k_C = k_{ABC} = \hat{e}\ (P, P)^{abc}$$

k_{ABC} is the common session key. Success of this protocol lies in the hardness of the Bilinear Diffie-Hellman problem (BDHP).

Just like the unauthenticated two party Diffie-Hellman protocol, Joux's protocol is subject to a classic man-in-the-middle attack. Including authentication in the protocol can thwart this attack. Al-Riyami and Paterson [SAK02] proposed a few improvements to the Joux's protocol. They are called Tripartite Authenticated Key agreement (TAK) Protocols.

A Certification Authority (CA) is used in the initial set up stage to provide certificates, which binds user's identities to long-term Keys. The certificate for A will be of the form:

$$Cert_A = (I_A \parallel \mu_A \parallel P \parallel S_{CA}(\, I_A \parallel \mu_A \parallel P)).$$

Where I_A denotes the identity of A, $\parallel$ denotes the concatenation of data items, S_{CA} denotes the CA's signature. x, y, and z are A, B and C's long term private Keys, and $\mu_A = xP$, $\mu_B = yP$, $\mu_C = zP$ are the long term public Keys of A, B and C. Short-term Keys a, b, $c \in Z_q^*$ are selected uniformly at random by A, B, and C respectively.

Protocol messages:

$$A \to B, C : aP$$
$$B \to A, C : bP$$
$$C \to A, B : cP$$

TAK Key generation:

Four types of key generation are given below. The keys computed by the entities are given below

Type 1

$$K_A = \hat{e}(bP, cP)^a.\hat{e}(yP, zP)^x$$
$$K_B = \hat{e}(aP, cP)^b.\hat{e}(xP, zP)^y$$
$$K_C = \hat{e}(aP, bP)^c.\hat{e}(xP, yP)^z$$
$$K_{ABC} = K_A = K_B = K_C = \hat{e}(P, P)^{abc+xyz}$$

Type 2

$K_A = \hat{e}(bP, zP)^a \hat{e}(yP, cP)^a \hat{e}(bP, cP)^x$

$K_B = \hat{e}(aP, zP)^b \hat{e}(xP, cP)^b \hat{e}(aP, cP)^y$

$K_C = \hat{e}(aP, yP)^c \hat{e}(xP, bP)^c \hat{e}(aP, bP)^z$

$K_{ABC} = K_A = K_B = K_C = \hat{e}(P, P)^{(ab)z+(ac)y+(bc)x}$

Type 3

$K_A = \hat{e}(yP, cP)^a \hat{e}(bP, zP)^a \hat{e}(yP, zP)^x$

$K_B = \hat{e}(aP, zP)^b \hat{e}(xP, cP)^b \hat{e}(xP, zP)^y$

$K_C = \hat{e}(aP, yP)^c \hat{e}(xP, bP)^c \hat{e}(xP, yP)^z$

$K_{ABC} = K_A = K_B = K_C = \hat{e}(P, P)^{(xy)c+(xz)b+(yz)c}$

Type 4

$K_A = \hat{e}(bP + H(bP \parallel yP)yP, cP + H(cP \parallel zP)zP)^{a+H(aP\parallel xP)x}$

$K_B = \hat{e}(aP + H(aP \parallel xP)xP, cP + H(cP \parallel zP)zP)^{b+H(bP\parallel yP)y}$

$K_C = \hat{e}(aP + H(aP \parallel xP)xP, bP + H(bP \parallel yP)yP)^{c+H(cP\parallel zP)z}$

$K_{ABC} = K_A = K_B = K_C = \hat{e}(P, P)^{(a+H(aP\parallel xP)x)(b+H(bP\parallel yP)y)(c+H(cP\parallel zP)z)}$

For a single round protocol, TAK-4 is the most secure, followed by TAK-2.

4. Identity-based signature scheme by Paterson [KGP02]

The system settings are as described in section 2.3. Apart from these, a few hash functions are defined.

$$H_1 : \{0,1\}^* \to G_1 \qquad H_2 : \{0,1\}^* \to \mathbb{Z}_q \qquad H_3 : G_1 \to \mathbb{Z}_q .$$

The public key of a user is given by QID = H1(ID), and the secret key is SID = [s] QID , where $s \in \mathbb{Z}_q^*$ is the secret key of the key generation center.

To sign a message $M \in \{0,1\}^*$, (the user with identity ID) first chooses a random $k \in \mathbb{Z}_q^*$ and computes his signature on a message M as the pair $(R,S) \in G_1 \times G_1$, where $R = kP$; $S = k^{-1}(H_2(M)P + H_3(R)S_{ID})$.

