An Escrow-Free Two-party Identity-based Key Agreement Protocol without Using Pairings for Distinct PKGs

Thokozani Felix Vallent[1], Eun-Jun Yoon[2], and Hyunsung Kim[2]

[1] Department of IT Convergence, Kyungil University / Gyeongsan, South Korea tfvallent@gmail.com
[2] Department of Cyber Security, Kyungil University / Gyeongsan, South Korea {ejyoon, kim}@kiu.ac.kr

***** Corresponding Author: Hyunsung Kim

Received July 14, 2012; Revised July 28, 2012; Accepted August 14, 2012; Published June 30, 2013

** Regular Paper*

***Abstract*:** Key escrow is a default property that is inherent in identity-based cryptography, where a curious private key generator (PKG) can derive a secret value shared by communicating entities in its domain. Therefore, a dishonest PKG can encrypt and decrypt ciphers or can carry out any attack on the communicating parties. Of course, the escrow property is not completely unwanted but is acceptable in other particular applications. On the other hand, in more civil applications, this key escrow property is undesirable and needs to be removed to provide maximum communication privacy. Therefore, this paper presents an escrow-free identity-based key agreement protocol that is also applicable even in a distinct PKG condition that does not use pairings. The proposed protocol has comparable computational and communicational performance to many other protocols with similar security attributes, of which their security is based on costly bilinear pairings. The protocol's notion was inspired by McCullagh et al. and Chen-Kudla, in regard to escrow-free and multi-PKG key agreement ideas. In particular, the scheme captures perfect forward secrecy and key compromise impersonation resilience, which were lacking in McCullagh et al.'s study, as well as all other desirable security attributes, such as known key secrecy, unknown key-share resilience and no-key control. The merit in the proposed protocol is the achievement of all required security requirements with a relatively lower computational overhead than many other protocols because it precludes pairings.

***Keywords*:** Identity-based key agreement, Key escrow, Distinct PKG, Pairing free and perfect forward secrecy

1. Introduction

Identity-based key agreement techniques have changed public key cryptography protocols considerably and its novel construction has attracted significant research attention in designing key exchange protocols for different applications. The most desirable property that goes with identity-based cryptography (IBC) is its simplicity and efficiency in the establishment of key exchange among corresponding parties. In this public key scenario, the user's public keys are derived directly from their respective identities, such as email address, IP address or any publicly known string of any identification credential. Therefore, the technique offsets the need for public key infrastructure(PKI) services in certificate management and distribution, and is deemed to have less public key management overhead. On the other hand, by nature, IBC allows the PKG to derive the session key established by the parties involved in the communication because of its inherent property known as key escrow. In this case, the PKG can eavesdrop, modify the communication content or masquerade as one of the legal participants, a likely attack in Mobile Ad Hoc Network (MANET) communication environment, where a mobile node can be masqueraded by a spy node imposed by a PKG [1]. Therefore, based on the application area, the key escrow property can either be viewed as desirable or undesirable. For example, this property may be desirable in government and military-

Preliminary results of this paper were presented at KICS winter conference 2013. This research was supported by the National Research Foundation of Korea Grant funded by the Korean Government (MEST) (NRF-2011-0008890) and the MSIP(Ministry of Science, ICT & Future Planning) support program (NIPA-2013- H0301-13-2004) supervised by the NIPA(National IT Industry Promotion Agency).

based applications as well as all cases where an audit trail and confidentiality are legally required [2]. One possible area that requires an escrow key agreement protocol is the u-healthcare setting to allow the tracking of past transactions in the case of conflict between a patient and their doctor. In all, this property is viable wherever administrative monitoring is a necessity of the system transactions. On the other hand, the property of key escrow is not needed in communications that require strict confidentiality, integrity and authenticity. Therefore, to ensure the security of user's sensitive information in civil applications, escrow-free identity-based protocols provide an ideal cryptosystems. In this way, all attacks that could be performed by a dishonest or curious PKG, through both passive attack and active attack, can be avoided.

A key agreement protocol between communicating entities *A* and *B* is said to provide implicit authentication (of *B* to *A*) if *A* is assured that no other entity besides *B* can ascertain the value of the secret key. When a key agreement protocol provides mutual implicit key authentication, it is known as an authenticated key agreement protocol (AK protocol). On the other hand, a key agreement protocol is said to provide key confirmation (of *B* to *A*) if *A* is assured that *B* possesses the secret key. Accordingly, a protocol that provides mutual key authentication as well as mutual key confirmation is called an authenticated key agreement with key confirmation protocol (AKC protocol) [3]. Either an AK protocol or KC protocol needs to possess the following security attributes.

