情報保護學會論文誌
第14券 第5號, 2004. 10

인증서 기반이 아닌 효율적인 공개키 암호화 기법

이 영 란†, 이 향 숙‡
이화여자대학교

An Efficient Certificateless Public Key Encryption Scheme

Young-Ran Lee† and Hyang-Sook Lee‡
Ewha Womans University

요 약

Al-Riyami 와 Paterson[1]은 Certificateless 공개키 시스템이라 부르는 새로운 공개키 패러다임을 제안하였다. 이 시스템은 공개키 암호기법과 ID-기반 암호기법 각각의 장점을 가지고 있다. 즉, 기존의 공개키 기반 구조상의 인증서를 필요로 하지 않으면서도 ID기반 암호 시스템의 본질적 성질인 키위탁(key escrow) 관련 파생문제를 가지지 않는다. 본 논문에서 우리는 양방향 사용자 인증을 만족하는 인증서를 사용하지 않는 효율적 암호 스킴을 제안한다. 제안하는 스킴의 안전성은 computational Diffie-Hellman 문제(CDHP)와 bilinear Diffie-Hellman 문제(BDHP)의 어려움에 기반한다. 기밀성과 위조 불가능성을 위한 안전성을 증명하기 위하여 모델을 설정하고 제안된 스킴이 랜덤한 오라클(random oracle) 모델에서 안전함을 증명한다.

ABSTRACT

Al-Riyami and Paterson[1] suggested the new public key paradigm which is called the certificateless public key system. This system takes the advantages of both traditional PKC and ID-based PKC. It does not require the use of certificates of the public key and does not have the key escrow problem caused from the ID-based cryptosystem. In this paper, we propose an efficient certificateless public key encryption scheme which satisfies mutual authentication. The security of our protocol is based on the hardness of two problems; the computational Diffie-Hellman problem(CDHP) and the bilinear Diffie-Hellman problem(BDHP). We also give a formal security model for both confidentiality and unforgeability, and then show that our scheme is probably secure in the random oracle model.

Keywords : *certificateless public key cryptosystem, confidentiality, unforgeability*

1. 서 론

1984년 Shamir는 identity(ID)를 기반으로 하는 암호 시스템의 개념을 제안하였다[15]. 2001년이 되어서야 Boneh와 Franklin[2,3]이 실용적인 ID 기반의 암호시스템을 발표하였는데 이 논문에서 Boneh와 Franklin은 초특이 타원곡선에서 bilinear 사상을 이용하여 실용적이고 증명 가능한 ID 기반의 암호 스킴을 설계하였다. ID 기반의 암호 스킴 방법은 사용자의 신원을 나타낼 수 있는 e-mail 주소나 IP 주소, 주민번호 등을 이용한 공개키를 사용함으로써 공개키 인증을 생략하는 것이다. 이러한 시스템이 운용되기 위하여 시스템의 매개변수와 관련

접수일 : 2004년 8월 11일 ; 채택일 : 2004년 10월 12일
* 본 연구는 2003년도 과학재단 여대기반 사업의 연구비 지원에 의해 수행되었음.
† 주저자 : sens1990@yahoo.co.kr
‡ 교신저자 : hsl@ewha.ac.kr

된 마스터키와 비밀키를 생성할 수 있는 믿을만한 비밀키 생성센터(PKG)가 존재해야 한다. ID 를 기반으로 하는 시스템은 Boneh와 Franklin 논문 이후 암호화 스킴[1,2,3,13], 서명스킴[4,11,14,18,19], 키공유 프로토콜[6,7,16], signcryption[12] 등과 관련하여 많이 발표되어 왔다. 가장 잘 알려진 전통적인 공개키 기반구조의 문제는 철회, 보관, 검증 등의 공개키 인증을 포함한 키 관리에 있다. 이러한 문제들을 고려할 때 ID 기반의 공개키 시스템은 전통적인 공개키 기반구조와 비교 했을 때 장점을 갖는다. 한편 ID 기반의 공개키 시스템은 PKG와 관련하여 단점을 갖는데, 각 사용자의 비밀키를 생성하는 특권을 갖는 PKG에 대한 의존이 키 위탁 문제를 야기할 수 있기 때문이다. 예를 들면 개인의 프라이버시를 침해하거나 정직하지 못한 PKG가 사용자의 비밀 키를 이용해 합법적인 사용자인 것처럼 위장하는 것이다. 이러한 점들을 고려하여 기존의 PKI(public key infrastrucutre) 기반의 계층구조 개념을 ID기반 시스템에 적용한 Hierarchical ID-based 암호시스템에 관한 연구도 활발히 이루어지고 있는데 Gentry와 Silverberg의 논문[10]이 대표적이라고 할 수 있겠다. 최근 국내에도 이와 관련된 효율성을 고려한 논문이 김태구외 3인에 의해 발표되었다[17].

가장 실제적인 ID 기반 암호시스템으로 평가 받는 Boneh-Franklin(이하 BF로 표기)의 논문이 나온 이후, 이것을 인증측면에서 확장시킨 인증된 암호화 스킴(Authenticated Encryption scheme)이 Lynn에 의해 제안되었다. Lynn은 [13]에서 BF의 스킴이 사용자에 대한 인증이 고려되지 않은 점에 착안하여 암/복호화 과정에서 관련된 수신자의 공개키/비밀키 정보를 적절히 사용하는 방법을 사용하여 인증성을 추가하고자 했다. 그러나 이미 언급한 바와 같이 ID-기반 시스템에서 비밀키를 생성해서 사용자의 비밀키를 알고 있는 키생성 센터(PKG)가 통신자를 위장(impersonation) 할 수 있는 상황은 발생할 수 있고 이런 경우 사용자 인증은 무의미하게 된다. 따라서, 본 논문에서는 Al-Riyami와 Paterson에 의해 제안된 시스템의 특징-ID기반과 인증서 기반의 장점을 취하는 성질-을 받아들이면서 키생성 센터를 배제한 통신 당사자 쌍방만이 계산 가능한 정보를 넣어 암/복호화 하는 과정을 통해 송/수신자 양 쌍방의 사용자 인증을 만족하면서 인증서 기반이 아닌 (certificateless) 암호화 스킴을 제안하고자 한다. 위탁문제를 극복하기 위하여 제안된 모델에서 사용자는 PKG가 알 수 없는 short term 키로부터 Diffie-Hellman 키 공유 방식을 이용한다. 또한 long term 키는 부인봉쇄와 인증을 위하여 쓰인다. 결과적으로 우리의 스킴은 PKG로부터 기밀성을 유지하고 암호문 위조 방지 성질까지 주는 장점을 갖는다. 더구나 또 다른 장점은 PKG로부터 마스터키의 노출이 있어도 암호화된 원문의 기밀성이 보장된다는 점이다. 우리 시스템의 안전성은 계산적 Diffie-Hellman 가정과 bilinear Diffie-Hellman (BDH) 가정에 기반한다. 이 가정을 기반으로 우리 스킴은 무결성을 위한 EUF-CMA 안전성을 제공하고 비밀성 보장을 위한 IND-CCA 안전성을 제공한다. 효율성에 있어서는 Al-Riyami와 Paterson의 스킴과 비교했을 때 프로토콜에서 두 번의 pairing 계산량을 줄일 수 있다.

