JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지(J. Korea Inst. Inf. Commun. Eng.) Vol. 20, No. 6 : 1109~1122 Jun. 2016

무인증서 공개키 암호 기법의 재고: 안전성 모델 및 설계

김송이 · 박승환 · 이광수*

Certificateless Public Key Encryption Revisited: Security Model and Construction

Songyi Kim · Seunghwan Park · Kwangsu Lee*

Graduate School of Information Security, Korea University, Seoul 02841, Korea

요 약

무인증서 공개키 암호(Certificateless Public Key Encryption scheme)는 사용자 ID를 공개키로 사용함으로써 공개키 암호 시스템의 인증서 관리 문제를 해결하고 ID기반 암호 기법의 키 위탁(key escrow) 문제를 해결할 수 있는 기술이다. 이에 대한 연구가 활발히 진행되었음에도 불구하고, 기존의 여러 무인증서 암호 기법들은 사용자가 선택한 비밀값과 복호화 키 노출 공격에 대한 안전성을 고려하지 않고 설계되었다. 비밀값과 복호화 키 노출 공격이란 한 번이라도 공개키가 교체된 이후 이전에 사용했던 비밀값과 복호화 키가 노출된다면 그로부터 ID에 대응하는 부분 개인키를 획득해 현재의 정당한 복호화 키를 연산할 수 있는 공격이다. 본 논문에서는 키 노출 공격에 대해 안전한 새로운 안전성 모델을 제안하고, 해당 안전성 모델에서 기존의 무인증서 공개키 암호 기법들이 안전하지 않음을 보인다. 또한, 제안한 모델에서 안전한 새로운 무인증서 공개키 암호 기법을 제시하고, DBDH(Decision Bilinear Diffie-Hellman) 가정을 기반으로 안전성을 증명한다.

ABSTRACT

Certificateless public key cryptography is a technique that can solve the certificate management problem of a public key cryptosystem and clear the key escrow issue of ID-based cryptography using the public key in user ID. Although the studies were actively in progress, many existing schemes have been designed without taking into account the safety of the secret value with the decryption key exposure attacks. If previous secret values and decryption keys are exposed after replacing public key, a valid private key can be calculated by obtaining the partial private key corresponding to user's ID. In this paper, we propose a new security model which ensures the security against the key exposure attacks and show that several certificateless public key encryption schemes are insecure in the proposed security model. In addition, we design a certificateless public key encryption scheme to be secure in the proposed security model and prove it based on the DBDH(Decisional Bilinear Diffie-Hellman) assumption.

키워드 : 무인증서, 암호, 키 노출 공격, DBDH 가정

Key word : Certificateless, Cryptography, Key Exposure Attack, Decision Bilinear Diffie-Hellman assumption, etc

Received 27 April 2016, **Revised** 28 April 2016, **Accepted** 09 May 2016

* **Corresponding Author** Kwangsu Lee(E-mail:guspin@korea.ac.kr, Tel:+82-2-3290-4887)
Graduate School of Information Security, Korea University, Seoul 02841, Korea

Open Access http://dx.doi.org/10.6109/jkiice.2016.20.6.1109 print ISSN: 2234-4772 online ISSN: 2288-4165

Ⅰ. 서 론

1.1. 개요

전통적인 공개키 암호 시스템(public key encryption system)에서 송신자가 수신자에게 암호문을 전송했을 때, 송신자가 생성한 암호문은 오직 수신자의 개인키로만 복호화할 수 있다. 이때, 송신자가 메시지를 안전하게 전송하기 위해 수신자의 공개키에 대한 신뢰가 전제되어야 한다. 공개키는 난수성을 가져서 공개키만으로는 소유자가 누구인지 알 수 없다. 이를 이용하여 공격자가 자신의 공개키를 정당한 수신자의 공개키로 위장하더라도 송신자는 정당한 공개키인지 구분할 수 없기 때문에, 공개키 암호 시스템을 안전하게 운용하기 위해서는 공개키와 소유자 사이를 인증해주는 인증서가 필요하다. 인증서를 통해 공개키의 소유 여부를 인증할 수 있으며 이는 제3의 신뢰기관(trusted third authority)을 통해 발급된다. 하지만 공개키 암호 시스템에서 인증서를 이용할 때 인증서 저장, 폐기 및 분배 등 관리 차원의 문제가 발생한다. 따라서 이러한 공개키 암호 기법의 인증서 문제를 해결하기 위해 ID기반 암호 기법이 제안되었다.