Known-key secrecy: Each protocol run should produce a unique session key independently of each other, so that the compromise of a particular session's secret value should not lead to the compromise of a shared secret value of another session. Session keys should be independent of each other, such that with knowledge of one session key, an adversary cannot deduce another protocol run's session key.

Key compromise impersonation (*KCI*) *resilience*: If the long-term private key for entity *A* is compromised, an adversary is only allowed to impersonate *A* to other entities, but should not be able to impersonate other entities to *A*.

Unknown key share security: An entity, *A*, should not be coerced into sharing a key with any other entity, *C*, when in fact *A* is believed to be sharing the key with another entity *B*.

Forward secrecy: If long-term private key(s) of one or more entities are compromised, an attacker should not be able to determine the previously established session keys. *Perfect forward secrecy* (PFS) implies an attacker, armed with the long-term private keys of all participants, cannot determine the old session keys. PFS indicates that an attacker armed with some but not all the participants' long-term private keys cannot determine the old session keys. In an identity-based system the corruption of the PKG's master secret implies the compromise of all the users' long-term private keys. Therefore, *PKG* - PFS means that the compromise of the PKG's master secret key should not affect the security of the session keys established by any users. This certainly suggests PFS.

No-key control: Neither entity should be able to force the session key to be a pre-selected value.

Therefore, this paper presents a protocol with the following contributions: (1) an efficient escrow-free identity-based authenticated key agreement (ID-AK) protocol without pairings; (2) provision KIC and PFS security attributes, which have been lacking in other protocols, besides the satisfaction of all other required attributes; and (3) a collusion-free variant of the protocol instantiated under distinct PKG conditions.

This paper is structured as follows. Section 2 reviews some related work, and section 3 introduces some preliminaries outlining the mathematical background and complexity assumption definitions. Section 4 presents the proposed protocol in a single domain setting with escrow-free properties as well as another form of the protocol in the separation domain setting with a collusion-free property. Section 5 analyzes the satisfaction of the security attributes of the protocol and its performance analysis. Finally, the paper is concluded in section 6.

2. Related Works

The notion of IBC was first introduced by Shamir in 1984 as a way of simplifying public key cryptography by eliminating the certificate management overhead on the part of PKI [2-4]. Based on the IBC, the fully functional identity-based scheme was not proposed until 2001. Boneh and Franklin designed a practical ID-based encryption scheme [5] using bilinear pairing on elliptic curves with security proof in a random oracle. Since then, a significant number of ID-AK protocols based on the pairing idea have been proposed [2-6]. In 2003, Sakai and Kasahara streamlined the IBC by pairing on the elliptic curves [2], whereas Smart [6] proposed the first ID-AK protocol using pairings based on the idea reported by Boneh and Franklin. Since then, many more ID-AK protocols have been proposed based on the hypothesis in [5], but Shim [7] identified a security weakness with the protocol in reference [6]. Many identity-based protocols have been designed to achieve either key escrow mode or key escrow-free mode [2, 8] but they were reported to be weak against some well-known attacks. Chen-Kudla [9] modified the identity-based key agreement schemes to enable two entities belonging to different domains (PKGs) to establish a common shared secret. Multi-domain key agreement protocols are well suited for global scale applications that require compatibility between different networks. For example, it can be used to secure VoIP applications. Based on the idea reported in [9] McCullagh et al. devised a protocol instantiated both in escrow and escrow-free mode as well as in a multi-PKG setting. Unfortunately, the protocol suffers from well-known attacks, such as KCI, and does not provide PFS, as unveiled by Axie [10]; note the masquerading attack in reference [11]. In 2006, Gentry proposed another IBE system that is fully secure in the standard model and has several advantages [12]. Based on Gentry's construction Wang et al. [8] designed two new ID-based, implicit key authenticated key agreement protocols that can be used either in escrow or escrow-less mode. On the other hand, Hou et al. in [3] reported that the protocol does not satisfy the PKG forward secrecy. Therefore, this paper proposes