Here k^{-1} is the inverse of k in $\mathbb{Z}_q^*$.

To verify a signature (U,V) on message M, the verifier computes $\hat{e}(U,V)$ and compares it to the value $\hat{e}(P,P)^{H_2(M)}.\hat{e}(P_{KGC}, Q_{ID})^{H_3(U)}$. The signature is accepted if these values in G_2 match.

If (R,S) is a valid signature on M, then we have

$$\begin{aligned}\hat{e}(R,S) &= \hat{e}(kP, k^{-1}(H_2(M)P + H_3(R)S_{ID})) \\ &= \hat{e}(P, H_2(M)P + H_3(R)S_{ID}) \\ &= \hat{e}(P,P)^{H_2(M)}.\hat{e}(P_{KGC}, H_3(R)Q_{ID}) \\ &= \hat{e}(P,P)^{H_2(M)}.\hat{e}(P_{KGC}, Q_{ID})^{H_3(R)}\end{aligned}$$

Thus, verifying a signature (U, V) for a message M, $\hat{e}(U,V) = \hat{e}(P,P)^{H_2(M)}\hat{e}(P_{KGC}, Q_{ID})^{H_3(U)}$ can be done by any user who receives the signature.

5. Tripartite ID-based key agreement with signatures

A new tripartite key agreement protocol for ID-based systems is proposed in this section. Some ID-based tripartite key agreement protocols proposed in [NR03] suffered passive attacks, and Joux's protocol [AJ00] suffered man-in-the-middle attack. To overcome these, it is found that including signature in Joux's protocol resulted in much simpler ID-based tripartite key agreement protocols. The proposed protocol is as follows.

The initial system settings are same as any ID-based system (described in section 2.3). Apart from these, a hash function $H : G_1 \to \mathbb{Z}_q^*$ us defined. The three participants *A*, *B*, and *C* select random values *a*, *b*, and *c* from $\mathbb{Z}_q^*$, and each one of them sends messages to the other two in a single round. The common key is computed using the received values after verifying the signature for authentication.

Protocol messages*:*

$$A \to B, C : U_A = aP; V_A = a^{-1}(H(U_A)S_A)$$
$$B \to A, C : U_B = bP; V_B = b^{-1}(H(U_B)S_B)$$
$$C \to A, B : U_C = cP; V_C = c^{-1}(H(U_C)S_C)$$

Verification and key computation*:*

A verifies

$$\hat{e}(U_B, V_B).\hat{e}(U_C, V_C) = \hat{e}(P_{KGC}, H(U_B)Q_B + H(U_C)Q_C)$$

and computes $k_A = \hat{e}(U_B, U_C)^a = \hat{e}(P, P)^{abc}$

B verifies

$$\hat{e}(U_A, V_A).\hat{e}(U_C, V_C) = \hat{e}(P_{KGC}, H(U_A)Q_A + H(U_C)Q_C)$$

and computes $k_B = \hat{e}(U_A, U_C)^b = \hat{e}(P, P)^{abc}$

C verifies

$$\hat{e}(U_B, V_B).\hat{e}(U_A, V_A) = \hat{e}(P_{KGC}, H(U_B)Q_B + H(U_A)Q_A)$$

and computes $k_C = \hat{e}(U_B, U_A)^c = \hat{e}(P, P)^{abc}$

This verification ensures the authenticity of the senders.

The common key is the value $K_{ABC} = V(k_A) = V(k_B) = V(k_C) = V(\hat{e}(P, P)^{abc})$ where V is a key derivation function defined as $V : G_2 \to \{0,1\}^*$ [NPS01].

5.1 Security Analysis

The session key computed is $K_{ABC} = V(\hat{e}(P, P)^{abc})$ which is dependent on the ephemeral private key *a*, *b*, and *c*. The values *aP*, *bP*, and *cP* are exchanged publicly to determine the session key. These three values are called the ephemeral public keys. The security of the protocol lies on the assumption of hardness of the bilinear Diffie-Hellman problem.

The authenticity of the ephemeral public values is achieved by sending the signature of the sender along with it. The idea of the signature scheme of Paterson [KGP02] is used in the protocol wherein the value V_{ID} (for *ID* = *A*, *B*, *C*) is computed using their static private key. V_{ID} can be considered as *ID* 's signature for the message *iP* (where *i* is a value randomly selected by user with identity *ID*). The authenticity of the protocol is based on the security of the following signature scheme:

Signing: Suppose that the message to be signed is *m* = *aP*, then the signature of *m* is computed to be $a^{-1}(H(m)S_{ID})$ where S_{ID} is the static private key of the signing entity.

