An Enhanced Certificateless Authenticated Key Agreement Protocol

Razieh Mokhtarnameh, Sin Ban Ho, Nithiapidary Muthuvelu

Faculty of Information Technology, Multimedia University, 63100, Cyberjaya, Malaysia

{razieh.mokhtarnameh06, sbho, nithiapidary}@mmu.edu.my

Abstract— **Authenticated key agreement protocol is used to share a secret key for encrypting data being transferred between two or more parties over a public network. An implementation of this protocol is the certificateless key agreement which utilizes the features of the identity-based public key cryptography and the traditional public key infrastructure. This implementation can produce multiple public keys for a corresponding private key. In this paper, an alternative key generation technique is proposed for certificateless public key cryptography in order to have one public key for one private key. This will improve the security features of the relevant key generation. Furthermore, the efficiency of the proposed protocol is presented in terms of computational operation. The comparison analysis shows that the proposed protocol conveys better efficiency with all the known security attributes compared to the existing protocols.**

Keywords— **Authenticated key agreement; key generation; certificateless public key cryptography; identity-based public key cryptography; efficiency; security**

I. INTRODUCTION

A key agreement protocol is used to allow two or more parties to establish a session key over open networks. Each party can encrypt any message such that only the parties sharing the secret session key can decrypt the message. Authenticated key agreement should not only be secure against passive adversaries who are eavesdropping communications between parties, but also active adversaries who impersonate one party to communicate with another party. The idea of key agreement protocols has been realized in Public-Key Infrastructure (PKI) [1], IDentity-based Public Key Cryptography (ID-PKC) [2], Certificate-Based Public Key Cryptography (CB-PKC) [3], and Certificateless Public Key Cryptography (CL-PKC) [4].

In a traditional PKI [1], the binding between the public and private keys, and verifying those keys are achieved through the use of a certificate. Shamir [2] first proposed ID-PKC in which the public key is generated from some publicly identifiable information, such as an entity's email address or hostname. The binding between the private key and the entity's identity data is managed by a trusted authority (called a Key Generation Center, KGC) [5].

PKI protocols experience a heavy certificate management load while ID-PKC requires all the participants to trust an authority exuberantly (key escrow). A malicious KGC can compute the session keys of the participating entities. Thus, fully trusting an authority is a very strong assumption especially over open networks. Hence, ID-PKC seems more suited for smaller networks or closed groups.

CL-PKC combines the advantages of the ID-PKC and the traditional PKI. In CL-PKC, first, an identity-dependent partial private key is received from a KGC. Then, the entity computes its private key using partial private key and a secret known only to the entity. The entity generates a public key which matches the private key too. As a result, the trust is formed in an implicit way and reduced on KGC. Thus, CL-PKC is more suitable for open networks especially in distributed environments.

The work is focused on efficient key agreement in an open network. Current set up of key generation in CL-PKC allows an entity to create more than one public key for a partial private key. This can be pleasing in some applications, but undesirable in security-critical applications [4]. The proposed protocol is motivated by ID-PKC, provides a simple binding technique which ensures that entities can create only one public key for a corresponding private key. In addition, it reduces the degree of trust that the entities need to have on the KGC.

The rest of this paper is organized as follows: The related work on certificateless key agreement protocols is presented in Section II. Section III delivers the technical background preliminaries of the protocols related to key agreement. We present the structure of the proposed certificateless authenticated key agreement protocol in Section IV. The security level and complexity of the proposed protocol are described in subsections of Section V. Finally, Section VI concludes this paper by suggesting future work.

II. RELATED WORK

The first certificateless key agreement scheme was proposed by Al-Riyami and Paterson [4] as a side note to their certificateless encryption scheme. The scheme requires each party to compute four bilinear pairings which are computationally intensive. Mandt and Tan [6] proposed a similar protocol which relies on the difficulty of the Bilinear Diffie-Hellman problem. It is more efficient than the former protocol as it requires only two bilinear pairing computations. However, it does not provide key-compromise impersonation

and known session specific temporary information security attributes.

