한국통신학회논문지 '13-07 Vol.38C No.07
http://dx.doi.org/10.7840/kics.2013.38C.7.595

개선된 두 참여자간 식별자 기반 인증된 키 동의 프로토콜

발렌트 토코자니◆, 김 혜 정*, 윤 은 준**, 김 현 성°

Improved Two-Party ID-Based Authenticated Key Agreement Protocol

Thokozani Felix Vallent◆, Hae-Jung Kim*, Eun-Jun Yoon**, Hyunsung Kim°

요 약

공개된 네트워크 상에서 통신하는 두 참여자를 위한 안전한 인증된 키 동의 프로토콜(AKA)을 고안하는 것은 중요한 연구이다. McCullagh등은 단일 도메인과 두 개의 도메인을 위해 사용될 수 있는 제 3자 키 기탁(escrow)과 제 3자 키기탁이 필요없는 두가지 속성을 지원하는 두 참여자간 식별자 기반 인증된 키동의 프로토콜을 제안하였다. 본 논문은 McCullagh등의 두 개의 도메인을 위한 프토토콜이 가장 공격(masquerading attack)에 취약함으로서 주장하는 보안을 만족하지 않음을 보인다. McCullag등의 기법에 존재하는 가장 공격은 충분한 개체 인증과 무결성 보증의 부족 때문에 발생한다. McCullagh등의 프로토콜 문제점을 해결하기 위해서 인증절차에 서명 원리가 포함된 효율적인 검증가능한 키 동의 프로토콜을 제안한다.

Key Words : Information security, Three-party authentication, Authenticated key agreement

ABSTRACT

Devising a secure authenticated key agreement (AKA) protocol for two entities communicating over an open network is a matter of current research. McCullagh et al. proposed a new two-party identity-based AKA protocol supporting both key escrow and key escrow-less property instantiated by either in a single domain or over two distinct domains. In this paper, we show that their protocol over two distinct domains suffers from masquerading attack and therefore does not satisfy the claimed security. The attack is made possible due to the lack of sufficient authentication of entity and integrity assurance in the protocol. We then propose an efficient verifiable key agreement protocol by including signature primitive in the authentication procedure to solve the problem of McCullagh et al.'s protocol.

Ⅰ. Introduction

The fundamental cryptographic primitive allows two or more party key agreement protocol enabling communicating entities to establish a shared secret key over an insecure channel. Once in possession of

※ This work was supported by the National Research Foundation of Korea Grant funded by the Korean Government (MEST) (NRF-2010-0021575), partially by the National Research Foundation of Korea Grant funded by the Korean Government (MEST) (NRF-2011-0008890) and the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program (NIPA-2012-H0301-12-2004) supervised by the NIPA (National IT Industry Promotion Agency)

◆ 주저자 : 경일대학교 IT융복합학과, tfvallent@gmail.com, 학생회원
° 교신저자 : 경일대학교 사이버보안학과, kim@kiu.ac.kr, 정회원
* 계명대학교
** 경일대학교, ejyoon@kiu.ac.kr, 종신회원
논문번호 : KICS2013-03-139, 접수일자 : 2013년 3월 26일, 최종논문접수일자 : 2013년 6월 28일

a secure shared key subsequent messages are also secured by encryption by that key. The key agreement establishment may be achieved in two ways either by transport protocol or enveloping, whereby a key is created by one entity and then securely transmitted to the counterpart, or by key agreement protocol, whereby both entities jointly contribute to the shared secret value[1-3]. The most common mechanism for AKA is by joint contribution to the key material in a way that prevents a third party from discovering the shared secret nor let either party pre-determine the shared value. The first key agreement protocol based on asymmetric cryptography is the classic Diffie-Hellman protocol by using exponentiation, whose security is based on the intractability of the Diffie-Hellman problem and the discrete logarithm problem[4,5]. In a case of identity based cryptography (IBC) user's public keys are derived from their respective identities, such as an email address or any string of an identification credential. This is made possible by the key generation centre (KGC) which is a trusted third party entrusted with the role of generating system's parameters and users public keys and private keys. Thus due to having the security basis for every user's private key, in ID-based protocols, the KGC has the ability to derive the same session key as of the communicating entities under its authority, this property is called key escrow.