이 논문의 구성은 다음과 같다. 2장에서는 기본적인 개념과 정의를 설명하고 스킴의 안전성 개념을 논의한다. 3장에서는 제안된 스킴이 CDHP와 BDHP가 계산적으로 어렵다는 것을 가정했을 때 선택암호문 공격으로부터 안전할 뿐 아니라 임의의 오라클 모델에서 능동적 선택문 공격하에서 위조가능하지 않음을 증명한다. 또한 4장에서는 제안된 스킴의 안전성을 분석한다. 5장에서는 발표된 다른 스킴들과 안전성 및 효율성을 비교한다. 마지막으로 6장에서는 결론을 서술한다.

2. 서 문

2.1. 기본 정의 및 배경

먼저 우리가 제안하고자 하는 공개키 스킴에서 중요한 역할을 하는 수학적 프리미티브인 허용가능한 겹선형 사상(admissible bilinear map)을 살펴보기로 한다.

겹선형 사상(Bilinear map). G_1을 소수위수 q를 갖는 덧셈군이라 하고, G_2를 같은 위수 q를 갖는 곱셈군이라 하자. P를 G_1의 생성자라고 두자. 이산 대수 문제(discrete logarithm problem:DLP)가 G_1과 G_2에서 모두 어렵다고 가정할 때 다음 성질들을 만족하는 사상 $\hat{e} : G_1 \times G_1 \rightarrow G_2$을 허용가능한 겹선형 사상(admissible bilinear map)이라 부른다.

1. 겹선형성 (Bilinearity) : 모든 $P, Q \in G_1$와

$a,b \in Z_q^*$에 대해 $\hat{e}(aP,bQ)=\hat{e}(P,Q)^{ab}$.

2. 정상성 (Non-degeneracy) : $\hat{e}$은 $G_1 \times G_1$ 상의 모든 점을 G_2상의 항등원으로 대응시키지 않는다. (결과적으로, 만일 P가 G_1상의 생성자라면, $\hat{e}(P,P)$는 G_2상의 생성자가 된다.)
3. 계산 가능성 (Computability) : 모든 $P, Q \in G_1$에 대해 $\hat{e}$을 계산하는 효율적인 알고리즘이 존재한다.

일반적으로 $\hat{e}$은 유한체상의 타원곡선 위에서 정의되는 Weil이나 Tate pairing으로부터 파생된다. 우리 스킴의 안전성은 computational Diffie-Hellman 문제와 Bilinear Diffie-Hellman 문제의 어려움에 기반한다. 이제 안전성 기반이 되는 어려운 문제들에 대한 정형화된 설명을 살펴보기로 하자.

Computational Diffie-Hellman 문제. 어떤 $a,b \in Z_q^*$ 에 대해 P, aP, bP이 주어졌을 때, abP를 계산한다.

Computational Diffie-Hellman 가정. 다항식 예상 실행 시간내에 상당한 확률을 가지고 CDH 문제를 해결할 수 있는 알고리즘은 존재하지 않는다.

Bilinear Diffie-Hellman 문제. G_1과 G_2를 소수위수 q를 갖는 군이라고 하자. $\hat{e}: G_1 \times G_1 \rightarrow G_2$를 허용가능한 겹선형사상(admissible bilinear map)이라 두고 P를 G_1의 생성자라 하자. G_1과 G_2상의 BDH 문제는 다음과 같다. : 어떤 $a,b,c \in Z_q^*$에 대해, $<P, aP, bP, cP>$가 주어졌을 때, $W=\hat{e}(P,P)^{abc} \in G_2$를 계산한다.

$\Pr[A(P,aP,bP,cP)=\hat{e}(P,P)^{abc}] \geq \epsilon$일 때, 알고리즘 A는 G_1, G_2상의 BDH 문제를 해결하는데 ϵ만큼의 이점을 가진다고 한다. 단 이때의 확률은 임의의 원소 $a,b,c \in Z_q^*$의 선택과 알고리즘 A의 임의의 비트(bit)선택에 따라 결정된다.

Bilinear Diffie-Hellman 매개변수 생성자. [BF1, BF2]에서와 같이, IG 가 안전매개변수 k를 입력값으로 취하고 k에 관한 다항식 시간내에 수행하여, 소수 q, 소수위수 q의 군 G_1, G_2와 겹선형사상 $\hat{e}: G_1 \times G_1 \rightarrow G_2$에 관한 값을 출력할 때 난수화된 알고리즘 IG를 BDH 매개변수 생성자라 한다.

Bilinear Diffie-Hellman 가정. 모든 확률적 다항식 시간 알고리즘 A에 대해 만일 아래의 확률값이 무시해도 좋을 정도라면(negligible) BDH 매개변수 생성자 IG는 BDH 가정을 만족한다고 한다.

$$\Pr[(G_1, G_2, \hat{e}) \leftarrow IG(1^k) P \leftarrow G_1; a,b,c \leftarrow Z_q^* : A(G_1, G_2, \hat{e}, P, aP, bP, cP) = \hat{e}(P,P)^{abc}]$$

본 논문의 나머지 부분에서, 우리는 BDH 가정을 만족하는 고정된 BDH매개변수 생성자로 IG를 쓰고 그것의 출력 값을 표현하는데 있어 G_1, G_2, $\hat{e}$, q 를 쓰기로 한다.

2.2. 인증서 기반이 아닌 공개키 암호화 기법

앞서 언급한 바와 같이 본 논문에서 제안하는 스킴은 Al-Riyami와 Paterson[1]의 인증서 기반이 아닌 공개키 암호화 기법(CL-PKE)과 같은 선상에 있다고 볼 수 있다. 이 절에서는 [1]에서 정의한 공개키 암호화 기법의 일반적 정의를 살펴본다.

- 인증서 기반이 아닌 공개키 암호화 기법(Al-Riyami와 Paterson의 일반적 정의)

CL-PKE는 다음의 7단계로 구성된다.

- 설정(setup): 이 단계는 PKG(키생성센터)에 의해 이루어지며 기본적으로 시스템 변수와 마스터키(master key)를 설정하는 과정이다.
- 부분 비밀키 추출(partial-private-key-extract): PKG에 의해 수행되는 단계로서 각 사용자 A의 부분 비밀키(partial private key) D_A를 설정해서 해당 사용자 A에게 비밀이 보장되는 인증된 채널을 통해 전송되어 진다.
- 비밀값 설정(set-secret-value): 사용자 A가 시스템 변수와 자신의 신분식별정보(identifier) ID_A를 입력값으로 하여 향후 비밀키에 쓰일 비밀값 x_A를 출력한다.
- 비밀키 설정(set-private-key): 사용자 A는 시

스템 변수, 자신의 부분 비밀키값 D_A와 비밀값 x_A를 입력값으로 하여 비밀키 S_A를 출력한다.

비밀값 설정과 비밀키 설정 과정에서 알 수 있듯이, 비밀값 x_A와 비밀키 D_A는 사용자 A에게만 알려진 정보이다.