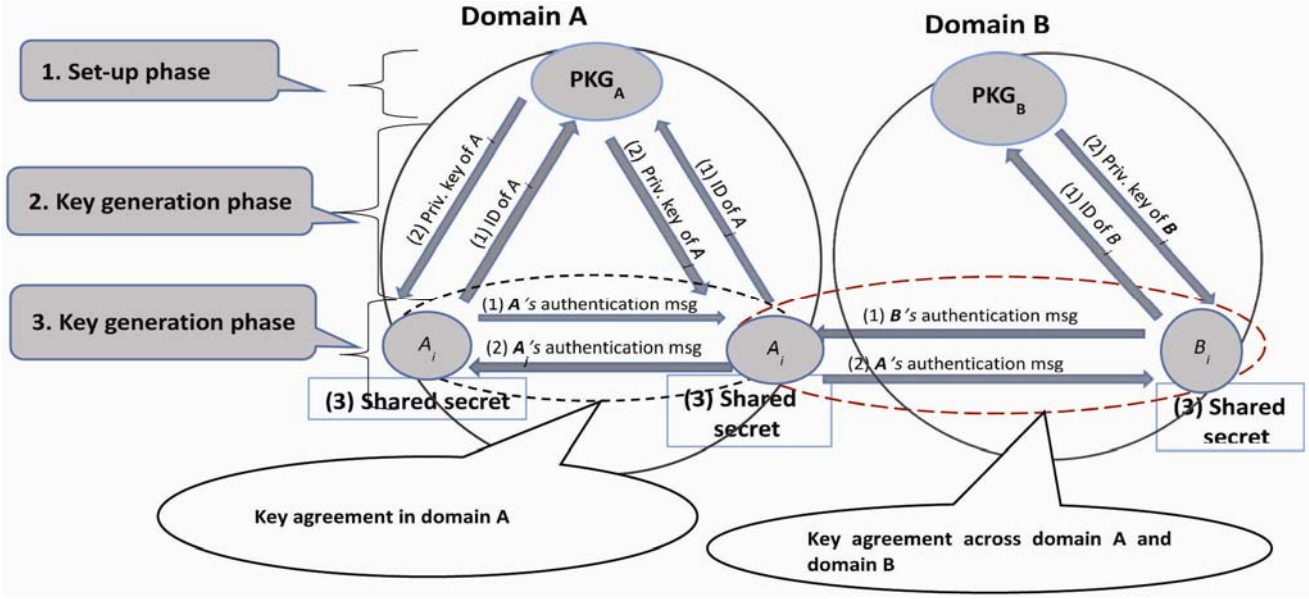

Fig. 1. ID-AK framework in a single domain and separate domains.

an efficient protocol that provides PFS and KCI resilience while achieving an escrow-free key agreement in both a single domain and separate domain scenario.

3. Preliminaries

This section first introduces some mathematical background and the definitions of complexity assumptions that form the security basis of the proposed protocol as follows:

3.1 Elliptic curve group and mathematical difficult problems definitions

Elliptic curve cryptography (ECC) is an approach for public key cryptography (PKC) based on the algebraic structure of elliptic curves over a finite field. The use of an elliptic curve in cryptography is based on the infeasibility of finding the discrete logarithm of a random elliptic curve element with respect to a publicly known base point in the elliptic curve over a finite field. The attraction of ECC is its efficiency and high security with small sized keys. The size of the elliptic curve determines the difficulty of the problem. The use of ECC was suggested independently by Koblitz and Miller in 1985 [13]. E/F_q denote an elliptic curve, E, over a prime finite field, F_q, defined by the equation, $y^2=x^3+ax+b$ with $a,b\in F_q$ and the discriminant $\Delta=4a^3+27b^2\neq 0$. All points on E/F_q including the point at infinity, O, form a group $G=\{(x,y)|x,y\in F_q;(x,y)\in E/F_q\}\cup\{O\}$. G is a cyclic group under point addition "+" with O as an additive identity, such that, $P+(-P)=O$, for any point P on E/F_q. Therefore, similar to the normal finite field a group of points on a finite field over elliptic curve, E/F_q is also defined under addition and scalar multiplication. Many difficult problems associated with scalar multiplication used in elliptic curve cryptography are encountered, as defined below.

Definition 3.2: Elliptic Curve Discrete Logarithm Problem (ECDLP)

Given an elliptic curve E, defined over a finite field, F_q, let P and Q be points on E/F_q. The ECDLP is the problem of finding an integer n such that $Q=nP$. This integer n is denoted as $n=log_P Q$ [13].

Definition 3.3: Elliptic Curve Computational Diffie-Hellman Problem (ECDHP)

Let E/F_q be an elliptic curve over a finite field and let P be a point in E/F_q of order n (i.e. $nP=O$). *ECDHP* is a problem of computing the value, n_1n_2P, from the known values of n_1P and n_2P [14].