The certificateless key agreement schemes were further improved by multiple researchers such as Xia et al. [7], Wang et al. [8], Shao [9], and Shi and Li [10]. Swanson [11] analyzed all these certificateless schemes and showed some generic attacks that can break the notions of security attributes claimed by the respective authors. Lippold, Boyd, and Nieto [12] proposed a one round protocol that withstands all of Swanson's attacks, motivated by Mandt and Tan's protocol [6] and Xia et al [7]. However, the protocol [12] involves three exponentiations and five pairing computations.

In a recent work, Wang et al. [13] presented the first certificateless authenticated key agreement protocol for grid computing based on the Diffie-Hellman key agreement protocol and CL-PKC. However, Hou and Xu [14] found the scheme cannot withstand key compromise impersonation attack and key replicating attack. Hence, Hou and Xu [15] proposed another certificateless two-party authenticated key agreement protocol based on the certificateless encryption scheme originated from Sun and Zhang [16]. Furthermore, they proposed another protocol [17] which is based on the certificateless encryption scheme suggested by Libert and Quisquater [18].

Hou and Xu achieve the most known security attributes in both the protocols, [15] [17]. In 2010, Zhang et al. [19] proposed certificateless two-party authenticated key agreement protocol which is provably secure and efficient. The protocol involves one pairing operation and five multiplications.

III. PRELIMINARIES

A. Security Attributes of Key Agreement Protocols

The followings summarize the definitions of the security attributes of key agreement protocols as adopted from [20]:

1) Known-key secrecy: Each run of the protocol should result in a unique session key. Key generated in one protocol round is independent and should not be exposed if other session keys are compromised.

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

3) Perfect forward secrecy: If the long-term private keys of all the entities are compromised, the secrecy of previously established session keys should not be affected.

4) KGC forward secrecy: If the master key of KGC is corrupted, the security of session keys previously established should not be compromised by any entity.

5) Key-compromise impersonation: When entity A's long-term private key is compromised, the adversary should not be able to share a session key with A by acting as another entity B. For a more detailed discussion on this security attribute, see Section V-A-4.

6) Unknown key-share resilience: Entity A should not share a key with entity C when in fact A thinks that it is sharing the key with entity B.

7) No key control: The session key should be determined jointly by both entities. None of the entities can control the key alone.

8) Known session-specific temporary information security: The compromise of randomized input used in a protocol run should not reveal the agreed session keys.

B. Bilinear Groups

Let G_1 be a cyclic additive group of prime order q and G_2 be a cyclic multiplicative group of prime order q, P is a generator of G_1; assume that the Discrete Logarithm Problem (DLP) is hard in both G_1 and G_2. DLP is explained in the following subsection. An admissible pairing e is a bilinear map e: $G_1 \times G_1 \rightarrow G_2$, which satisfies the following three properties:

1) Bilinear: for $\forall P, Q \in G_1$ and, $\forall a, b \in Z_q^*$, we have $e(aP,bQ) = e(P,Q)^{ab}$;

2) Non-degenerate: $e(P, P) \neq 1$;

3) Computable: The map e is efficiently computable. The Weil [21] and modified Tate [22] pairings on elliptic curves can be used to construct such bilinear maps.

C. Some Computational Assumptions

The security of the proposed protocol relies on the standard Computational Diffie-Hellman (CDH) and Bilinear Diffie-Hellman (BDH) problem assumptions which are understood to be computed with minor probability.

1) Discrete Logarithm Problem (DLP): Given $P, Q \in G_1$, find $n \in Z_q^*$ such that $P = nQ$ whenever such n exists.

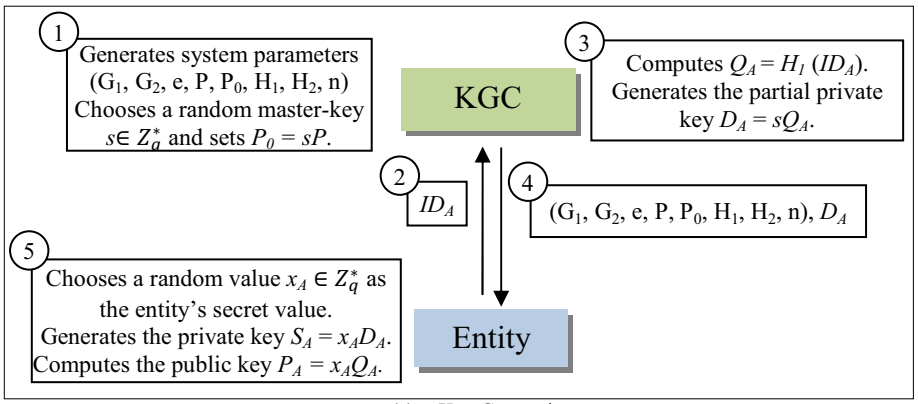
2) Computational Diffie–Hellman Problem (CDHP): Given a tuple $(P, aP, bP) \in G_1$ for a, $b \in Z_q^*$, find the element abP.