In 1984, Shamir first proposed the idea of identity-based encryption which has desirable property of mitigating key management overhead associated with certificate for each public key[6]. This idea was comprehensively applied by Boneh and Franklin who proposed an identity-based encryption scheme (IBS) in 2001 by utilizing bilinear mapping[6]. Since then many identity based key agreement protocols on bilinear maps have been proposed and though some of them are elegant and practical still more a good number of them are prone to attacks[8-13]. For instance, Smart proposed an ID-based AKA based on Weil pairing, using Boneh and Franklin's IBE [7], however Shim in [9] and Chen et al. in [10] found that the protocol does not support perfect forward secrecy. Shim proposed an efficient ID-based AKA protocol using Weil pairing and claimed that her protocol is secure against well-known attacks but to the contrary Sun et al. later [11] showed that it is insecure against man-in-the middle attack. Similarly, Ryu et al.'s ID-based protocol in [12] was found subject to key compromise impersonation attack by Boyd et al. in [13]. In another work by McCullagh et al. in [14] a new two-party ID-based AKA protocol influenced by separate ideas of Sakai and Kasahara's and of Chen and Kudla respectively was proposed nevertheless Xie in [15] showed its weak against key compromise impersonation attack.

The key escrow property can either be viewed as acceptable or unacceptable property depending on the environment's interaction policy and requirements. For instance to allow institution audit trail and system surveillance policy while preserving message confidentiality of communicating entities, escrow key agreement mode would be ideal. For example to preserve confidential communication between patient and doctor in a u-healthcare system while adhering to system's interaction policy escrow mode of AKA is viable, to ensure tracking of past transactions in case of an eruption of conflict. In simple terms, key escrow property would be suitable to employ in order to allow systems administrators to monitor transactions or audit trail to check the satisfaction of organizations standard[14,16]. However, for personal and highly privacy sensitive scenarios like in e-commerce, escrow-less key agreement is the most desirable property that disallows eavesdropping of transactions even by a TA. With this awareness about the importance of both escrow and escrow-less modes McCullag et al.'s was designed to be instantiated in both scenarios. The basic format of message formation in their protocol was motivated by Sakai and Kasahara's work on the public and private key extraction from the user's identity and the KGCs master secret key[10,15]. Further McCullagh et al. adopts the concept of key agreement protocol supporting escrow-less mode between entities under two separate KGCs which was first proposed by Chen et al.[11]. Key

establishment across different domains is particularly important in enabling and providing a framework for achieving inter-connectivity and communication across heterogeneous operations system to the benefit of global scale networks such as telecommunication companies (VoIP). The established secure session key in the AKA is then used in a cryptosystem thereby achieving confidentiality, data integrity and efficiency in communication[6]. This means that key exchange should be authenticated so that each party involved is assured that the session key is shared with the intended partner not with an adversary. To achieve this requirement the key establishment process should satisfy explicit key authentication, by using message authentication code (MAC) that assures corresponding parties that the session key is correctly computed by either of them. The same notion of explicit authentication could be inferred by using signature attribute in the key agreement protocol.

McCullagh et al.'s protocol allows escrow-less key agreement, but in this paper we will unveil that it is vulnerable to masquerading attack and then we will propose a remedy for the security problem. The remedy to the protocol's flaw is fixed by verifying both message source and message integrity by employing signature attribute to the sent messages. The proposed protocol supports the conventional security requirements desirable for any key agreement scheme such as; known key security resilience, key compromise impersonation resilience, unknown key share resilience, forward secrecy and does not allow key control by either party individually. Even in both escrow and escrow-less mode the proposed remedy is valid.

In the remaining part of the paper we organize as follows: section 2. we outline elliptic curve group and mathematical difficult problems definitions, the back ground of bilinear pairing and then review McCullagh et al.'s protocol. We attack the protocol in section 3 and present a remedy in section 4, followed by security analysis of the proposed remedy in section 5 and then we conclude in section 6.

Ⅱ. Preliminaries for Mc Cullagh et al.'s Protocol

This section gives the definitions of the mathematical hard problems that forms the basis for security of the transmitted messages. Then follows a simple review of McCullagh et al.'s protocol that takes three phases: set up, extract and key agreement [13].