4. The Proposed Scheme

This section present an efficient escrow-free ID-AK protocol that can either be instantiated both in escrow-free modes with single PKG and distinct PKGs scenarios. Fig. 1 presents the protocol framework. The following depict the phases involved in the entities to share a secret value, which are the set up phase, key generation phase and key agreement phase.

The PKG produces system parameters in the set up phase so it is responsible for deriving an entity's public key from the identity. In the key generation phase, an entity requests for a private key corresponding to its identity. Upon verification of a claimed identity, the PKG offers the private key securely to an entity. In the key agreement, the phase communicating entities can establish a secret shared value verifiably using public and private

parameters. The design allows secret key sharing both in the single domain and separate domains. Unlike many other protocols, which are designed to allow escrow-free property and separate the PKGs key agreement, the proposed protocol is not based on the expensive bilinear pairing primitives. This feature accredits the proposed protocol to the computation overhead because pairing computation is at least 10 times heavier than scalar multiplication in the same field [15]. With two passes, the protocol manages to establish a mutually secure key agreement by incorporating the signature attributes in the message exchange.

4.1 An escrow-free key ID-based key agreement protocol

The protocol consists of three phases, the set-up, key generation and key agreement phases.

Set-up: The PKG takes the secret parameter k and a master key s and performs the following:

(1) Choose a k-bit prime p and determine the tuple; $\{F_q, E/F_q, G, P\}$.
(2) Choose the master secret key $s \in Z_q^*$ and compute the *system public key* $P_{pub}=sP$.
(3) Choose two cryptographic secure hash functions $H_1: \{0,1\}^* \rightarrow Z_q^*$ and $H_2: \{0.1\}^* \rightarrow \{0,1\}^k$.
(4) Then PKG publishes public system parameters; $\{F_q, E/F_q, G, q, P, H_1, H_2\}$ and keeps the master secret s secret.

Key generation: The PKG checks that the user's online identifier is correct before issuing a private key to a user. The PKG takes as input the system parameters, master key and user's identifier, carries out the computations and then returns the user's ID-based long-term private key. For a user, A, with a particular online identifier, e.g. ID_A, the PKG by using the key generation algorithm, works as follows:

(1) Map a user's online identifier ID_A to an integer elements, e.g. $H_1: \{ID_A\} \rightarrow a \in Z_p^*$.
(2) Compute A's public key as $A_{pub}=(a+s)P$ and A's private key as $A_{pri}=(a+s)^{-1}P$. The public key is a publically computable value to anyone with knowledge of the identifier, because $A_{pub}=H_1(ID_A)P+sP$.

In a similar manner, the algorithm generates B's pair of public and private keys: $B_{pub}=(b+s)P$ and $B_{pri}=(b+s)^{-1}P$, respectively.

Key agreement: Entity $A(ID_A)$ and entity $B(ID_B)$ run the following algorithm to establish a securely shared session key.

Step 1: A initiates a session with B as follows:
(1) Entity A chooses a random ephemeral key, $x \in Z_q^*$, and calculates $T_A=xP$ and signature $S_A=xA_{pri}+H(ID_A||T_A)A_{pri}$.
(2) Then A sends, ID_A, T_A and S_A to B.
$A \rightarrow B: \{ID_A, T_A, S_A\}$

Step 2: Upon receipt of A's message, B does the following.
(1) Check the validity of the received message, $\{ID_A, T_A, S_A\}$ from A and then verify the authenticity by using the signature part, S_A as follows: B confirms if $S_A A_{pub}=T_A P+H(ID_A||T_A)P^2$ using A's public key. This verification authenticates both the message and its source. The hash computation, $H(ID_A||T_A)$, also ensures the integrity of the ephemeral key, $x \in Z_q^*$, in T_A, used for key agreement as follows:
(2) If the verification holds, B, chooses a random ephemeral key, $y \in Z_q^*$, and calculates $T_B=yP$ and the signature part, $S_B=xB_{pri}+H(ID_B||T_B)B_{pri}$.
(3) B first calculates $Z_{BA} = yT_A$ and then computes the session key as $SK=H_2(ID_A||ID_B||Z_{BA})$.
(4) B then sends, ID_B, T_B and S_B to A.
$B \rightarrow A: \{ID_B, T_B, S_B\}$

Step 3: Upon receipt of B's message A does the following.
(1) Check the authenticity of the message by confirming if $S_B B_{pub}=T_B P+ H(ID_B||T_B)\ P^2$, otherwise B quits the session.
(2) Upon correct verification results, A computes $Z_{AB}=xT_B$, and then computes the session key as, $SK=H_2(ID_A||ID_B||\ xT_B)$.