3) Bilinear Diffie–Hellman Problem (BDHP): Given $(P, xP, yP, zP) \in G_1$ for some x, y, z chosen at random from Z_q^*, compute $e(P, P)^{xyz} \in G_2$.

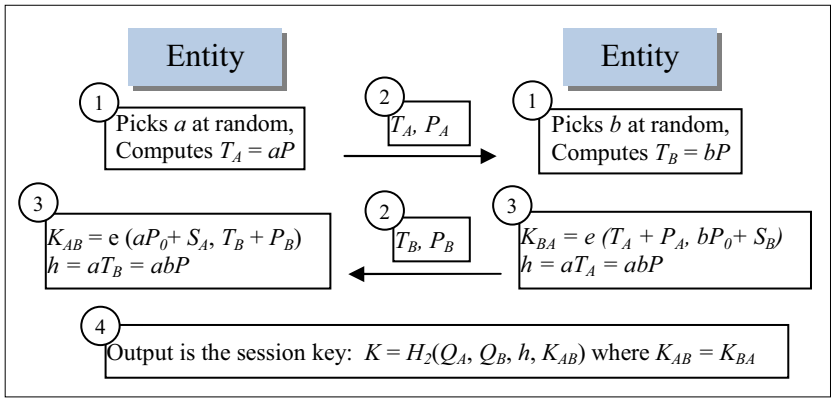
IV. PROPOSED PROTOCOL

The target is to achieve higher degree of security by creating one public key for a corresponding private key using the features of ID-PKC [23]. The relevant proposed algorithms are presented in this section.

Figure 1 shows the process flows of the proposed key generation and key agreement involved in CL-PKC. KGC executes Setup algorithm to generate master-key and system parameters. Then, it runs Partial-Private-Key-Extract algorithm to extract the partial private key for each entity. Every entity chooses a secret value and computes its public and private key. Subsequently, two entities run key agreement algorithm online in order to share a session key.



(a) Key Generation



(b) Key Agreement

Figure 1. Certificateless Key Generation and Agreement

Setup and Partial-Private-Key-Extract (Fig. 1(a))

1) KGC performs the following steps during the Setup process:
   a) Select a cyclic additive group G_1 of prime order q, a cyclic multiplicative group G_2 of the same order, a generator P of G_1, and a bilinear map e: $G_1 \times G_1 \rightarrow G_2$.
   b) Choose a random master-key, $s \in Z_q^*$ and set $P_0 = sP$.
   c) Choose cryptographic hash functions, H_1: $\{0, 1\}^* \rightarrow G_1$, H_2: $G_2 \rightarrow \{0, 1\}^n$.
2) Entity A sends its identity ID_A to KGC.
3) KGC generates the partial private key for entity A using the following steps:
   a) Compute $Q_A = H_1 (ID_A)$.
   b) Generate the partial private key $D_A = sQ_A$.
4) The system parameters (G_1, G_2, e, P, P_0, H_1, H_2, n) are published while the master-key $s \in Z_q^*$ is kept in KGC.
5) Entity A executes:
   a) Set-Secret-Value: choose a random value, $x_A \in Z_q^*$ as the entity's secret value.
   b) Set-Private-Key: generate the private key, $S_A = x_A D_A$.
   c) Set-Public-Key: compute the public key, $P_A = x_A Q_A$.