Shamir [1]는 기존의 공개키 암호 시스템의 인증서 관리 문제에 대한 대안으로 ID기반 암호 기법(identity based encryption scheme)을 처음으로 제시하였다. ID 기반 암호 기법은 이메일 주소 혹은 IP 주소 등의 문자열이 사용자의 ID로 공개키가 되어 그 자체로 사용자의 공개키 소유 권한이 입증되기 때문에 공개키 기반 구조의 인증서 관리 문제가 발생하지 않는다. 이때, 모든 사용자의 ID에 대응하는 개인키는 키 생성 기관(key generation center, KGC)에 의해 생성되어 각 사용자에게 발급된다. 키 생성 기관이 모든 개인키를 생성해 발급해주기 때문에 모든 개인키를 알게 되는 키 위탁(key escrow)문제가 발생한다. 이러한 키 위탁 문제를 해결하기 위해 무인증서 공개키 암호 시스템(certificateless public key encryption system)이 등장하였다.

무인증서 공개키 암호 시스템은 난수성을 갖는 공개키와 사용자 ID를 함께 공개키로 사용하는 방식으로 ID기반 암호의 키 위탁 문제를 해결하였다. 무인증서 공개키 암호 시스템이 제안된 이후, 기법의 효율성을 위한 연구와 안전성 모델과 증명에 대한 연구 등이 활발하게 이루어졌다[2-5, 7, 8, 11]. 하지만, 기존에 연구되어온 무인증서 공개키 암호 기법들은 여전히 공개키 교체 이후, 이전의 비밀값과 복호화 키가 노출되었을 때의 안전성이 고려되지 않았다. 이러한 키 노출 공격에 대해서도 안전한 무인증서 공개키 암호 기법이 요구된다.

1.2. 관련연구

2003년 Al-Riyami와 Paterson이 새로운 공개키 암호 패러다임으로 무인증서 공개키 암호 기법을 제안하였다[2]. 무인증서 공개키 암호 기법의 공격자 모델을 정의하고 제안한 모델에서 랜덤 오라클 모델(random oracle model)을 이용하여 안전성을 증명하였다. 이후 2005년에 Al-Riyami와 Paterson에 의해 효율성을 높인 무인증서 암호 기법을 제시되었다[3]. 하지만 이후 Libert, Quisquater 그리고 Zhang와 Feng은 각각 Al-Riyami와 Paterson이 2005년에 제안한 기법이 안전하지 않음을 보이고 수정된 무인증서 암호 기법을 제안하였다[4, 5]. Libert와 Quisquater는 기존의 선택 평문 공격에 안전한 기법을 Fujisaki-Okamoto 변형 [6]을 이용해 선택 암호문 공격에 안전한 기법의 일반적인 설계 방법을 제시하고, 효율적이며 랜덤 오라클 모델에서 안전성을 증명한 무인증서 암호 기법을 설계하였다[4]. 이 기법은 Shi와 Li [7]가 제시한 기법과 유사하지만 조금 더 효율적이다. Zang와 Feng이 수정한 기법은 비록 엄격한 안전성 증명을 하지 않았지만 별도의 공격이 알려지지 않고 있다[5]. 위의 기법들은 랜덤 오라클 상에서 증명되거나 가정이 다소 엄격한 모델에서 증명되었기 때문에 표준 모델(standard model)에서 강한 안전성을 가지는 무인증서 암호 기법을 설계하는 것이 문제였다. 이 문제는 Dent, Libert 와 Paterson에 의해 표준 모델에서 강한 안전성을 가지는 기법을 설계 및 증명하면서 풀렸다[8]. 2013년 Xiong 등은 복호화 키의 일부 노출에 대하여 안전한 노출-저항(leakage-resilient) 무인증서 암호 기법을 제안하였다[9]. Xiong 등의 기법은 부채널 공격 등에 의한 복호화 키 일부 노출에 대해 안전성을 고려한 기법으로써 전체 복호화 키 노출을 고려하는 본 논문에서의 정의와는 차이가 있다. 최근, Sun 등이 복호화 키 노출에 안전하며, 키 폐기 기능을 제공하는 무인증서 암호 기법을 제안하였다[10]. 이 기법은 노출된 복호화 키와 함께 업데이트 키를 조작하는 공격에 대해서는 안전하다. 하지만, 본 논문에서 고려하는 비