Fig. 2 gives a description of the protocol's message flow, between A and B. Entities A and B now agree on the same session key because $Z_{AB}=xT_B=xyP$ and $Z_{BA}=yT_A=yxP$. Although the PKG has knowledge of the private key material for either entity, it cannot calculate the shared

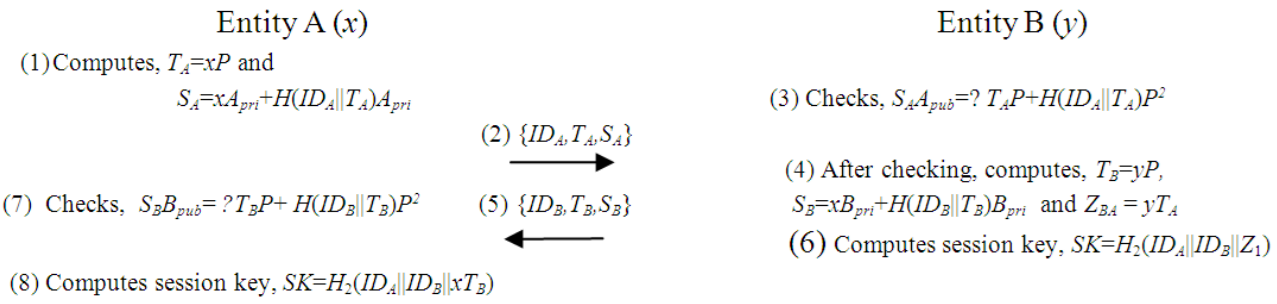

Fig. 2. Escrow-free and pairing-free ID-AK protocol.

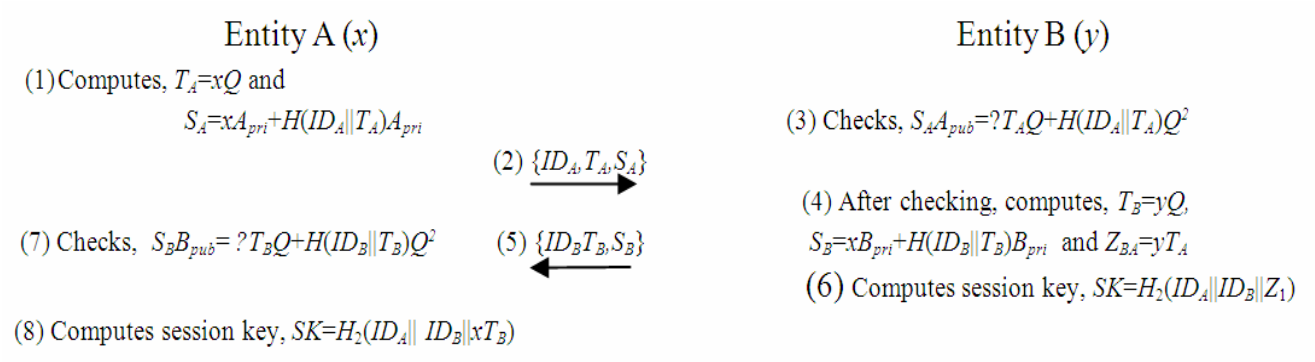

Fig. 3. A collusion-free and pairing-free ID-AK protocol over distinct domains.

session key between *A* and *B* due to the *ECDLP* of $T_A=xP$ and $T_B=yP$. Therefore the protocol provides an escrow-free property without using the heavier bilinear pairing computation. Furthermore, the proposed protocol is escrow-free because the PKG will not be able to calculate the session key, $SK=H_2(ID_A||\ ID_B||\ xT_B)$, from the public messages regardless of knowing the parties private keys due to the *ECDHP* of $Z_{AB}=xyP=Z_{BA}$.