- 공개키 설정(set-public-key): 사용자 A는 시스템 변수와 자신만의 비밀값 x_A를 입력값으로 하여 공개키 P_A를 출력한다. (단, 이때의 x_A는 비밀키 설정 단계의 출력값이며, 비밀키 설정 과정에서 쓰인 값과 같다. 대체적으로 x_A값은 난수로서 적당히 큰 집합에서 선택되어진다.)
- 암호화(encrypt) : 송신자 B는 시스템 변수, 평문 M, 수신자의 공개키 P_A와 개인 신분 식별정보(identifier)를 입력값으로 취해서 암호문을 출력하거나 암호화가 실패했을 경우에는 실패를 의미하는 상징 $\perp$를 출력한다. 암호화가 실패하는 경우는 암호화의 입력값으로 사용되는 공개키 P_A가 부적절한 형태일 때 발생하게 된다.
- 복호화(decrypt) : 수신자 A는 전달된 암호문 C를 복호화 하기위해 시스템 변수, 자신의 비밀키 S_A, 주어진 암호문 C를 입력값으로 하여 평문 M 혹은 복호화 실패를 의미하는 상징 $\perp$를 출력한다.

시스템 변수, 비밀키 S_A, 암호문 C를 입력값으로 해 결과값으로 나온 평문 M은 시스템 변수, 수신자 공개키 P_A와 개인 식별 정보 ID_A, 평문 M을 입력값으로 해서 암호화과정을 거쳐서 생성된 암호문 C상에 복호화 과정을 적용해서 나온 결과여야만 한다.

2.3. 안전성 개념

다음 절에서, 우리는 BDH가정, CDH 가정에 근거한 랜덤 오라클(random oracle) 모델과 Fujisaki-Okamoto 변환[8]을 이용하여 제안하는 스킴의 기밀성(confidentialiy)과 위조 불가능성(unforgeability)을 증명한다. 이를 위해 먼저 기밀성과 위조불가능성에 대한 정형화된 정의들을 살펴보기로 한다.

2.3.1 기밀성 (Confidentiality)

만일, 다음 게임에서 다항식적으로 제한된 공격자가 challenger에 대항하여 어떤 이점도 가질 수 없을 때 그 스킴은 IND-CCA 안전성을 가진다고 말한다.

준비(Setup). challenger는 안전매개변수 k를 취해, setup 알고리즘을 수행한 후, 공격자에게 $P_{pub} = sP$를 만족하는 값 s와 매개변수들을 제공한다.

본 논문에서 제안하는 스킴은 ID-기반은 아니지만 시스템 상에 사용자들의 long-term 비밀 키를 발행하는 제3기관(PKG)이 존재한다. 이런 기관(PKG)의 부정을 방지하기 위해, 우리의 안전성 모델은 다른 암호화 스킴들의 안전성 모델보다 공격자들을 제어하는데 있어 좀 더 강화되어졌다. 서론에서 언급한 바와 같이 본 논문에서 제안하는 암호화 스킴은 Al-Riyami/Paterson의 certificateless 공개키 패러다임 형식이므로 기존의 안전성 증명 모델과는 다른 형태가 요구 되어진다. 간략하게 말하자면, 우리는 공격자가 마스터키에 접근할 수 있음을 가정한다.

단계 1. 공격자는 질의(query) $q_1, q_2, \cdots, q_m$을 요청할 수 있다. 단, q_i는 다음 질의 중의 하나이다.

ID_I형태의 비밀 키 질의(Extraction query). 이런 형태의 질의를 요청 받았을 때 challenger는 알고리즘 Extract(ID_I)를 수행해서 $S_I = x_I d_I$ (단, d_I는 PKG에 의해 생성된 long-term 비밀 키이다.)로 응답한다.

(ID_I, ID_J, M)형태의 암호화 질의 (Encryption query). 이런 형태의 질의를 요청 받았을 때, challenger는 Extract(ID_I)를 수행해서 S_I 값을 얻은 후, 알고리즘 Encrypt(S_I, ID_J, M)을 수행한다. 그 응답 값으로 암호문을 준다.

(ID_I, ID_J, C)형태의 복호화 질의(Decryption query). 이런 형태의 질의를 요청 받았을때, challenger는 알고리즘 Extract(ID_J)을 수행해서 출력값 S_J를 얻은 후 알고리즘 Decrypt(ID_I, S_J, C)를 수행한다. 응답 값으로 평문 M을 준다.

이런 일련의 질의들은 능동적으로(adaptively) 요청되어 질 수 있다. 즉, 각 질의 q_i는 응답들

$q_1, q_2, \cdots, q_{i-1}$에 따라 변할 수 있다.

Challenge. 공격자는 일단 단계 1이 끝났다고 결정한 후, 자신이 도전(challenge)하길 원하는 두 개의 같은 길이를 갖는 평문 $M_0, M_1 \in M$와 두 개의 identity ID_A, ID_B를 결과 값으로 낸다. 이 때의 제한점은 challenge ID가 단계 1 상의 어떤 Extraction query에도 나타난 것이 아니어야 한다는 것이다. challenger는 임의의 비트(bit) $b \in \{0,1\}$를 뽑고 알고리즘 Extract(ID_A)를 수행한 후 Encrypt(S_A, ID_B, M_b)를 수행한다. 결과 값인 암호문 C^*를 공격자에게 응답 값으로 준다.

단계 2. 이 단계 동안에, 공격자는 다음의 제한점을 가지고서 단계 1 상에 실행된 형태의 질의들을 $q_{m+1}, \cdots, q_n$ 좀 더 요청할 수 있다.

- ID_A와 ID_B에 관한 비밀키 질의(Extraction query)는 허용되지 않는다.
- (ID_A, ID_B, C^*) 형태의 복호화 질의(Decryption query)는 허용되지 않는다.

이외의 형태의 질의들은 Phase 1에서 처럼 능동적으로(adaptively) 요청될 수 있고, challenger는 마찬가지로 응답을 계속한다.

추측(Guess). 마지막으로 공격자는 비트 $b' \in \{0,1\}$을 결과값으로 내고, $b' = b$일 경우 이 게임에서 이기게 된다.

우리는 그런 유형의 공격자를 IND-CCA 공격자(attacker)라고 부르기로 한다. 공격자 A의 이점을 $Adv(A) = |\Pr[b = b'] - \frac{1}{2}|$로 정의하기로 하자. 이때의 확률은 challenger와 공격자에 의한 임의의 비트(bit)에 의존한다.

2.3.2 위조 불가능성(Unforgeability)

만일 다음 게임에서 다항식적으로 제한된 공격자가 어떤 이점도 얻을 수 없다면 그 스킴이 암호문 위조에 대해 안전하다고 말한다.

준비(Setup). challenger는 안전매개변수 k를 취해, 매개변수들과 마스터키 s를 얻기 위해 Setup 알고리즘을 수행한다. challenger는 공격자에게 $P_{pub} = sP$를 만족하는 값 s를 매개변수와 함께 제공한다.

공격(Attack). 이 단계에서 공격자는 challenger에게 다음과 같은 질의(query)들을 요청할 수 있다.