밀값 노출에 대해서는 안전하지 않다. 비밀값을 결합하여 복호화 키를 생성하는 과정이 Dent 등이 제안한 무인증서 암호 기법과 동일하게 복호화 키 구조상에서 비밀값이 재랜덤화되는 것은 아니기 때문이다.

또한 무인증서 암호 기법의 효율성을 높이는 연구가 함께 진행되었는데 Baek, Safavi-Naini와 Susilo이 기법의 효율성을 위해 처음으로 페어링(pairing) 연산을 이용하지 않는 무인증서 공개키 암호 기법을 제안하였다[11]. 이 기법은 연산 비용이 높은 페어링 연산을 제거함으로써 이전의 무인증서 공개키 암호 기법들과 비교해 효율성을 더욱 높였다. 이후 Lai와 Kou가 변형된 형태의 무인증서 암호 시스템인 자가 생성 인증서 기반 공개키 암호 기법을 페어링 연산 없이 가능하도록 설계하였다[12]. 이러한 강한 안전성과 높은 효율성을 갖는 무인증서 공개키 암호 기법에 대한 연구 등이 많이 진행되었음에도 불구하고, 기존의 여러 무인증서 공개키 암호 기법은 질의에 제약을 두었지만 비밀값과 복호화 키 노출 공격을 고려하여 설계되지 않아 무인증서 암호 시스템의 안전성이 깨지는 문제가 발생하였다.

1.3. 기여도

무인증서 공개키 암호에서는 공개키에 대한 인증서가 없기 때문에 누구나 공개키 교체가 가능하다. 악의적인 공격자가 의도적으로 공개키를 교체하려는 상황 이외에도 현실 상황에서 사용자의 비밀값 또는 복호화 키가 의도하지 않게 외부로 노출되었거나 장기간의 키 사용으로 인해 사용자 스스로 공개키 교체를 원하는 상황이 발생할 수 있다. 이와 같은 상황들을 모두 고려하여 만약, 공개키가 교체된 이후 이전의 비밀값과 복호화 키로부터 현재의 암호문을 복호화할 수 있는 정보가 노출된다면 암호 시스템에 대한 안전성을 보장할 수 없다. 하지만, 이전의 비밀값과 복호화 키가 노출되었더라도 부분 개인키가 안전하게 보관되어 있고 현재 정당한 복호화 키를 연산할 수 없다면 여전히 암호 시스템의 안전성은 유지되어야 한다.

본 논문에서는 악의적인 공격자 또는 사용자가 공개키를 교체하려는 상황을 모두 고려하여 키 노출 공격을 고려한 새로운 안전성 모델을 정의하고, 제안하는 안전성 모델에서 기존의 여러 기법들이 안전하지 않음을 보인다. 또한 본 모델에서 안전한 새로운 무인증서 공개키 암호 기법을 설계하고 안전성을 증명한다.