4.2 Collusion-free ID-based key agreement protocol over distinct PKGs

In the same manner, the proposed protocol can also be extended to establish the authenticated key between entities of distinct PKG, as described below. In the case of key agreement over distinct domains, the protocol takes the same order in: set-up, key generation and key agreement phases. The only exception is that the entities of different domains can establish a secure shared key using different public and private keys generated from the distinct PKGs. Therefore, in the set-up phase, there are two PKGs, PKG_1 and PKG_2, responsible for generating the corresponding parties' private keys using their master secret keys. For example, party *A* obtains the private key $A_{pri}=(a+s_1)^{-1}Q$,from PKG_1 with a master secret key s_1, where *Q* is the generator of a prime order group, *G*. Similarly, a party, *B*, belonging to PKG_2 obtains the private key, $B_{pri}=(b+s_2)^{-1}Q$, in the key generation phase. The protocol's steps and message flows are shown in Fig. 3.

The protocol is collusion-free because if PKG_1 and PKG_2 collude to compute a session key shared between *A* and *B*, they cannot manage just because of the difficult problem of solving $Z_{AB}=xyP=Z_{BA}$ because they do not know the ephemeral key, just the public values, T_A and T_B. Therefore, they encounter the same difficult problems of *ECDLP* and *ECDHP* as in a single PKG scenario.

5. Analyses

This section shows the satisfaction of the security attributes required for a secure key agreement that are achieved in the proposed protocol and a performance analysis with the main focus on efficiency.

5.1 Security Analysis

This subsection shows how the desirable security properties are satisfied by the proposed protocol. Although the description for security requirements is based on the escrow-free mode with communicating entities belonging to the same domain, the same security basis applies to a distinct PKG scenario.

Known Key Security: Even if one session key is compromised, still more other session keys apart from the compromised ones remain secure. This is simply because every session key is unique due to the randomly chosen ephemeral key for each protocol run. Therefore, an attacker would not know any other session key from the knowledge of a compromised one because the session key computation depends on the random ephemeral keys, which is given by $SK=H_2(ID_A||ID_B||xyP)$.

Key Compromise Impersonation (KCI) *Resilience*: In the case where *A*'s long-term private key is in the hands of an adversary (*M*), it should only be possible to impersonate *A* to other entities, i.e. *B*, but be impossible to impersonate other entities to *A*. This security attribute is well satisfied in the protocol because any sender of a message endorses its authenticity by sending a verifiable signature component, $S_A=xA_{pri}+H(ID_A||T_A)A_{pri}$, that proves the ownership of the *ID* and corresponding public key. Therefore, without knowledge of the private key, e.g. $B_{pri}=(b+s)^{-1}P$ for *B*, (*as an entity to be impersonated*), no adversary can form a verifiable signature component S_B. Therefore, the proposed protocol secures against a KCI attack.

Unknown Key Share Resilience: No legal entity, *A*, of the corresponding parties can be duped unwittingly into sharing a session key with an illegal malicious entity, *C*, (an adversary), instead of the proper and intended one, entity *B*. This is simply because immediately before agreeing on a session key, every entity (say *B*) needs to verify the authenticity of the sender (*A*) and the message sent using the sender's public key to determine the validity of the equation $S_AA_{pub}=?T_AP+H(ID_A||T_A)P^2$. Therefore, verification of the signatures, S_A and S_B, implicitly authenticates both terms, T_A and T_B, together with their

Table 1. Performance Comparisons.

Operation / Protocol	Pairing	Exponentiation	Scalar Multiplication	Map to a point	Group addition
McCullagh et al. [2]	1	1	2	1	1
Hou et al. [3]	2	5	-	2	-
Smart [6]	2	-	2	1	-
Wang et al. [8]	2	4	-	1	-
Chen et al. [9]	1	-	4	2	1
Our protocol	-	-	4	3	2

corresponding identities, ID_A and ID_B. Accordingly, the protocol foils the adversary's ability of forging a signature, thereby checking the authenticity of the source of the message and the integrity of the *ECDLP* component, T_A and T_B, achieving the unknown key share security attribute.

PKG Perfect forward secrecy: If the PKG's master secret key is corrupted, still more previously established session keys between *A* and *B* remain uncompromised. This is because the session key is derived, as shown in $SK=H_2(ID_A||ID_B||xyP)$. The component, *xyP*, is a *ECDHP* of which PKG cannot calculate merely by having access to, $T_A=xP$ and $T_B=yP$. Therefore, in a similar manner, the protocol achieves *partial forward secrecy* and PKG. Partial perfect forward secrecy is when compromise of the long-term private key of one entity does not lead to the compromise of all previously established session keys. Although perfect forward secrecy is the scenario when both communication entities' long-term private keys are compromised, the previously agreed upon session keys remain secure from reach of an adversary. This property is achieved using the same argument as that for *PKG*-PKG.