2.1. Elliptic Curve Group and Mathematical Difficult Problems Definitions

We introduce the elliptic curve group and the definitions of difficult problems over it that form the basis of elliptic curve cryptography, whose idea was independently suggested by Koblitz and Miller[17]. We let E/F_p to denote an elliptic curve E over a prime finite field F_p, defined by an equation $y^2=x^3+ax+b$ with $a,b\in F_p$ and with discriminant $\triangle=4a+27b^2\neq 0$. The points on E/F_p to together with an extra point O, called the point at infinity, form a group $G=\{(x,y)|x,y\in F_p;(x,y)\in E/F_p\}\cup\{O\}$. G is a cyclic additive group under point addition "+", with O as the group's identity, defined as follows: That is $P+(-P)=O$, for any point P on E/F_p . Let $P,Q\in G$ and let l be a straight line containing P and Q (tangent line to E/F_p if $P=Q$), and R, the third point of intersection of l and E/F_p . Let l be the line connecting R and O. Then P "+" Q is the point such that l intersects E/F_p at R and O and P "+" Q. This process of adding any two points on E by taking a sum of points lying on l to yield a third point also on E is called *chord-and-tangent composition* [18] for a best geometric illustration consult[17]. Scalar multiplication over E/F_p can be computed as follows: $nP=P+P+...+P_{(n\ times)}$. The following are hard problems over elliptic curve group on which the security of the cryptosystem rests upon[19].

Definition 1: **Elliptic Curve Discrete Logarithm Problem (ECDLP)**

Given an elliptic curve E defined over a finite field F_q, a point $P\in E(F_q)$ of order n, and a point

$Q \in \langle P \rangle$, Find the integer $k \in [0, n-1]$ such that $Q=kP$. The point k is called the *discrete logarithm problem of Q to the base P*, denoted $k=\log_P Q$[5].

Definition 2: **Elliptic Curve (computational) Diffie-Hellman problem (ECDHP)**
Given an elliptic curve E defined over a finite field F_q, a point $P \in E(F_q)$ of order n, and points $A=aP$, $B=bP \in \langle P \rangle$, find the point $C=abP$[5].

Definition 3: **Elliptic Curve Decision Diffie-Hellman problem (ECDDHP)**
Given an elliptic curve E defined over a finite field F_q, a point $P \in E(F_q)$ of order n, and points $A=aP$, $B=bP$ and $C=cP \in \langle P \rangle$, determine whether $C=abP$ or equivalently, whether $c=ab(\mathrm{mod}n)$[5].

Definition 4: **Elliptic Curve Gap Diffie-Hellman problem(ECGDHP)**
For $a,b \in_R Z_q^*$ and P the generator of G, given (aP,bP) as well as an oracle that solve the ECDDHP on G computing abP.

It is assumed that ECDHP and ECGDHP are hard i.e they are impossible to solve in polynomial time in a security parameter used to define the problem instances, while the ECDDHP is not as hard in bilinear pairings[19].

2.2. Background of Pairing Concept

Pairing is a mathematical construction which maps elements of two cryptographic groups to a third group that necessitate the derivation of secure cryptographic systems such as, used for building identity based *AKA* protocols and other security schemes[19]. Since the practical realization of IBC, many pairing based protocols have been proposed for different applications such as: identity based encryption, signatures, key agreement and short signature protocols. There are two pairings studied for cryptographic use which are the Weil pairing and the Tate pairing[20]. A pairing is a computable bilinear map between two groups G_1 and G_2 such that; $\hat{e}: G_1 x G_1 \rightarrow G_2$, which can either be modified Weil pairing or Tate pairing. An admissible bilinear pairing satisfies the following properties.

- Bilinear: If $P,P_1,P_2,Q,Q_1,Q_2, \in G_1$ and $a \in Z_q^*$ then; $\hat{e}(P_1+P_2,Q)=\hat{e}(P_1,Q)\hat{e}(P_2,Q)$, and $\hat{e}(aP,Q)=\hat{e}(P,aQ)=\hat{e}(P,Q)^a$
- Non-generate: There exists a $P \in G_1$ such that $\hat{e}(P,P) \neq 1$
- Computable: If P, $Q \in G_1$, one can compute $\hat{e}(aP,Q)$ in polynomial time

In the later part of the paper we will use modified Tate pairing which is more efficient bilinear pairing, denoted by the function $\hat{\imath}:(P,Q)$. So the protocol is composed of the following phases: the set up phase, the extract phase and the key agreement phase.

2.3. Review of McCullagh et al.'s Protocol

In this sub-section we present the AKA for McCullagh et al.'s protocol applicable for separate *KGC*s scenario, which involves, set-up, extract and key agreement phases.