ID_I 형태의 비밀 키 질의(Extraction query). 이런 형태의 질의를 받았을 때 challenger는 알고리즘 Extract(ID_I)을 수행하고 그 결과값인 $S_I = x_I d_I$로 응답한다.(단, d_I는 PKG에 의해 생성된 long-term 비밀키이다.)

(ID_I, ID_J, M) 형태의 암호화질의(Encryption query). 이런 형태의 질의를 받았을 때 challenger는 Extract(ID_I)를 수행한 후, Encrypt(S_I, ID_J, M)를 수행한다. 질의에 대한 응답 값으로 수행 결과 값인 암호문을 준다.

(ID_I, ID_J, C) 형태의 복호화질의(Decryption query). 이런 형태의 질의를 받았을 때 challenger는 Extract(ID_J)를 수행한 후, Decrypt(ID_I, S_J, C)를 수행한다. 질의에 대한 응답 값으로 수행 결과 값인 평문 M을 준다. (때때로 공격자는 질의한 암호문 C가 무효(invalid)하다는 통보를 받기도 한다.)

위조. 이전의 과정에서 질의된 A와 B의 비밀키 관련 Extraction query가 아니라는 제한아래, 공격자는 송신자 A로부터 수신자 B에게로 가는 임의의 유효한 암호문(valid ciphertext) C를 만들어 내길 시도한다. 만일 그 해당 암호문이 유효(valid)하다면 공격자는 이기게 된다.

우리는 이런 유형의 공격자를 EUF-CMA 공격자(existential unforgeable chosen message attacker)라고 부르기로 한다.

3. 인증서 기반이 아닌 효율적인 암호 스킴

본 논문에서 제안하는 스킴은 다음의 네가지 알고리즘 : Setup, Key Extraction, Encrypt, Decrypt 로 구성된다.

준비(Setup) : 안전 매개변수 k가 주어졌을때,

setup 알고리즘은 다음과 같이 수행된다.

(1) 입력값으로 k를 받아 IG를 수행하여 소수 q, 소수 위수 q를 갖는 군 G_1, G_2, 허용가능한 겹선형 사상 $\hat{e}: G_1 \times G_1 \rightarrow G_2$를 출력한다. 임의의 G_1의 생성자 P를 선택한다.

(2) 랜덤하게 $s \in Z_q^*$를 뽑고, $P_{pub} = sP$라고 둔다.

(3) 암호학적 해쉬함수 $H_1: \{0,1\}^* \rightarrow G_1^*$, $H_2: H_2: G_1^* \rightarrow \{0,1\}^n$, $H_3: \{0,1\}^n \times G_2 \rightarrow \{0,1\}^n$, $H_4: \{0,1\}^n \times \{0,1\}^n \rightarrow Z_q$, $H_5: \{0,1\}^n \rightarrow \{0,1\}^n$ 들을 선택한다. 일련의 과정이 끝난 후 시스템 매개변수들인 $G_1, G_2, q, \hat{e}, P, P_{Pub}, H_1, H_2, H_3, H_4, H_5$와 마스터키 s를 출력 값으로 얻는다. 평문 공간은 $M = \{0,1\}^n$이고 암호문 공간은 $C = G_1 \times \{0,1\}^n \times \{0,1\}^n$이다.

키 추출(Key Extraction): 문자열 $ID \in \{0,1\}^*$이 주어졌을 때, 알고리즘은 다음을 수행한다.

(1) $Q_{ID} = H_1(ID) \in G_1^*$을 계산한다.

(2) 난수 $x_{ID} \in Z_q$을 선택하고, 공개키를 $X_{ID} = x_{ID}P$, $Y_{ID} = x_{ID}Q_{ID}$ 로 둔다.

(3) 비밀키를 $d_{ID} = sQ_{ID}, S_{ID} = x_{ID}d_{ID} = x_{ID}sQ_{ID}$ 로 둔다. (단, 이때 s는 마스터키이다.)

이상의 준비(Setup)와 키추출 과정(Key extraction)이 끝난 뒤, 제안하는 프로토콜을 이용해 비밀통신을 하려는 통신 당사자들은 사전에 쌍방의 ID (사용자 고유의 정보를 나타내는)정보와 공개키 정보인 X_{ID}, Y_{ID} 들을 알고 있다고 가정하자.

암호화(Encrypt): 평문 $M \in M$을 암호화하기 위해 다음을 수행한다.

(1) 임의의 $\sigma \in \{0,1\}^n$를 선택한다.

(2) $r = H_4(\sigma, M)$라 둔다.

(3) $x_A X_B = T$를 계산한다.

(4) 암호문을 $C = < rQ_A, \sigma \oplus H_3(H_2(T), \hat{e}(d_A, Y_B)^r), M \oplus H_5(\sigma) >$로 둔다.(단, 이때 Y_B는 수신자의 공개키이다.)

복호화(Decrypt): $C = < U, V, W > \in C$를 전송된 암호문이라고 하자. 이 암호문을 비밀키 $S_B = x_B d_B$를 이용하여 복호화 하기 위해 다음을 수행한다.

(1) $x_B X_A = T$를 계산한다.

(2) $V \oplus H_3(H_2(T), \hat{e}(U, S_B))$를 계산하여 σ를 구한다.(단, $S_B = x_B d_B$이다.)

(3) σ를 이용하여 $W \oplus H_5(\sigma)$를 계산해서 M을 구한다.

(4) $r = H_4(\sigma, M)$라고 두고 $U = rQ_A$인지 검증한다. 만약 수식을 만족하지 않는다면, 해당 암호문을 폐기한다.

(5) 수식을 만족할 경우 M을 암호문 C의 복호화된 값으로 출력한다.

겹선형성(bilinearity)에 의해 송/수신자간의 계산상의 일치성은 다음과 같이 쉽게 체크된다.

$$\hat{e}(d_A, Y_B)^r = \hat{e}(sQ_A, x_B Q_B)^r = \hat{e}(rQ_A, x_B Q_B)^s$$
$$= \hat{e}(rQ_A, x_B s Q_B) = \hat{e}(U, x_B d_B) = \hat{e}(U, S_B)$$

수신자는 수식 $rQ_A = U$가 성립하는지를 체크함으로서 암호화된 메시지의 발신지(origin)를 확신할 수 있게 된다. 설사 수신된 메시지가 잘못된 공개키로 암호화된 것일지라도, 수신자는 마지막 단계의 수식을 테스트함으로서 오류를 간파해 낼 수 있다.

4. 스킴의 안전성 분석

4.1. 무결성(Integrity) 증명

다음의 정리는 G_1, G_2상의 BDH 문제의 어려움과 G_1상의 CDH 문제의 어려움을 가정 했을 때 이 논문에서 제안하는 스킴이 키위탁(key escrow) 성질을 배제하면서도 암호문의 위조에 안전함을 설명한다.