No-Key control: Because both parties jointly contribute to the inputs of the session key, i.e. $T_A=xP$ and $T_B=yP$, from *A* and *B*, respectively, it means none of them has full power to influence the session key to a pre-selected value. All that an entity can manage to determine is to keep the key within certain desirable bits by carefully selecting the ephemeral key. Therefore, it is advisable to set a short time-out on a particular run of the protocol to avoid further manipulation of the shared value.

5.2 Performance Analysis

This subsection reports the results of a comparison analysis of the computational overhead in relation to other key agreement protocols. This subsection accounts for only the substantial operations involved, i.e. pairing, exponentiation, scalar multiplication, map to a point, and group addition. Therefore, that pairing operation is at least 10 times as heavy as scalar multiplication in the same field [15, 16]. Hence, the proposed protocol fairs well in efficiency terms more than all comparative protocols [2, 3, 6, 8, 9]. With this criterion, the proposed protocol achieves security with considerable efficiency because it is pairing-free unlike many other ID-based key exchange protocols, as shown in Table 1. The proposed protocol involves no pairing and no exponentiation operation, whereas the remaining protocols includes at least one pairing or one exponentiation operation, this feature determines the reduction of the overall computational cost. The protocols in [6] and [9] do not use exponentiation, but their computational load is determined more by the pairing operation so they are still heavier than the currently proposed protocol. Similarly, although protocols [3] and [8] do not use both scalar multiplication and group addition, their computation load is overweighed by the inclusion of 2 pairing operations. Therefore, comparatively, the proposed protocol has a lighter computational cost.

6. Conclusion

This paper presented an efficient ID-based key agreement protocol that is pairing- and escrow-free in a single PKG, and is also collusion-free in two distinct PKGs key exchange scenarios. The protocol achieves all the desirable security attributes with minimum computational cost based on the ECDLP and ECDHP problems. In addition to the security properties of no key control, unknown key share resilience and known key security, the proposed protocol also provides KCI resilience and PFS, which are properties lacking in many other ID-based key agreement protocols. The proposed protocol integrates a signature component in the message flows for common key computations, which ensures message integrity and the authenticity of the source of the message. The merit of the protocol is that it achieves security at a very low computational cost, making it ideal for applications to hand held devices.

References

[1] K. Hoeper and G. Gong, "Preventing or Utilizing Key Escrow in Identity-Based Schemes Employed in Mobile Ad Hoc Networks," International Journal of Security Networks 2, Vol. 3, pp. 239-250, 2007. Article (CrossRef Link)

[2] N. McCullagh, P. S. L. M. Barreto, "A new two-party identity-based authenticated key agreement," Proc. of the topics in Cryptology-CT-RSA 2005, pp. 262-274, 2005. Article (CrossRef Link)

[3] M. Hou and Q. Xu, "A Secure ID-Based Explicit Authenticated Key Agreement Protocol Without Key Escrow," IAS'09, fifth International Conference, Vol. 1, pp. 487-490, 2009. Article (CrossRef Link)