Set Up : The key generation centre inputs a security parameter k into a Bilinear Diffie Hellman (BDH) parameter generator B_t that returns three groups G_0, G_1, and G_2 which are isomorphic finite Abelian groups, all of prime order q with $P \in G_0$ and $Q \in G_1$ as generators of the respective groups. By using these two different domain parameters generated by P and Q, the protocol achieves escrow-less key agreement. Whereas if only P (or Q) is used then the protocol will be under key escrow mode. Here, we just summarize the properties of bilinear mapping required, for more details one can check [14]. There is a mapping $\hat{\imath}$: G_0 x $G_1 \rightarrow G_2$, called an admissible Tate pairing, which efficiently generate G_2 with $H:\{0,1\}^* \rightarrow G_0$ and $H_1:\{0,1\}^* \rightarrow G_1$ as cryptographic function outputting constant strings cs_0 and cs_1 respectively. Where $\hat{\imath}(P,Q)$ is the same as $\hat{\imath}(P, \psi(P))$ for the mapping $\hat{\imath}$: G_0 x $G_1 \rightarrow G_2$, where $\psi: G_1 \rightarrow G_2$, is an efficiently computable distortion map. The *KGC* has a master secret $s \in_R Z_q^*$ and calculates a master public key sP.

Extract : To enable key escrow for distinct domains scenario, the KGC_1 generates a public and private

key pair as $A_{pub}=(a+s_1)P$ and $A_{pri}=(a+s_1)^{-1}P$, for each eligible user respectively, where a is an online identifier for an entity a mapped by some random oracle function H and a random number $a \in_R Z_q^*$. Whilst KGC_2 similarly issues public and private key pairs as $B_{pub}=(b+s_2)P$ and $B_{pri}=(b+s_2)^{-1}P$ to an entity with online identifier b mapped by some random oracle function H and a random number $b \in_R Z_q^*$. In case of escrow-less mode for distinct domains KGC_1 generates a pair of public and private keys as follows;
$A_{pub}=(a+s_1)P$ and $A_{pri}=(a+s_1)^{-1}Q$. Similarly KGC_2 generates the following keys for an entity B, $B_{pub}=(b+s_2)P$ and $B_{pri}=(b+s_2)^{-1}Q$.

Key Agreement : Each entity between A and B chooses a unique ephemeral random numbers x, $y \in Z_q^*$, respectively. A and B now perform the authenticated key agreement as follows.

Step 1: A initiates a session by computing $A_{KA}=x(b+s_2)P$ and then send the computed message to B.

Step 2: In turn B computes $B_{KA}=y(a+s_1)P$ and send to A.

Step 3: The both A and B then computes the shared session key as: $Key_{ab}=\hat{\imath}(B_{KA},A_{pri})^x$ and $Key_{ba}=\hat{\imath}(A_{KA},B_{pri})^y$, respectivley.

The shared key agrees as follows:
$Key_{ab}=\hat{\imath}(B_{KA},A_{pri})^x = \hat{\imath}(y(a+s_1)P,(a+s_1)^{-1}P)^x$
$=\hat{\imath}(P,P)^{xy} = \hat{\imath}(x(a+s_2)P,(a+s_2)^{-1}P)^y$
$=\hat{\imath}(A_{KA},B_{pri})=Key_{ba}$

This apparently is an escrowlable mode key agreement between KGC_1 and KGC_2 after colluding to derived the shared session key between A and B. Since KGC_1 can compute : $xP=(a+s_1)^{-1}A_{KA}$ while KGC_2 can compute $yP=(b+s_2)^{-1}B_{KA}$,so for either party to escrow the session key, it is only required to cooperate with the counterpart, eventually the concerned KGC achieves the value $\hat{\imath}(xP,yP)$ just the same as A and B did. However KGC_1 and KGC_2 cannot collude to derive a valid session key for the escrow-less mode, where A and B share the key given by; $Key_{ab}= \hat{\imath}(P,Q)^{xy}=\hat{\imath}(B_{KA},A_{pri})^x$ and $Key_{ba}=\hat{\imath}(P,Q)^{xy}=\hat{\imath}(A_{KA},B_{pri})^y$ respectively, where $A_{pri}=(a+s_1)^{-1}Q$ and $B_{pri}=(b+s_2)^{-1}Q$. This is obvious because if KGC_1 wishes to escrow the session key, it colludes with KGC_2 only to get $yP=(b+s_2)^{-1}B_{KA}$ from KGC_2, while by its own powers and ability KGC_1 only manages to extract $xP=(a+s_1)^{-1}A_{KA}$ and so it can just manage to compute $\hat{\imath}(P,P)^{xy} \neq \hat{\imath}(P,Q)^{xy}$, contrary to A's and B's computations. This resilience to escrow problem is due to the ECDLP to determine individually the values x and y from xP and yP respectively.