1.4. 설계원리

기존의 무인증서 암호 기법에서 비밀값은 Z_p의 원소로 이루어져 있고 부분 개인키와 단순한 결합으로 복호화 키를 생성한다. 이에 따라, 이전에 사용한 비밀값과 복호화 키의 노출은 자연스럽게 부분 개인키의 노출까지 이어져 현재의 암호문을 복호화 할 수 있는 복호화 키 생성까지 가능했다.

본 논문에서 제안한 기법은 공개키에 대응하는 Z_p의 원소에 새로운 난수를 추가적으로 사용하여 난수성질을 갖는 그룹 G_1의 원소로써 비밀값을 생성한다. 비밀값과 부분 개인키를 이용해 복호화 키를 생성할 때에도 재랜덤화(re-randomization) 되는 과정을 통해 복호화 키가 난수성질을 갖게 되어 공격자에게 교체 이전의 복호화 키와 비밀값이 완전히 노출되더라도 부분 개인키를 연산해낼 수 없도록 설계하였다. 또한, 공격자가 노출된 교체 이전의 복호화 키와 비밀값으로 정당한 복호화 키를 연산할 수 없도록 시간정보를 이용한다. 이러한 방식으로 다음 그림 1과 같은 키 노출 공격에 안전한 무인증서 공개키 암호 기법을 설계할 수 있다.

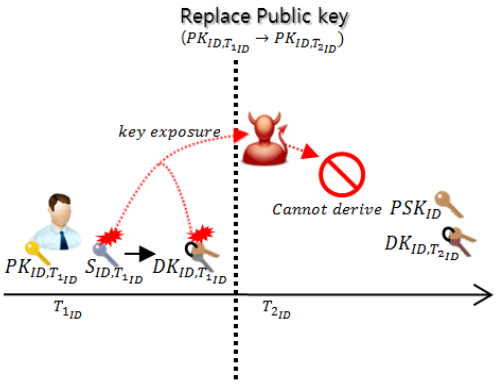

Fig. 1 Secure CL-PKE against key exposure attack

이후 논문의 구성은 다음과 같다. Ⅱ장에서는 기법의 이해에 필요한 배경지식을 설명한다. Ⅲ장에서는 기존의 무인증서 공개키 암호 기법이 키 노출 공격에 안전하지 않음을 보인다. Ⅳ장에서는 키 노출 공격에 안전한 무인증서 공개키 암호 기법을 제안하고, 제안한 기법의 안전성을 증명한다. Ⅴ장에서는 무인증서 공개키 암호와 관련한 추가적인 연구에 대해 논의하고, 마지막으로 Ⅵ장에서 결론을 맺는다.

Ⅱ. 배경지식

2.1. 겹선형 함수

위수를 소수 p로 갖는 곱셈 군 G_1와 G_2의 생성원은 해당하는 군의 부분집합으로 원소와 역원을 곱하여 군의 모든 원소를 표현할 수 있는 집합이다. 이때, G_1의 생성원(generator)이 g라고 가정하자. 군의 생성원은 해당하는 군의 부분집합으로 원소와 역원을 곱하여 해당하는 군의 모든 원소를 표현할 수 있는 집합이며, Z_p^*상의 원소는 $\{1,2,..,p-1\}$에 해당하는 정수 값을 의미한다. 함수 $e : G_1 \times G_1 \rightarrow G_2$는 다음 조건을 만족한다.

1) 겹선형성(bilinearity): 임의의 두 원소 $g, h \in G_1$와 $a, b \in Z_p^*$에 대해 $e(g^a, h^b) = e(g,h)^{ab}$가 된다.
2) 비소실성(non-degeneracy): $e(g,g) \neq 1$을 만족하는 $g \in G_1$가 존재한다.
3) 계산 가능성(computability): 임의의 $g, h \in G_1$에 대해서 $e(g,h)$를 효율적으로 계산할 수 있는 알고리즘이 존재한다.