Verification: After getting $m = aP$ and its signature $V = a^{-1}(H(m)S_{ID})$, the verifier accepts the signature if and only if the following equation holds.

$$\hat{e}(m,V) = \hat{e}(P_{KGC}, H(m)Q_{ID})$$

To show that the signature scheme is secure against existential forgery under an adaptively chosen message attack in the random oracle model, the proof is given similar to the one by Hess [FH02].

Suppose that there is a polynomial time probabilistic turing machine E which takes the message m and Q_{ID} as input, and output an existential forgery of signature from a user A with a non negligible probability. (Here H is assumed to be a random oracle). Then we show that there is another polynomial time algorithm E' which takes advantage of the turing machine E, and solves the diffie-hellman problem.

To show that the proposed signature scheme resists existential forgeries against adaptively chosen message attacks under the random oracle model and the gap diffie-hellman assumption, the following lemma known as the Forking lemma [PS96] [PS00] is needed.

Lemma: For a Signature scheme that belongs to the class of general signature schemes which produce a tuple (U, h, V) as a signature of a message m, where U takes random values, h is the hash value of (m,U), and V is determined by (m, U, h), suppose that $\mathcal{A}$ is an adversary machine that outputs a message m and a valid signature (U, h, V) for m with non-negligible probability ε and running time T. If the triples (U, h, V) can be simulated without knowing the secret key, with an indistinguishable distribution probability, then a replay of $\mathcal{A}$, where interactions with the signing oracle are simulated, produces two valid signatures (U, h, V) and (U, h', V') for a common message m such that $h = h'$, within time $T' < 23QT/\varepsilon$ and with probability $\varepsilon' > 1/9$, where Q is the number of signing queries made by $\mathcal{A}$.

Simulation of the extraction and signature oracles can be done as follows:

Extraction oracle queries: Given an identity ID', the extraction oracle computes a random $\lambda \in \mathbb{Z}_q^*$, $Q_{ID} = \lambda P$ and $S_{ID} = \lambda P_{KGC}$. Then it defines $H(ID) = Q_{ID}$ and returns S_{ID}. These values are stored in a hash value list so that the extraction oracle returns the same value when queried for the same ID again.

Signature oracle queries: For any given message m and identity ID, this oracle will produce a signature from the user with identity ID on the message m.
The signing oracle can be constructed as follows:

1. Choose random values $a, h \in \mathbb{Z}_q^*$
2. Compute $U = aP_{KGC}; V = a^{-1}hQ_{ID}$
3. Fail if hash value of U already exists in the hash value list. Otherwise record h as the hash value for U and output (U, h, V).

Since (U, P_{KGC}, V, hQ_{ID}) = (aP_{KGC}, P_{KGC}, $a^{-1}hQ_{ID}$, hQ_{ID}) is a valid Diffie-Hellman tuple, the output (U, h, V) is a valid signature. Also a and h are randomly chosen.

Thus, the tuple (U, h, V) can be simulated without knowing the secret key, with an indistinguishable distribution probability.

By forking lemma, the adversary $\mathcal{A}$ may obtain two forgeries of the same message m : (U, h, V) and (U, h', V'). Since the value of U is same in both,

$$\hat{e}(U, V - V') = \hat{e}(P_{KGC}, (h - h')Q_{ID})$$

$$\hat{e}(U, V - V') = \hat{e}(P_{KGC}, Q_{ID})^{(h-h')}$$

Since h and h' are random values output by the oracle, the value (h – h') is a random value, and hence $\hat{e}(P_{KGC}, Q_{ID})^{(h-h')}$.

(V – V') is output from the signing oracle, and the value $\hat{e}(P_{KGC}, Q_{ID})^{(h-h')}$ is a random value from G_2. Thus, the adversary is able to solve the following problem:

$\hat{e}(a, r) = c$ is solved for 'a' for a random $c \in G_2$ where r is a known value in G_1.

Solving $\hat{e}(a, r) = c$ is uniformly as hard as solving $\hat{e}(a, P) = c$ for a prescribed P, since G_1 is cyclic. That is, the adversary is able to compute the inverse of a pairing.
It is shown by Yacobi [YY02] that inverting the pairing $\hat{e}(\bullet, r)$ is at least as hard as solving the diffie-hellman problem in both G_1 and G_2.

Thus the adversary is able to solve an instance of the diffie-hellman problem within time T' and with probability ε'.