Ⅲ. Attack on McCullagh et al.'s Protocol

In this section, we show that McCullagh et al.'s protocol is still vulnerable to masquerading attack due to the lack of sufficient authentication and assurance to integrity of the session key material.

Masquerading Attack : An attacker, Mallory situated between the communicating parties, Alice and Bob, is able to fabricate computations to establish a session key to masquerade as Bob to Alice as shown below.

Step 1: When A initiates a session by computing $A_{KA}=x(b+s_2)P$ and then send the computed message to B ,Mallory (M) intercepts the message before it reaches B.

Step 2: M then pick a random number, $m' \in_R Z_q^*$ and fabricate a message $M_{KA}=m'(b+s_2)P(a+s_1)P$ using both A's and B's public keys.

Step 3: M computes the shared key as; $Key_{ma}=\hat{\imath}(A_{KA},2P)^{m'}$ whilst A carry on with the normal way of computing shared as ; $Key_{am}=\hat{\imath}(M_{KA},A_{pri})^x$.

The attack is possible due to lack of proper authentication of party B's public key and its publicly sent message. Thus, Mallory is able to exploit the slightest weakness in the protocol to perform masquerading attack. The shared key agrees as follows:

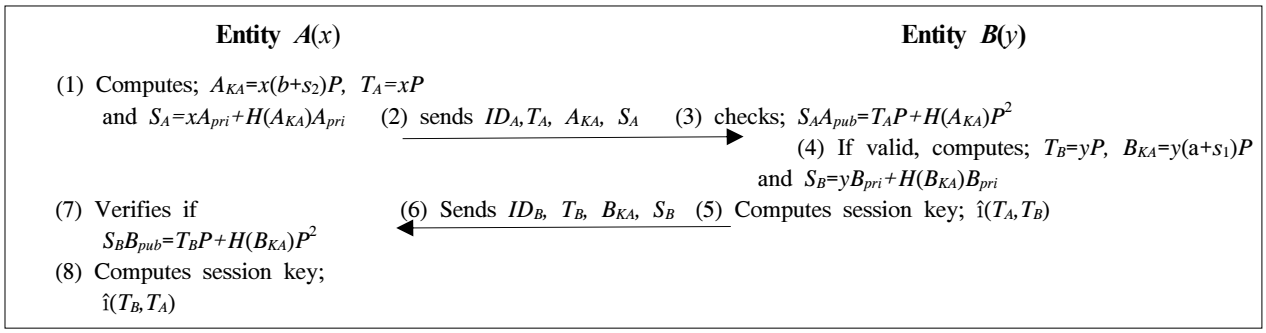

Fig. 1. Remedy of McCullagh et al.'s Protocol

Party A computes; $Key_{am}=\hat{\imath}(M_{KA},A_{pri})^x=\hat{\imath}(m'(b+s_2)P(a+s_1)P, (a+s_1)^{-1}P)^x$ which reduces to $\hat{\imath}(2m'(b+s_2)P,P)^x=\hat{\imath}(x(b+s_2)P,2P)^{m'}$ whilst on the other hand M fabricates, $Key_{ma}=\hat{\imath}(A_{KA},2P)^{m'}$. The computed keys agree since, $\hat{\imath}(x(b+s_2)P,2P)^{m'}=\hat{\imath}(A_{KA},2P)^{m'}$.

An adversary can launch a similar attack even on the escrow-less mode key agreement and manage to masquerade as B to A in a like manner. An interested reader can verify that the same attack is indeed possible in escrow-less mode.

Ⅳ. Remedy of AKA Protocol over Two Distinct Domains

In this section, we will discuss how to fix the flaw in McCullagh et al.'s AKA protocol over two separate KGCs. We already proposed a trivial fix to the flaws of the two-party ID-based AKA protocol in our early work for a conference paper in [21] by employing MAC and in which there were three message transmissions. Besides being a little inefficient MAC on number of handshakes, it does not secure fully the attack on the integrity of public key. Now we give a more formal and efficient remedy that just takes two message transmissions by employing a signature scheme, for A and B under distinct domains, which follows the same set up and extract as in preceeding protocol while key agreement algorithm proceeds as follows:

Key agreement procedure: For A and B to establish a secure and valid session key with two message flows, they carry out the following procedure.