Key-Agreement (Fig. 1(b))

Assume that an entity A with identity ID_A has a longterm private key $S_A = x_A D_A$ and public key $P_A = x_A Q_A$, and an entity B with identity ID_B has private key $S_B = x_B D_B$ and public key $P_B = x_B Q_B$. A and B participate in the key agreement protocol as follows:

1) A chooses a short-term private key, $a \in Z_q^*$ randomly and computes $T_A = aP$. B chooses a short-term private key, $b \in Z_q^*$ randomly and computes $T_B = bP$.
2) A sends (P_A, T_A) to B. B sends (P_B, T_B) to A.
3) A computes $h = aT_B$ and $K_{AB} = e\ (T_A + P_A,\ bP_0 + S_B)$. B computes $h = bT_A$, and $K_{BA} = e\ (aP_0 + S_A,\ T_B + P_B)$.
4) A and B have the same shared secret $K_{AB} = K_{BA} = e(P, P)^{abs}\ e(P, Q_B)^{asx_B}\ e(Q_A, P)^{bsx_A}\ e(Q_A, Q_B)^{sx_A x_B}$, The session key is $K = H_2\ (Q_A, Q_B,\ h,\ K_{AB})$.

V. Discussion

In this section, the performance of the proposed protocol is analyzed in terms of security attributes and algorithm complexity.

A. Security Attributes

1) Known-key secrecy: A and B choose random $a \in Z_q^*$ and $b \in Z_q^*$ respectively in each protocol run; they will have distinct session key in each run. Thus, compromising the secret keys will not affect the next session key to be generated.

2) Forward secrecy: Even if the adversary knows the long-term private keys of A and B, the adversary still needs to compute h from T_A and T_B which is a CDH problem.

Therefore, compromising the long-term private keys of all entities will not reveal previously established session keys. As a result, the proposed protocol achieves perfect forward secrecy.

3) KGC forward secrecy: CL-PKC based schemes do not have key escrow problem. If an adversary has the KGC's master private key, s, the previously established session keys will not be exposed. Although the adversary may generate the partial private key, both the short-term and long-term private keys of an entity are needed in order to compute the session key.

4) Key-compromise impersonation: Assume that an adversary knows the private key of A, S_A, and impersonates B to share the session key with A. The adversary will have the knowledge on S_A, aP, and b, however, he would not be able to compute $e(P, Q_B)^{asx_B}$ as S_B is unknown. Another option is to compute asx_BP which is a CDH problem.

5) Unknown key-share resilience: As Q_A and Q_B are used for computing the session key, each entity knows who he shares the key with.

6) No key control: Minimum two entities collaborate together to generate a session key using their random short-term private keys. However, key control can be imperfect when A sends its (P_A, T_A) to B, but B does not send its (P_B, T_B) to A. This particular security attribute can be supported externally using special error checking or troubleshooting methods in the protocols.

7) Known session-specific temporary information security: Even the adversary compromises the short-term private keys of a session; he will not be able to compute the session key as the long-term private keys are unknown to him.

8) Passive attack: Assume that the adversary observes the messages (P_A, T_A, T_B, P_B) transferred between the entities and he knows the master key of KGC, s. The adversary will not be able to compute the session key as he needs to calculate abP from aP and bP. This is a CDH problem.

9) Man-in-the-middle attack (active attack): If an adversary is planning to implement man-in-the-middle attack, he replaces $T_A = aP$ with cP and substitutes $T_B = bP$ with vP. Then, $K_{AV} = e(P,P)^{avs}\ e(P,Q_B)^{asx_B}\ e(Q_A,P)^{vsx_A}\ e(Q_A,Q_B)^{sx_Ax_B}$. The adversary knows v, aP, and P_A, hence, he can compute $e(P,P)^{avs}$ and $e(Q_A,P)^{vsx_A}$. However, if he wants to compute $e(P,Q_B)^{asx_B}$ and $e(Q_A,Q_B)^{sx_Ax_B}$, he must know S_B or asP from aP and sP , which is a CDH problem.

B. Algorithm Complexity

The complexity of the proposed key agreement protocol and other existing protocols are compared and discussed in terms of communication and computation overhead. Table I shows the corresponding comparisons and the security attributes of the protocols. Communication overhead reflects the total parameter blocks being transferred between two entities during the key agreement process. The parameter blocks are: (P_A, T_A, P_B, T_B); each entity sends minimum two parameters to the other entity.