[정리 1] 해쉬함수 H_1, H_2, H_3, H_4, H_5를 랜덤 오라클(random oracle)로 두자. 그러면 IG에 의해 형성된 군(group)들 상에서 BDH 문제와 CDH 문제의 어려움을 가정했을 때, 우리 스킴은 암호문 위조가 불가능한 공개키 암호화 기법이 된다. 구체적으로 말하자면, A를 ϵ만큼의 이점을 가지고 암호문을 위조할 수 있으며, 동시에 많아야 q_E번의 Extraction query와 H_1, H_2, H_3 각각에 대하여 $q_{H_1}, q_{H_2}, q_{H_3}$만큼의 Hash query를 요청하는 다항식적으로 제한된

공격자가 존재한다고 가정한다고 하면 $\epsilon / \binom{q_{H_1}}{2}^2 q_D$ 만큼의 이점을 가지고 BDH 문제와 CDH 문제를 해결하는 다항식적으로 제한된 알고리즘 B가 존재한다.

(증명)

알고리즘 B는 랜덤하고 균일하게 분포된 BDH 문제와 CDH 문제 각각의 instance(P,aP,bP,cP), (P,xP,yP)를 입력으로 받아들인다. 알고리즘 A의 도움을 받아 $\hat{e}(P,P)^{abc}, xyP$ 값들을 구하기 위해, B는 해쉬함수 H_1, H_2, H_3들을 제어한다. 각각의 Hash query에 응답하기 위해 B는 목록 $L_{H_1}, L_{H_2}, L_{H_3}$를 만들어서 보관 유지한다. 이때의 목록 각각은 해당 해쉬함수에 대해 요청된 Hash query에 대한 정보들을 저장하고 있으며, 초기값은 없는 것으로 생각한다. 편의상 모든 H_1-query는 다르며(응답은 저장된다), ID_A와 관련된 모든 질의에는 항상 ID_A에 관한 H_1-query가 선행되어야 함을 가정한다.

그밖에 복호화 오라클(decryption oracle)과 상호 작용함에 있어서 공격자 A의 행위에 대해 몇 가지 가정을 다음과 같이 두기로 한다.

- A는 자신의 추측 값(guess)을 출력하기 전에 그 값에 관련된 복호화 질의(decryption query)를 요청해야 한다.
- A는 암호화 오라클(encryption oracle)로부터 응답으로 받은 암호문이나 자신이 계산 할 수 있는 암호문에 관해서는 복호화 질의를 요청할 수 없다. 후자가 나타내는 암호문의 경우는, 공격자 A가 송/수신자의 키 관련 비밀키 질의(Extraction query)를 요청해서 그 응답 값으로 비밀키 정보를 가지고 있으므로 암호화가 가능할 수 있는 경우를 의미한다.
- 위의 가정들이 주어진 상황에서, 암호문에 대한 모든 복호화 질의(decryption query) 후에 만일 그 응답 값이 평문(plaintext)이라면,(즉 질의된 암호문이 유효한 경우) A는 게임을 멈추고 응답으로 받은 평문에 상응하는 암호문을 출력한다.

B는 A와 다음과 같이 상호작용을 한다.

<u>준비(Setup)</u> : 게임을 시작할 때, B는 A에게 시스템 매개변수(system parameter)들 $< G_1, G_2, q, \hat{e}, P, P_{pub}>$, $H_1 H_2, H_3, H_4 H_5 >$과 $P_{pub} = sP$를 만족하는 값 s를 제공한다.

<u>H_1-질의</u> : B는 서로 다른 I와 J를 1과 q_H사이에서 선택한다. A가 자신의 선택에 따른 Identity들로 H_1-query를 다항식적으로 제한된 수만큼 요청될 때 B는 다음과 같이 응답한다.

(i) I번째 H_1-query에 대해 B는 $b_I Q_I$로 응답한다. 단, Q_I는 임의의 공개값이다. 좀 더 정확히 말하자면, 만약 ID_A가 리스트상에 존재하지 않으면서 A에 의해 요청된 서로 다른 I번째 H_1-query일 경우 B는 $b_I \in Z_q^*$를 선택하고, 리스트 L_{H_1}내에 $< ID_I, b_I Q_I, b_I, \perp >$를 추가한 후, $H_1(ID_I) = b_I Q_I$를 응답으로 준다.

(ii) J번째 H_1-query에서 B는 $b_J Q_J$로 응답한다. 단, Q_J는 임의의 공개된 값이다. 좀더 정확히 말하자면, 만약 ID_A가 리스트상에 나타나지 않고 A에 의해 요청된 J번째의 서로 다른 H_1-query일 경우 B는 $b_J \in Z_q^*$를 선택하고, 리스트 L_{H_1}내에 $< ID_J, b_J Q_J, b_J, \perp >$를 추가한 후, $H_1(ID_J) = b_J Q_J$를 응답으로 준다.

(iii) $H_1(ID_e)$(단, $e \neq I, J$)에 대해 B는 $b_e, \alpha_e \in Z_q^*$를 선택하고, 리스트 L_{H_1}내에 $< ID_e, b_e P, b_e, \alpha_e >$를 추가한 후, $H_1(ID_e) = b_e P$를 응답으로 준다.

<u>H_2-질의</u> : ID_A, ID_B에 대한 H_2-query는 다음과 같이 처리된다.

(i) 만약 ID_A와 ID_B가 ID_I, ID_J가 아니라면 B는 $\alpha_A \alpha_B P$를 계산하고, 리스트 L_{H_2}에 $< ID_A, ID_B, \alpha_A \alpha_B P, h_2 >$를 추가하고 h_2로 응답한다.

(ii) ID_A와 ID_B가 ID_I, ID_J라면, B는 $z \in Z_q^*$를 선택하고, 리스트 L_{H_2}에 $< ID_I, ID_J, zP, h_2 >$ 추가한 후, $h_2 = H_2(zP)$를 응답으로 준다.

<u>H_3-질의</u> : A는 언제든지 (h_2, U, ID_A, ID_B)로서 H_3-query를 요청할 수 있다. B는 A의 질의에 응답하기 위해 다음과 같이 H_3-simulation 알고리즘을 수행한다.

(i) $ID_A = ID_I$이고 $ID_B = ID_J$인 경우, B는 리스트 L_{H_2}를 점검하여 어떤 α에 대해 $< ID_A, ID_B, \alpha P, h_2 >$형태가 있는지를 살펴본다.

- 만일 그런 형태가 리스트에 있다면, 리스트상의 구체적인 형태는 $< ID_I, ID_J, zP, h_2 >$이다. 다음으로 B는 $d^* \in G_1^*$를 랜덤하게 선택하고, $\hat{e}(U, d^*) = w$을 계산해서 $h_3 = H_3(h_2, w)$를 설정한 후, 리스트 L_{H_3}에 $< ID_I, ID_J, U, (h_2, w), h_3 >$을 추가하고 응답 값으로 h_3를 준다.
- 만일 그런 형태가 리스트에 없을 경우에, B는 $z' \in Z_q^*$을 선택하고 리스트 L_{H_3}에 $< ID_I, ID_J, z'P, h_2' >$를 추가한다. 앞의 경우에서와 유사하게 B는 $< ID_I, ID_J, U, (h_2', w'), h_3' >$형태를 얻을 때까지 L_{H_2}-리스트상에 존재하는 정보들을 이용하여 남은 과정을 반복하여 수행한다.

(ii) $ID_A \neq ID_I$, $ID_B \neq ID_J$인 경우에 B는 리스트 L_{H_3}상에서 어떤 α에 대해 $< ID_A, ID_B, \alpha P, h_2 >$형태가 존재하는지를 먼저 체크한다.