Step 1: A initiates a session with B like this (a) Chooses a random ephemeral key $x\in Z_q^*$ and computes $T_A=xP$, $A_{KA}=x(b+s_2)P$ and a signature $S_A=xA_{pri}+H(A_{KA})A_{pri}$. (b) Sends $\{ID_A, T_A, A_{KA}, S_A\}$ to B.

Step 2: On receipt of A's message B does (a) Checks validity of the received message $\{ID_A, T_A, A_{KA}, S_A\}$ from A. Then B verifies the authenticity of the message with the help of the signature S_A as follows: B verifies if $S_AA_{pub}=T_AP+H(A_{KA})P^2$, by using A's public key ($A_{pub}=(a+s_1)P$), since $S_AA_{pub}=(xA_{pri}+H(A_{KA})A_{pri})A_{pub}=xP^2+H(A_{KA})P^2$. If this verification holds, it authenticates the source of the received message or else B quits the session. Where as the component $H(A_{KA})$ assures the integrity of the important part embedding the ephemeral key, for key agreement as we will see below. (b) Chooses a random ephemeral key $y\in Z_q^*$ and computes $T_B=yP$, $B_{KA}=y(a+s_1)P$ and its own signature too as $S_B=yB_{pri}+H(B_{KA})B_{pri}$; (c) Then computes the shared session key as $\hat{\imath}(T_A,T_B)$. (d) Then sends $\{ID_B, T_B, B_{KA}, S_B\}$ to A.

Step 3: On receipt of B's message A does;

(a) Verifies the authenticity of the signature by checking if, $S_B B_{pub}=T_B P+H(B_{KA})P^2$, that is if $S_B B_{pub}=(yB_{pri}+H(B_{KA})B_{pri})B_{pub}=yP^2+H(B_{KA})P^2$ else A quits the transaction. Specifically the past $H(B_{KA})$ checks the integrity of the ephemeral key and the use of B_{pub} to verify the signature, checks the authenticity of B.Notice that the verification processes simultaneously authenticates the message source and the integrity as well. (b) After verification of B's message A computes the shared session key as $\hat{\imath}(T_B, T_A)$.

The protocol's message flow is outlined in the Fig. 1 below. The session key agrees as follows: $SK_A=\hat{\imath}(T_B,T_A)=\hat{\imath}(P,P)^{xy}=\hat{\imath}(T_A,T_B)=SK_B$ by properties of bilinear pairing.

Ⅴ. Analysis

This section shows how the proposed remedy satisfies the required security attributes and also shows a comparison of efficiency with some related protocols.

5.1. Security Analysis

Now we show the security properties satisfied by the remedial protocol.

Known Key Security: The compromise of one session key will not guarantee the deduction of any other session keys either in the future or past. This follows simply from the random aspect of any distinct session key due to ephemeral keys x or y involved each time. As such each session key is unique and cannot be deducted from the other. Also either party outrightly uses the counter-part's public key to verify the authenticity of the source (the signature).

Key Compromise Impersonation resilience: In a case where the adversary (E) has A's long-term private key, it could be possible to only impersonate as A to other entities but never possible to impersonate as any other entity to A. This property is provided for, in our protocol due to the signature enhancement. If E wishes to impersonate some entity with identity, ID_i, then E must send ID_i, T_i, K_{KA}, S_i for that entity. Thus clearly E cannot forge $s_i=rK_{pri}+H(K_{KA})K_{pri}$ without knowledge of the private key of the entity K_{pri} . Obviously E has no private key of the entity he wishes to impersonate as to A, therefore it follows that she can't form a verifiable signature for this entity. Hence E wishing to carry out *key compromise impersonation* (KCI) attack to A will fail because of failing to create a verifiable signature due to lack of the signing private key. Thus the proposed protocol is resilient against KCI.