Summarising, we have the following theorem.
Theorem: The proposed signature scheme resists existential forgeries against adaptively chosen message attacks under the random oracle model. More precisely, if there is an algorithm breaking the signature scheme under adaptively chosen message attack within time T and with probability ε, then the computational diffie-hellman problem can be solved within time less than ($23T/\varepsilon$ + count) and with probability greater than $1/9 - \varepsilon''$, where Q is the upper bound of the number of signing queries made by $\mathcal{A}$ and ε'' is a negligible probability.

5.2 Security Attributes efficiency

Passive attacks are not possible in this protocol since there is a signature involved in it. V_i constitutes the signature for U_i (where i = A, B and C). Since the sender is required to send V_i computed using his/her long term secret, which is then verified by the receiver, man-in-the-middle attack is not possible. Two key compromise attack is also not possible since the key computed is a hash value. The protocol is immune to known key attacks and is also forward secure.

The total number of computations involved in signature verification and key computation are 4 weil pairings, 4 hash functions, 5 elliptic curve scalar multiplications, and 1 exponentiation for each entity. The following table shows the comparison of these computations to those in the tripartite protocol proposed by Zhang et al [ZLK02]. It can be concluded from the table that the new protocol proposed is more efficient in terms of computations. The number of messages communicated is also less in the new protocol.

	Weil Pairings	Scalar multiplications	Exponentiations	Hash functions
Zhang's protocol	8	6	8	3
Simplified version of Zhang's protocol	5	5	1	3
Proposed protocol	4	5	1	4

Table: Comparison of computations in Zhang et al's protocol and the new protocol.

6. Conclusions

A new Identity-based one round tripartite authenticated key agreement protocol is proposed in this paper. The key agreement part in the new protocol is similar to Joux's protocol, but it achieves authentication by introducing signatures along with the key agreement. It is also shown that the signature scheme in the new protocol resists existential forgeries against adaptively chosen message attacks under the random oracle model. This proof also proves the sigature scheme by Paterson [KGP02]. Comparing the computations to an existing ID-based tripartite key agreement protocol, the new protocol is found to be more efficient.

7. References

[AJ00] 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 , pages 385-394, Springer-Verlag, 2000.

[AS84] A. Shamir, Identity based cryptosystems and signature schemes. Advances in Cryptology – Proceedings of Crypto'84

[BF01] D. Boneh and M. Franklin. Identity-based encryption from the Weil Pairing. In Advances in Cryptology – CRYPTO 2001, Springer-Verlag LNCS 2139, 213-229, 2001.

[CC02] J. C. Cha and J. H. Cheon. *An Identity-based signature from Gap Diffie-Hellman groups*. Cryptology ePrint Archive, Report 2002/018, http://eprint.iacr.org/.

[DH76] W. Diffie and M. Hellman. New directions in cryptography. IEEE Trans. Info. Th., 22, 644-654, 1976.

[ERV01] Eric R. Verheul, Evidence that XTR is more secure than supersingular ECCs, In Advances in Cryptology – Eurocrypt 2001, Springer-Verlag, LNCS 2045, 195-210, 2001.

[FH02] F. Hess. Efficient Identity based signature schemes based on pairings. Proceedings of 9th workshop on selected areas in Cryptography – SAC 2002, Lecture notes in Computer Science, Springer-Verlag.

[KGP02] K.G.Paterson. ID-based signatures from pairings on elliptic curves. Cryptology eprint archive, Report 2002/004, available at http://eprint.iacr.org/

[NPS01] N.P. Smart. An Identity based authenticated Key Agreement protocol based on the Weil Pairing. Cryptology ePrint Archive, Report 2001/111, 2001. http://eprint.iacr.org/.

[NR03] Divya Nalla, K.C.Reddy, ID-based tripartite Authenticated Key Agreement Protocols from pairings, available at http://eprint.iacr.org/2002/004.

[PS96] D. Pointcheval and J. Stern, Security proofs for Signature Schemes, Proc. of Eurocrypt 96, Lecture Notes in Computer Sciences, Vol.1070, pp.387-398, Springer-Verlag, 1996.

[PS00] D. Pointcheval and J. Stern, Security Arguments for Digital Signatures and Blind Signatures, J. of Cryptology 13 (2000), 361-396.

[SAK02] Sattam S. Al-Riyami, Kenneth G. Paterson, Authenticated Three Party Key Agreement Protocols from Pairings, Information security group, Royal Holloway, University of London, March 2002.

[YY02] Y. Yacobi, *A Note on the Bilinear Diffie-Hellman Assumption*, Cryptology ePrint Archive, Report 2002/113

[ZLK02] Fangguo Zhang, Shengli Liu and Kwangjo Kim, ID-based one-round authenticated tripartite key agreement protocol with pairings, Cryptology eprint archive, Report 2002/122.