- 만일 그런 형태가 리스트에 존재하면, B는 (i)의 경우에서와 같은 과정을 반복 수행한다. 즉, B는 $w'' = \hat{e}(U, \alpha_B d_B)$값을 계산하게 된다. 또한 B는 $ID_B \neq ID_J$이므로 리스트 L_{H_1}으로부터 $\alpha_B d_B = \alpha_B s b_B P$값을 얻게 되어 w''값은 계산 가능하다. 이후 리스트 L_{H_3}상에 $< ID_A, ID_B, U, (h_2, w''), h_3'' >$을 추가하고 응답 값으로 h_3''을 준다.
- 만일 그런 형태가 리스트에 존재하지 않는다면, B는 랜덤하게 $z^* \in Z_q^*$를 선택하고 리스트 L_{H_2}에 $< ID_A, ID_B, z^*P, h_2^* >$를 추가하고 $w^* = \hat{e}(U, \alpha_B^* d_B)$값을 계산한다. (h_2^*, w^*)를 이용해서 H_3-oracle을 시뮬레이트(simulate)한 후, $h_3^* = H_3(h_2^*, w^*)$를 얻는다. 결과 값을 이용하여 리스트 L_{H_3}에 $< ID_A, ID_B, U, (h_2^*, w^*), h_3^* >$를 추가하고 h_3^*로 응답한다.

<u>비밀키 질의(Key extraction query)</u> : A가 ID_B에 대한 key extraction query를 요청할 때,

(i) 만약 ID_A가 ID_I이거나 ID_J인 경우, B는 공격에 실패하고 이 게임을 그만두게 된다.

(ii) 만일 $ID_A \neq ID_I$, ID_J일 경우, 리스트 L_{H_1}은 분명히 $< ID_A, b_A P, b_A \alpha_A >$를 포함하고 있게 된다. 이 결과를 이용하면 ID_A에 상응하는 복호화키(decryption key)는 $\alpha_A s Q_A = \alpha_A s b_A P = \alpha_A b_A s P$이 되고 이 값은 B에 의해 계산되어 A에게 응답된다.

<u>암호화 질의(Encryption query)</u> : A는 평문 M과 ID_A, ID_B에 대한 암호화 질의(Encrypt query)를 요청할 수 있다.

(i) 만일 $ID_A = ID_I$이고 $ID_B = ID_J$ 이라면 B는 랜덤하게 $r \in Z_q^*, \sigma \in \{0,1\}^n, \alpha_B \in Z_q^*$를 선택한 후 $U' = rQ_A$ $V' = \sigma \oplus H_3(H_2(zP), \hat{e}(U', \alpha_B s b_J Q_J))$, $W' = M \oplus H_5(\sigma)$를 계산한다.

(ii) 만일 $ID_A \neq ID_I, ID_B \neq ID_J$인 경우, B는 ID_A에 상응하는 비밀키를 계산하고, 이 값과 앞서 설명된 공개키 알고리즘을 이용해서 암호문을 작성한다.

<u>복호화 질의(Decryption query)</u> : A가 ID_A, ID_B 및 암호문 $C = < U, V, W >$에 대한 복호화 질의(decryption query)를 요청한다고 가정하자.

(i) 만일 $ID_A = ID_I, ID_B = ID_J$인 경우, 먼저 리스트 L_{H_3}상에 $< ID_I, ID_J, U, (h_2, w), h_3 >$형태가 존재하는지를 검색한다. 만일 그런 형태가 리스트상에 존재하면 $p = (h_2, w)$값을 리스트 L_p에 추가하고, A에게는 질의한 암호문 C가 무효한 것이라고 통보한다.

(ii) 만일 $ID_A \neq ID_I, ID_B \neq ID_J$인 경우, 리스트 L_{H_3}는 $< ID_A, ID_B, U, (h_2, w''), h_3'' >$형태를 반드시 포함하고 있으며 이 경우 ID_B에 상응하는 복호화키(decryption key)는 $\alpha_B s b_B P$가 된다. 그러면 암호문은 이 키를 이용해서 앞서 설명된 공개키 알고리즘에 따라 복호화 되어진다. 만약 질의된 암호문 C가 유효한 것이라면, 해당 평문은 A에게 주어지게 된다.(이 경우 A는 이 게임에서 승리하게 된다.)

마지막으로 A는 이 게임을 종료하게 되고 어떤 출력 값도 무시되어진다. 만약 리스트 L_p가 비어있을 경우 B는 이 게임에서 실패 하게 되고, 아닌 경우에는 이 리스트상의 임의의 값을 출력하게 된다.

분석(Analysis). A가 guessed ID중의 하나로 key extraction query를 요청하지 않을 확률은 적어도 $1/\binom{q_{H_1}}{2}$이다.(여기서 말하는 guessed ID라는 것은 질의로 요청된 ID가 ID_I, ID_J중의 하나일 경우를 의미한다.)

만일 A가 유효한 암호문을 질의로 요청했다면 이 경우 $1/\binom{q_{H_1}}{2}$보다 큰 확률로 A는 성공적으로 guessed ID들 사이의 암호문을 성공적으로 위조하게 된다.(그러나 이 경우의 암호문은 무효하다고 통보되게 된다.) 만일 $p=(H_2(xyP), \hat{e}(P,P)^{abc})$가 리스트 L_p안에 없다면 A의 공격 관점은 정확한 위조와는 별개의 것이 된다. 결과적으로 A가 $H_3(p)$형태의 질의를 요청할 확률은 적어도 ϵ이다. 만약 이런 경우가 발생한다면 B는 결코 실패할 수 없게되며 적어도 $\frac{1}{q_D}$만큼의 확률을 가지고 정확한 값을 출력하게 된다. 따라서 B가 기반이 되는 문제를 해결할 전체적인 확률은 적어도 $\epsilon/\binom{q_{H_1}}{2}^2 q_D$이 된다.

4.2 기밀성(confidentiality)에 대한 안전성

본 고에서 제안하는 스킴의 안전성은 BDHP와 CDHP의 어려움에 기반한다. 우리는 정리1과 유사한 방법을 이용하여 기밀성에 대한 안전성을 증명한다.

[정리 2] 해쉬함수 H_1, H_2, H_3, H_4, H_5를 랜덤 오라클로 보자. 우리 스킴이 암호문이 위조 불가능한 무결성을 가진다고 가정하자. 그러면 알고리즘 IG에 의해 생성된 군(group)상에서 BDHP와 CDHP가 어렵다고 가정할 때 우리 스킴은 선택 암호문 공격에 안전한 공개키 암호화 기법(IND-CCA secure public key encrytpion)이 된다. 좀 더 자세히 말하자면, A가 ϵ만큼의 이점을 가지고, 많아야 q_E번의 비밀키 질의(Extraction query), 많아야 q_D번의 복호화 질의(Decryption query), 해쉬함수 H_1, H_2, H_3 각각에 대해 많아야 $q_{H_1}, q_{H_2}, q_{H_3}$번의 질의를 요청하는 다항식으로 제한된 IND-CCA 공격자가 있다고 가정하자. 그러면 $\epsilon/q_{H_2}\binom{q_{H_1}}{2}^2$ 정도의 이점을 가지고 BDHP와 CDHP를 해결하는 다항식적으로 제한된 알고리즘 B가 존재한다.