Unknown Key share resilience: Alice cannot be duped by an adversary, Mallory, to believe that she shares a key with Bob while in actual sense she shares it with Mallory, because she does signature verification explicitly by using the sender's public key (Bob's), which typically should be an authenticated one. So verification of the signatures; S_A and S_B implicitly authenticates both the terms, T_A and T_B and their corresponding identities ID_A and ID_B , which are used for session key computation. Since the protocol foils Eve's capability to forge a signature and hence fails to pass verification due to lack of respective private key, it means that the protocol achieves entity authentication as well as message authentication, therefore supporting *unknown key share resilience*.

Forward Secrecy: Compromise of a session key does not give a clue at all to an adversary to recover any past session keys, due to the ephemeral keys x or y which are fresh for each protocol run and are infeasible to compute by bruce force due to the ECDLP. Thus the underlying use of random ephemeral keys in say, $A_{KA}=x(b+s_2)P$ and $B_{KA}=y(a+s_1)P$ for each protocol run leads to different session key for each different session. Therefore possession of any one valid session key would not still reveal any other previous session key because of uniqueness of session keys per protocol run.

Key Control: Since both parties jointly contribute the

input of the session key, $T_A=xP$ and $T_B=yP$, none of them has the influence to preselect the value for the session key. All that an entity can manage to determine is to keep the key within certain desirable bits by carefully selecting the ephemeral session keep. It would be advisable therefore to set a short time out on a particular run of the protocol to avoid further manipulation of the agreed shared value.

Table 1. Performance comparison

Operation / Protocol	PA	EX	SM	MP	GA
Smart [8]	2	-	2	1	-
Chen et al. [10]	1	-	4	2	1
McCullagh et al. [14]	1	1	2	1	1
Our protocol	1	-	4	1	1

PA : Pairing, EX : Exponentiation, SM : Scalar Multuplication, MP : Map to a point, GA : Group Addition

5.2. Performance Analysis

Table 1 gives a comparison of computational operations of our protocol with other related ones. Comparatively with well known ID-based key exchange protocols in Table 1, our protocol is efficient while providing desirable security at the same time. Bearing in mind that the computational efficiency of hash function and point addition operations we therefore don't account for them, even so our protocol is a little more efficient than Chen et al.'s key agreement protocol, due to the fact that one-map to a point hash operation is also more expensive than one scalar multiplication[22]. In regard to the fact that pairing operation is at least 10 times more than scalar multiplication in the same field [22] hence our protocol fairs better than [8] efficiency. Even thought there is 2 more scalar multiplications than [14] in our protocol, since exponentiation is also a heavier computation, relatively there is a substantial compensation by the exponentiation cost in [14] in comparison to our protocol. In all the proposed remedy does not achieve the desired security at the cost of heavy computational overhead.

Ⅵ. Conclusion

AKA is a very important cryptographic primitive to establish a secure channel for two or more entities over an open network to secure subsequent communication by using the shared key. McCullagh et al. proposed a new two-party identity-based AKA using bilinear pairing to support either escrowed or escrow-less mode applicable even over two different $KGCs$ without imposing extra computational steps. However, we have shown how McCullagh et al.'s protocol is prone to masquerading attack. Further we presented an efficient protocol to fix the attack by employing message integrity mechanism and message source authentication primitive. The proposed remedy is secure and satisfies the required security attributes like: known key security, key compromise impersonation resilience, forward security, unknown key share resilience and key control resilience.

References

[1] 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, Mar. 2003.

[2] J. Qiuyan, K. Lee, and D. Won, "Cryptanalysis of a secure remote user authentication scheme," *J. Korea Inform. Commun. Soc. (KICS)*, vol. 37C, no. 8, pp. 697-702, Aug. 2012.

[3] H.-J. Seo and H.-W. Kim, "User authentication method on VANET environment," *J. Korea Inform. Commun. Soc. (KICS)*, vol. 37C, no. 7, pp. 576-583, July 2012.

[4] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Trans. Inform. Theory*, vol. 22, no. 6, pp. 644-654, Nov. 1976.

[5] D. Hankerson, A. Menezes, and S. Vanstone, *Guide to elliptic curve cryptography*, Springer-Verlag, 2004.

[6] A. Shamir, "Identity-based cryptosystems

signature schemes," *Lecture Notes in Computer Science*, vol. 196, pp. 47-53, August 1985.

[7] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," *Lecture Notes in Computer Science*, vol. 2139, pp. 213-229, August 2001.

[8] N. P. Smart, "Identity-based authenticated key agreement protocol based on Weil pairing," *IEEE Electron. Lett.*, vol. 38, no. 13, pp. 630-632, June 2002.