Pairing, scalar multiplications, exponentiations, additions, and hash are computational operations involved in the protocols. The degree of complexity of each operation in comparison with other operations is: pairing > exponentiation > scalar multiplication > addition > hash. Pairing is the most expensive operation, whereas hash has the least computation overhead. According to [24], the cost of one pairing operation is approximately equivalent to the cost of three scalar multiplications.

TABLE 1. SECURITY ATTRIBUTES AND COMPLEXITY COMPARISONS OF KEY AGREEMENT PROTOCOLS

Protocols	Security Weakness[a]	Computational Operations			Communication Overhead (block)
		Pairing	Scalar multiplication	Exponentiation	
Scheme [4]	KSTIS, KRA	4	2	1	2
Scheme [13]	KCI, KRA	1	3	0	2
Scheme [10]	PFS, MIMA, KSTIS, KRA	1	2	1	2
Scheme [8]	KCI, KSTIS, KRA	2	2	1	2
Scheme [6]	KCI, KSTIS	2	3	1	2
Scheme [12]	-	5	0	3	2
Scheme [15]	-	2	3	1	2
Scheme [17]	-	1	2	3	2
Scheme [19]	-	1	5	0	3
Our scheme	-	1	3	0	2

[a]PFS: Perfect forward secrecy; KCI: Key-compromise impersonation; KSTIS: Known session-specific temporary information security; KRA: Key replicating attack; MIMA: Man-In-the-Middle Attack

As can be seen, the protocols in [4] [13] [10] [8] [6] have some security weaknesses. On the other hand, [12] [15] [17] [19] provide all the security attributes similar to the proposed protocol. The protocols that achieve all the known desirable security attributes are considered heavy-weight as they involve multiple pairing and multiplication computations during the session key agreement. In contrast, the proposed protocol is more efficient by using just one pairing and three multiplications.

According to our proposed protocol, if there are multiple runs between the same entities (e.g. frequent key exchange in a distributed environment), only fresh T_A and T_B need to be exchanged in each run. Thus, the protocol can be considered just as efficient as ID-based key exchange in which the public keys (P_A, P_B) are always known to the participating entities.

VI. CONCLUSION

In this paper, secure and efficient certificateless authenticated key generation and agreement protocol are presented which produces distinct public key for a corresponding private key. In the original scheme, a dishonest KGC could restore an entity's public key by one for which it knows the secret value without fear of being recognized. However, in our proposed scheme, the existence of two public key for an identity can only result from the existence of two partial private keys binding that entity to two different public keys; only KGC could have created these two partial private keys. Thus, the new binding technique makes the KGC's substitute of a public key noticeable.

The security analysis shows that the key agreement protocol achieves almost all of the known desirable security attributes such as known-key secrecy, key-compromise impersonation, unknown key-share, known session-specific temporary information security, forward secrecy and no key control. Furthermore, it conveys better efficiency in contrast to the existing protocols. In addition, the key generation and agreement protocols reduce the amount of trust on KGC. Currently, among the future work that we plan to pursue includes investigating the efficiency of the proposed protocol in distributed environments, e.g. peer-to-peer and grid computing platforms.

REFERENCES

[1] R. Younglove, "Public key infrastructure. how it works," *Computing & Control Engineering Journal*, vol. 12, pp. 99–102, 2001.

[2] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. CRYPTO 84 on Advances in cryptology*. New York, NY, USA: Springer-Verlag New York, Inc., 1985, p. 47–53.

[3] C. Gentry, "Certificate-based encryption and the certificate revocation problem," in *Proc. EUROCRYPT'03*. Berlin, Heidelberg: Springer-Verlag, 2003, p. 272–293.

[4] S. S. Al-riyami, K. G. Paterson, and R. Holloway, "Certificateless public key cryptography," in *Proc. Asiacrypt'03*. Springer-Verlag, 2003, p. 452–473.

[5] K. Paterson and G. Price, "A comparison between traditional public key infrastructures and identity-based cryptography," *Information Security*, vol. 8, no. 16, pp. 57–72, 2003.

[6] T. K. Mandt and C. H. Tan, "Certificateless authenticated two-party key agreement protocols," in *Proc. ASIAN'06,* Berlin, Heidelberg: Springer-Verlag, 2007, p. 37–44.