(증명)

기밀성에 대한 본 증명은 정리1의 증명과 유사한 과정이라 할 수 있다. 정리1에서 우리 스킴이 암호문이 위조 불가능함을 증명했으므로 이 정리의 증명에서는 이를 가정하므로 복호화 오라클(Decryption Oracle)의 질의 요청 처리과정이 변하므로, 단지 복호화 질의(Decryption query)에서 다를 뿐이다. 그 밖의 H_1, H_2, H_3에 대한 Hash query들은 정리1의 증명에서와 같이 알고리즘 B에 의해 처리된다. 공격자 A에 의해 요청되는 Encryption과 Key extraction query를 처리하는 방식은 정리 1의 증명에서와 같이 B에 의해 수행된다. 따라서 여기서는 정리1의 증명과 다른 부분(복호화 질의 : Decryption query)에 대한 설명만을 주기로 한다.

단계 1 : A가 복호화 질의를 요청할 때마다 요청한 암호문은 무효한(invalid) 것이라고 통보되어진다. 암호문이 위조불가능하다는 가정에 따라 A는 복호화 오라클의 시뮬레이션과 현실의 것을 구분할 수 없게 된다.

Challenge : 다항식적으로 제한된 수만큼의 질의를 요청한후에 A는 도전(challenge)하길 원하는 ID들을 선택한다. 선택된 ID들인 ID_A, ID_B와 두개의 같은 길이를 갖는 평문들을 생성해서 challenge할 때, B는 다음과 같이 질의를 처리해서 응답한다.

(i) 만약 질의 요청된 ID들이 guessed ID가 아니라면 B는 실패하게 되고 게임을 그만두게 된다.

(ii) 만약 질의 요청된 ID들이 guessed ID라면 암호문은 앞서 설명된 PKC 알고리즘에 따라 임의의 랜덤한 값 $r \in Z_q^*, \sigma \in \{0,1\}^n$, $M_b \in \{M_0, M_1\}$들을 이용해서 계산된다. B는 이렇게 처리된 암호문 $C=<U, V, W>$를 challenge의 응답 값으로 준다.

단계 2 : 이 단계에서는 정리 1의 단계 1에서와 마찬가지로 B는 키 추출, 암호화, 복호화 질의(*Extraction, Encryption, Decryption query*) 요청들을 처리해서 응답을 준다.(물론 앞서 언급한 대

로 decryption query 요청에 대해서는 예외적인 처리가 요망된다.) 다만 이 단계에서 A의 질의 요청에 따른 몇 가지 제한점을 다음과 같이 두기로 한다.

- 만약 A가 자신이 목적한 ID들을 선택하기 전에 ID_I나 ID_J에 관한 비밀키 질의 (key extraction query)를 요청한다면 B는 요청된 질의를 처리할 수 없으므로, 목적한 바를 이루지 못하고 실패하게 된다.
- 만약 A가 ID_I, ID_J에 관해 challenge를 요청한다면 A는 해당 ID들에 대해서는 키 추출 질의(key extraction 질의)를 요청할 수 없다.
- A는 challenge 상에서 요청한 ID들과 M_b를 암호화 하는데 사용된 관련된 공개키 들로 구성된 challenge의 응답 값인 암호문에 대해 다음 단계에서 복호화 질의(decryption query)를 요청할 수 없다.

추측 : 마지막으로 A는 자신의 b에 대한 추정 값인 b'을 게임의 결과 값으로 출력하게 되고 만약 $b=b'$일 경우 이 게임을 이기게 된다. B의 경우에는 자신이 리스트인 L_p를 살펴보아서 리스트가 어떤 정보도 포함하고 있지 않다면 B는 이 게임에서 목적하는 바를 성취하지 못하고 실패하게 된다. 그러나 만일 리스트 L_p가 정보들을 저장하고 있을 경우에는 그 리스트상의 임의의 원소를 출력하게 된다.

분석(Analysis). A가 위에서 묘사된 시뮬레이션(simulation)동안에 guessed ID에 대해 비밀키 질의(private key extraction query)를 요청한 경우에는 B가 실패함을 알 수 있다. 또한 $\binom{q_{H_1}}{2}$만큼의 ID쌍들이 존재하므로 그것들 중의 적어도 하나는 A로부터 key extraction query의 대상이 될 수 없음도 또한 알 수 있다. 따라서 적어도 $1/\binom{q_{H_1}}{2}$만큼의 확률을 가지고 A는 guessed ID들인 ID_I, ID_J에 대한 키 추출(key extraction) 요청을 하지 않는다. 또한 A의 challenge ID들이 guessed ID쌍인 (ID_I, ID_J)일 확률은 $1/\binom{q_{H_1}}{2}$이다. 만일 A가 $p=p=(H_2(xyP), \hat{e}(P,P)^{abc})$인 p에 대해 $H_3(p)$형태로 질의를 요청하지 않는다면 A의 공격에 대한 관점은 평문 M에 대해 무관하게 된다. 따라서 이런 경우 A는 실제와 시뮬레이션(simulation)을 구분할 수 없게 되므로 이 게임을 통해 어떤 이득도 얻을 수 없게 된다. 결과적으로 A가 $H_3(p)$형태의 질의를 요청할 확률은 적어도 ϵ 정도가 된다. 만일 $H_3(p)$형태의 질의가 A에 의해 요청되었다면 A는 실제와 시뮬레이션을 구분 할 수 있게 되지만 p는 B의 리스트 L_p상에 저장된 후이다. 만일 B가 출력할 정확한 L_p의 원소를 추정하게 된다면 B는 이 게임에서 목적한 바를 성취하게 된다. 이런 리스트 L_p의 크기는 q_{H_2}에 의해 한정되어지므로 결과적으로 전체적인 확률은 $Adv(B) \geq \epsilon / \binom{q_{H_1}}{2}^2 q_{H_2}$이다.

5. 타 스킴과의 효율성 비교

최근 들어 Weil/Tate pairing을 이용한 암호시스템들이 많이 제안되고 있는데 일반적으로 Pairing을 계산하는데 드는 노력은 다른 계산에 비해 매우 큰 것으로 알려져 있다. 이를 고려하여 본 논문에서 제안 하는 스킴은 AP[1]경우와 비교할 때 두 번의 pairing 계산 절감효과를 지니고 있다. 이는 [1]의 암호화 스킴에선 송신자가 암호화하기 전에 수신자의 공캐키 정당성을 확보하기 위해 pairing을 계산하는 단계를 두고 있음에 기인한다. 즉, [1]에서는 수신자의 공개키가 (X_A, Y_A)일 때 공개키 정당성을 확인하기 위해 $\hat{e}(X_A, P_{pub}) = \hat{e}(Y_A, P)$등식 성립 여부를 체크하는데 본 논문에서 제안하는 스킴은 이러한 과정을 요구하지 않는다. 송신자 A가 수신자 B의 잘못된 공개키 (X_A', Y_A')을 이용해서 암호문 $C=(U,V,W)$을 작성해서 B에게 보낸 경우를 가정해 보자. B는 받은 암호문을 가지고서 3장에서 기술된 복호화 과정을 수행해서 (σ', M)을 얻게 되는데 이 값은 A의 원래 선택값과는 다를 수 밖에 없다. 결국, 마지막 검증과정인 $U=rQ_A$를 통과 할 수 없게 된다. 수신자는 이를 통해 통신상의 오류나 공개키 변조여부를 감지 할 수 있으므로 별도의 pairing 체크를 수행하지 않아도 된다. 또한, 공격자가 이미 공개된 정보인 P_{pub}, P를 이용해서 등식을 만족하도록 (X_A, Y_A)를 구성하는 상황은 예측가능하고 이런 체크는 사용자 측면의 계산량 부담을 증가시킨다는 취지에서 제안하는 스킴에선 요구하지 않고 있다.