[9] K. Shim, "Efficient ID-based authenticated key agreement protocol based on Weil pairing," *IEEE Electron. Lett.*, vol. 39, no. 8, pp. 653-654, Apr. 2003.

[10] L. Chen and C. Kudla, "Identity based authenticated key agreement protocols from pairings," in *Proc. 16th IEEE Comput. Security Found. Workshop 2002*, pp. 219-233, Pacific Grove, U.S.A., June-July 2003.

[11] H.-M. Sun and B.-T. Hsieh, "Security analysis of Shim's authenticated key agreement protocols from pairings," *Cryptology ePrint Archive: Report 2003/113*, [Online], Available: http://eprint.iacr.org/2003/113/.

[12] E.-K. Ryu, E.-J. Yoon, and K.-Y. Yoo, "An efficient ID-based authenticated key agreement protocol from pairings," *Lecture Notes in Computer Science*, vol. 3042, pp. 1458-1463, August 2004.

[13] 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, Sep. 2005.

[14] N. McCullagh and P. S. L. M. Barreto, "A new two-party identity-based authenticated key agreement," in *Proc. Int. Conf. Topics Cryptology (CT-RSA '05)*, pp. 262-274, San Francisco, U.S.A., Feb. 2005.

[15] G. Xie, "Cryptanalysis of Noel McCullagh and Paulo S. L. M. Barreto's two-party identity-based key agreement," *Cryptology ePrint Archive: Report 2004/308*, [Online], Available: http://eprint.iacr.org/2004/308/.

[16] P. Kumar and H. Lee, "Security issues in healthcare application using wireless medical sensor network: a survey," *Sensors*, vol. 12, no. 1, pp. 55-91, Jan. 2012.

[17] J. Hoffstein, J. Pipher, and J. H. Silverman, *An introduction to mathematical cryptography*, Springer, 2008.v

[18] D. Merfert, "Bilinear Pairings in Cryptography," M.S. Thesis, Radboud Universitiet Nijmegen, Netherlands, 2009.

[19] X. Cao, W. Kou, and X. Du, "A pairing-free identity-based authenticated key agreement protocol with minimal message exchange," *Inform. Sci.*, vol. 180, no. 15, pp. 2895-2903, Aug. 2010.

[20] G. Frey, M. Muller, and H. Ruck, "The Tate pairing and the discrete logarithm applied to elliptic curves cryptosystems," *IEEE Trans. Inform. Theory*, vol. 45, no. 5, pp. 1717-1719, July 1999.

[21] T. F. Vallent, S.-W. Lee, E.-J. Yoon, and H. Kim, "Cryptanalysis and remedy of two-party identity-based authenticated key agreement protocol," in *Proc. KICS Winter Conf. 2013*, pp. 120-121, Yongpyeong, Korea, Jan. 2013.

[22] 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, no. 2, pp. 198-207, June 2007.

발렌트 토코자니 (Thokozani Felix Vallent)

2007년 말라위대학교 수학교육학과 졸업

2012년 3월~현재 경일대학교 IT융복합학과 석사과정

<관심분야> 정보보호, RFID보안, 클라우드컴퓨팅 보안

김 혜 정 (Hae-Jung Kim)

1987년 2월 경북대학교 수학과 학사
1989년 2월 경북대학교 전자공학과 석사
2005년 2월 경북대학교 컴퓨터공학과 박사
2005년~현재 계명대학교 교양학부 조교수

<관심분야> 정보검색, 정보보호, 컴퓨터교육

윤 은 준 (Eun-Jun Yoon)

2007년 2월 경북대학교 컴퓨터공학과 박사
2011년~현재 경일대학교 사이버보안학과 교수
2009년~2011년 경북대학교 대학원 전기전자컴퓨터학부 계약교수

<관심분야> 암호학, 정보보호, 유비쿼터스보안, 네트워크보안, 스테가노피, 인증프로토콜

김 현 성 (Hyunsung Kim)

2002년 2월 경북대학교 컴퓨터공학과 박사
2012년 3월~현재 경일대학교 사이버보안학과 교수
2010년 현재 정보융합보안연구소 소장
2012년 현재 경일대학교 학술정보원 원장

<관심분야> 인지무선네트워크 보안, 네트워크 보안, 암호 프로토콜, 암호구현, 정보보호