[7] L. Xia, S. Wang, J. Shen, and G. Xu, "Breaking and repairing the certificateless key agreement protocol from asian 2006," *Wuhan University Journal of Natural Sciences*, vol. 13, no. 5, pp. 562–566, Nov. 2008.

[8] W. Shengbao, C. Zhenfu, , and W. Licheng, "Efficient certificateless authenticated key agreement protocol from pairings," *Wuhan University Journal of Natural Sciences,* vol. 11, no. 5, pp. 1278–1282, Sept. 2006.

[9] S. Zu-hua, "Efficient authenticated key agreement protocol using self-certified public keys from pairings," *Wuhan University Journal of Natural Sciences,* vol. 10, no. 1, pp. 267–270, Jan. 2005.

[10] S. Yijuan and L. Jianhua, "Two-party authenticated key agreement in certificateless public key cryptography," *Wuhan University Journal of Natural Sciences*, vol. 12, no. 1, pp. 71–74, Jan 2007.

[11] C. M. Swanson, "Security in key agreement: Two-party certificateless schemes," Master's thesis, University of Waterloo, 2008.

[12] G. Lippold, C. Boyd, and J. Gonzalez Nieto, "Strongly secure certificateless key agreement," in *Proc. Pairing '09,* Berlin, Heidelberg, Germany: Springer- Verlag, 2009, p. 206–230.

[13] S. Wang, Z. Cao, and H. Bao, "Efficient certificateless authentication and key agreement (cl-ak) for grid computing," *International Journal of Network Security*, vol. 7, no. 3, pp. 342–347, 2006.

[14] M. Hou and Q. Xu, "On the security of certificateless authenticated key agreement protocol (cl-ak) for grid computing," in *Proc. CHINAGRID '09,* Washington, DC, USA: IEEE Computer Society, 2009, p. 128–133.

[15] M. Hou and Q. Xu, "Two-party authenticated key agreement protocol from certificateless public key encryption scheme," in *Proc. Icmecg (International Conference on Management of e-Commerce and e-Government), 2009*, vol. 0. Los Alamitos, CA, USA: IEEE Computer Society, 2009, p. 440–444.

[16] Y. Sun and F. Zhang, "Secure certificateless public key encryption without redundancy," Cryptology ePrint Archive, Report 2008/487, 2008, http://eprint.iacr.org/.

[17] M. Hou and Q. Xu, "Secure and efficient two-party certificateless authenticated key agreement protocol," in *Proc. CCCM (Colloquium on Computing, Communication, Control, and Management),* 2009, vol. 3, p. 308 –311.

[18] B. Libert and J. jacques Quisquater, "On constructing certificateless cryptosystems from identity based encryption," in *PKC 2006.* Springer-Verlag, 2006, p. 474–490.

[19] L. Zhang, F. Zhang, Q. Wu, and J. Domingo-Ferrer, "Simulatable certificateless two-party authenticated key agreement protocol," *Inf. Sci.*, vol. 180, no. 6, pp. 1020–1030, 2010.

[20] S. Blake-Wilson, D. Johnson, and A. Menezes, "Key agreement protocols and their security analysis," in *6th IMA International Conference on Cryptography and Coding*, ser. Lecture Notes in Computer Science, vol. 1355. Springer Berlin / Heidelberg, 1997, p. 30–45.

[21] A. Menezes, S. Vanstone, and T. Okamoto, "Reducing elliptic curve logarithms to logarithms in a finite field," in *Proc. STOC'91(Symposium on Theory of Computing),* New York, NY, USA: ACM, 1991, p. 80–89.

[22] G. Frey, M. Muller, and H.-G. Ruck, "The tate pairing and the discrete logarithm applied to elliptic curve cryptosystems," *IEEE Transactions on Information Theory*, vol. 45, no. 5, pp. 1717 –1719, July 1999.

[23] Q. Yuan and S. Li, "A new efficient id-based authenticated key agreement protocol," Cryptology ePrint Archive: Report2005/309, 2005.

[24] P. S. L. M. Barreto, H. Y. Kim, B. Lynn, and M. Scott, "Efficient algorithms for pairing-based cryptosystems," in *Proc. CRYPTO '02,* London, UK: Springer-Verlag, 2002, p. 354–368.