[4] S. Chow, "Removing Escrow from Identity-Based Encryption, New Security Notions and Key management Techniques," 12th International Conference on Pratice and Theory in Public Key Cryptography, pp. 256-272, 2009. Article (CrossRef Link)
[5] D. Boneh, M. Franklin, "Identity-based encryption from the Weil pairing," Lecture Notes in Computer Science, Vol. 2139, pp. 213-229, 2001. Article (CrossRef Link)
[6] N. P. Smart, "Identity-based authenticated key agreement protocol based on Weil pairing," IEE Electronics Letters, Vol. 38, No. 13, pp. 630-632, 2002. Article (CrossRef Link)
[7] K. Shim, "Efficient ID-based authenticated key agreement protocol based on Weil pairing," IEE Electronics Letters, Vol. 39, No. 8, pp. 653-654, 2003. Article (CrossRef Link)
[8] S. Wang, Z. Cao and K-K. R. Choo, "Provably Secure ID-Based Authenticated Key Agreement without Random Oracles," Cryptology ePrint Archive, pp. 1-6, 2006. Article (CrossRef Link)
[9] L. Chen, C. Kudla, "Identity based key agreement protocols from pairings," Proc. of the 16thIEEE Computer Security Foundations Workshop 2002, pp. 219-233, 2002. Article (CrossRef Link)
[10] G. Xie, "Cryptanalysis of Noel McCullagh and Paulo S. L. M. Barreto's two-party identity-based key agreement," Cryptology ePrint Archive, 2004/308, http://eprint.iacr.org/2004/308/. Article (CrossRef Link)
[11] T.F. Vallent, S-W Lee, E.J Yoon and H.S. Kim, "Cryptanalysis and remedy of two-party identity-based authenticated key agreement protocol," Proc. Of KICS winter conference 2013, pp. 120-121, 2013. Article (CrossRef Link)
[12] C. Gentry, "Practical identity-based encryption without random oracles", In Proc. Of EUROCRYPT, LNCS, Vol. 4004, pp. 445-464, Springer-Verlag, 2006. Article (CrossRef Link)
[13] J. Hoffstein, J. Pipher and J.H Silverman, An introduction to mathematical cryptography, Springer,Spring street New York, 2008. Article (CrossRef Link)
[14] D. Hankerson, A. Menezes, S. Vanstone, Guide to elliptic curve cryptography, Springer-Verlag, 2004. Article (CrossRef Link)
[15] R.W. Zhu, G. Yang and D.S. Wong, "An efficient identity-based key exchange protocol with KGS forward secrecy for low-power device," Theoretical Computer Science, Vol. 378, pp. 198-207, 2007. Article (CrossRef Link)
[16] X. Cao, W. Kou and X. Du, "A pairing-free identity-based authenticated key agreement protocol with minimal message exchange" Information Science, Vol. 180, No. 15, pp. 2895-2903. 2010. Article (CrossRef Link)
[17] L. Law, A. Menezes, M. Qu, J. Solinas and S. Vanstone, "An efficient protocol for authenticated key agreement," Designs, Codes and Cryptography, Vol. 28, No. 2, pp. 119-134, 2003. Article (CrossRef Link)
[18] W. Diffie and M.E Hellman, "New directions in cryptography," IEEE Transactions on Information Theory,Vol. 22, pp.644-654, 1976. Article (CrossRef Link)
[19] A. Shamir, "Identity-based cryptosystems signature schemes," Lecture Notes in Computer Science, Vol. 196, pp. 47-53, 1984. Article (CrossRef Link)
[20] H. Sun and B. Hsieh, "Security analysis of Shim's authenticated key agreement protocols from pairings," Cryptology ePrint Archive, 2003/113, http://eprint.iacr.org/2003/113/. Article (CrossRef Link)
[21] E. Ryu, E. Yoon and K. Yoo, "An efficient ID-based authenticated key agreement protocol from pairings," Lecture Notes in Computer Science, Vol. 3042, pp. 1458-1463, 2004. Article (CrossRef Link)
[22] C. Boyd and K. K. R. Choo, "Security of two-party identity-based key agreement," Lecture Notes in Computer Science, Vol. 3715, pp. 229-243, 2005. Article (CrossRef Link)
[23] P. Kumar and H.J Lee, "Security Issues in Healthcare Application Using Wireless Medical Sensor Network: A Survey," Sensors (142443220): Vol. 12, Issue. 1, pp.55-91, 2012. Article (CrossRef Link)

Thokozani Felix Vallent received his B.Ed. in Mathematics from University of Malawi-Chancellor College, Malawi, in 2007. Currently he is a graduate student of Department of IT Convergence at Kyungil University, Republic of Korea. His current research interests are cryptography, authentication, RFID security, cognitive radio network security and u-healthcare security.

Eun-Jun Yoon received his MSc degree in computer engineering from Kyungil University in 2002 and the PhD degree in computer science from Kyungpook National University in 2006, Republic of Korea. From 2007 to 2008, he was a full-time lecturer at Faculty of Computer Information, Daegu Polytechnic College, Republic of Korea. From 2009 to 2011, he was a 2nd BK21 contract professor at the School of Electrical Engineering and Computer Science, Kyungpook National University, Republic of Korea. Currently, he is a professor at the Department of Cyber Security, Kyungil University, Republic of Korea. His current research interests are cryptography, authentication, smart card security, network security, mobile security, and steganography.

Hyunsung Kim received his M.E. and Ph.D. degrees in Computer Engineering from Kyungpook National University, Republic of Korea, in 1998 and 2002, respectively. From 2000 to 2002, he worked as a senior researcher at Ditto Technology. He had been an associate professor from 2002 to 2011 with the Department of Computer Engineering, Kyungil University. Currently, he is an associate professor at the Department of Cyber Security, Kyungil University, Republic of Korea. His current research interests are cryptography, VLSI, authentication technologies, network security and ubiquitous computing security.