표 2에서 보는 바와 같이 본 스킴은 AP[1] 경우와 비교하여 볼 때 쌍방향 인증성을 만족한다. 즉, AP[1]상의 송신자는 암호문 작성시 본인이 메시지를 작성했음을 나타내는 어떤 정보도 넣지 않는다. 따라서 이렇게 작성한 암호문은 수신자 측면에선 익명의 편지로 인식된다. 제안하는 스킴은 송/수신자 모두가 쌍방의 정보를 넣어 암/복호화를 진행함으로써 쌍방향 인증성을 제공한다.

다음의 표는 본 논문에서 제안한 스킴(LL 스킴)과 유사한 타 스킴과의 효율성과 안전성 측면을 비교한 것이다.

표 1. 효율성

스킴	pairing의 계산횟수	곱셈의 계산회수	지수계산 횟수
BF [2,3]	2	1	1
L[13]	2	0	0
AP[1]	4	1	1
LL 스킴	2	3	1

표 2. 안전성

스킴	인증 (Authentication)	키위탁 (Escrow)	위조불가능 (Unforgeability)	기밀성 (confidentiality)
BF[2,3]	X	X	X	O
L[13]	O	X	X	O
AP[1]	O(half)*	O	O	O
LL 스킴	O**	O	O	O

* 일방향 인증(unilateral authentication)만을 만족한다.
** 쌍방향 인증(mutual authentication)을 만족한다.

6. 결 론

본 고에서 우리는 인증된 공개키 스킴을 제안하고 BDHP와 CDHP의 가정하에서 기밀성과 암호문 위조불가능성(무결성)을 증명하였다. 제안한 스킴은 Al-Riyami와 Paterson의 논문에서 설명된 스킴과 약간 유사하긴 하나 그들의 스킴은 단지 일방향 인증성만을 만족하지만 본 고의 스킴은 쌍방향을 제공하며 좀 더 효율적이다. 또한 우리의 스킴을 사용하는 사용자들은 각자의 일시적인 참여로부터 발생하는 Diffie-Hellman 형태의 고유 비밀 값을 사용함으로써 제3자(PKG)기관의 부정을 막을 수 있게 된다.

참 고 문 헌

[1] S. S. Al-Riyami, K. G. Paterson, Certificateless Public Key Cryptography. In Proc. Asiacrypt'03, LNCS 2784, Springer Verlag, Lecture Notes in Computer Science series, 2003.

[2] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing, In Proc. Crypto '01, LNCS 2139, pages 213-229, 2001. See [3] for the full version.

[3] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing, SIAM J. of Computing, Vol. 32, No. 3, pp. 586-615, 2003.

[4] J. C. Cha and J. H. Cheon. An identity-based signature from gap Diffie-Hellman group, Public Key Cryptography 2003: 18-30.

[5] L. Chen, K. Harrison, A. Moss, D. Soldera, and N. P. Smart. Certification of public keys within an identity based system. In A. H. Chan and V. D. Gligor, editors, Information Security, 5th International Conference, ISC, volume 2433 of LNCS, pages 322-333. Springer-Verlag, 2002.

[6] L. Chen and C. Kudla. Identity based authenticated key agreement from pairing. CSFW 2003: 219-233.

[7] R. Dupont, A. Enge, Provably secure non-interactive key distribution based on pairings, to appear in Discrete Applied Mathematics. Preliminary version in Proceedings of the International Workshop on Coding and Cryptography, Versailles-WCC 2003.

[8] E. Fujisaki, T. Okamoto, Secure integration of asymmetric and symmetric encryption schemes, Advances in Cryptology-Crypto'99, LNCS 1666, Springer, pp.537-554, 1999.

[9] C. Gentry, Certificate-Based Encryption and the Certificate Revocation Problem, In E. Biham, editor, Advances in Cryptology-EUROCRYPT 2003, volume 2656 of LNCS, pages 272-193, Springer-Verlag, 2003.
[10] C. Gentry, A. Silverberg, Hierarchical ID-based cryptography, Advances in Cryptology-Asiacrypt'02, Lecture Notes in Computer Science 2501, Springer-Verlag, pp.548-566, 2002.
[11] F. Hess, Efficient identity based signature schemes based on pairings, to appear in proceedings of SAC '2002. Springer Verlag, Lecture Notes in Computer Science series.
[12] B. Libert and J. J. Quisquater. New identity based signcryption schemes based on pairings, IEEE Information Theory Workshop 2003, Paris, France, or full version in Cryptology ePrint Archive, Report 2003/023, 2003, http://eprint.iacr.org/.
[13] B. Lynn, Authenticated identity-based encryption. Cryptology ePrint Archive, Report 2002/072, 2002, http://eprint.iacr.org/.
[14] K. G. Paterson, ID-based signatures from pairings on elliptic curves, Electronics Letters, Vol. 38 (18) (2002), 1025-1026.
[15] A. Shamir. Identity-based cryptosystems and signature schemes. In Proc. Crypto '84, LNCS 196, pages 47-53, 1984.
[16] N. P. Smart. An identity-based authenticated key agreement protocol based on the Weil pairing. Electronics Letters, Vol 38, pp 630-632, (2002).
[17] 김태구, 염대현, 이필중, 보다 효율적인 Hierarchical ID-based cryptosystem, 정보보호 학회 논문지 제13권 제3호, 2003.
[18] 김현주, 오수현, 원동호, 효율적인 ID 기반 부분 은닉 서명에 관한 연구, 정보보호 학회 논문지 제13권 제6호, 2003.
[19] 이정연, 천정희, 김태성, 진승헌, Bilinear 함수를 이용한 ID 기반 대리서명 기법, 정보보호 학회 논문지 제13권 제2호, 2003.

〈著者紹介〉

이 영 란(Young-Ran Lee) 학생회원
1996년 : 국민대학교 수학교육과 졸업
1998년 : 이화여대 수학과 석사
1999년 3월~현재 : 이화여대 수학과 박사과정
〈관심분야〉 암호프로토콜

이 향 숙 (Hyang-Sook Lee) 학생회원
1986년 : 이화여대 수학과 졸업
1988년 : 이화여대 수학과 석사 졸업
1993년 : 노스웨스턴 대학교 수학과 박사 졸업
〈관심분야〉 암호학