2.2. DBDH(Decisional Bilinear Diffie-Hellman) 가정[13]

위수를 소수 p로 갖는 군 G_1가 있고 겹선형 함수 $e : G_1 \times G_1 \rightarrow G_2$가 있을 때 $g \in G_1$ 을 G_1의 생성원이라고 가정하자. DBDH 문제는 지수 부분의 a,b,c가 Z_p^*상의 원소이고 인자 (g, g^a, g^b, g^c, T)가 주어졌을 때, $e(g,g)^{abc} = ?\ T$ 인지를 결정하는 것이다. 이때, 군의 생성원은 해당하는 군의 부분집합으로 원소와 역원을 곱하여 해당하는 군의 모든 원소를 표현할 수 있는 집합이며, Z_p^*상의 원소는 $\{1,2,..,p-1\}$에 해당하는 정수값을 의미한다. 알고리즘 B는 $e(g,g)^{abc} = T$ 혹은 $e(g,g)^{abc} \neq T$을 결정함에 따라 비트 $b \in \{0,1\}$을 출력하는 알고리즘이다. 이때, 다항식 시간 안에 의미 있는(non-negligible) 확률로 계산할 수 있는 알고리즘 B가 존재하지 않는다면 DBDH 문제는 풀기 어렵다고 정의하며, 알고리즘 B가 DBDH 문제를 풀 수 있을 이점(advantage) ϵ을 다음과 같이 정의한다.

$$Adv(B) = \left| \begin{matrix} \Pr[B(g,g^a,g^b,g^c,e(g,g)^{abc}) = 0] \\ -\Pr[B(g,g^a,g^b,g^c,T) = 0] \end{matrix} \right| \geq \epsilon \quad (1)$$

2.3. 무인증서 공개키 암호시스템

2.3.1. 알고리즘

무인증서 공개키 암호시스템은 다음과 같이 7가지 알고리즘으로 구성되며, Al-Riyami와 Paterson[9] 정의한 알고리즘과 동일하다. 이때, 키 생성 기관에 의해 부분 개인키가 발급되고, 사용자 비밀값과 공개키 생성 시에 시간정보 T_{ID}를 이용한다.

- $Setup(k)$: 설정 알고리즘은 키 생성 기관에 의해 수행되며 보안 상수 k를 입력받아 마스터키 MK와 공개 파라미터 PP를 출력한 후, 키 생성 기관은 마스터키를 저장하고, 공개 파라미터는 누구에게나 공개된다.
- $Extract-Partial-Private-Key(PP, MK, ID)$: 부분 개인키 추출 알고리즘은 사용자마다 한 번씩 키 생성 기관에 의해 수행된다. 키 생성 기관은 공개 파라미터, 마스터키와 사용자 아이디를 입력 받아 부분 개인키 PSK_{ID}를 출력하여 사용자에게 전송한다.
- $Set-Secret-Value(PP, ID)$: 비밀값 설정 알고리즘은 사용자에 의해 수행되며 공개 파라미터와 사용자 아이디를 입력 받아 비밀값 S_{ID}을 출력한다.
- $Set-Public-Key(PP, ID)$: 공개키 설정 알고리즘은 사용자에 의해 수행되며 공개 파라미터, 사용자 아이디를 입력받아 공개키 PK_{ID}을 출력한다.
- $Set-Decryption-Key(PP, ID, PSK_{ID}, S_{ID})$: 복호화 키 설정 알고리즘은 사용자에 의해 수행되며 공개 파라미터, 부분 개인키와 비밀값을 입력 받아 복호화 키 DK_{ID}를 출력한다.
- $Encrypt(PP, ID, PK_{ID}, M)$: 암호화 알고리즘은 공개 파라미터, 사용자의 아이디와 공개키, 평문 메시지 M을 입력받아 암호문 C을 출력한다.
- $Decrypt(PP, DK_{ID}, C)$: 복호화 알고리즘은 공개 파라미터, 복호화 키와 암호문을 입력 받아 평문 메시지를 반환한다.

2.3.2. 안전성 모델

무인증서 암호 기법의 안전성 모델에서 공격자 유형은 공격자 유형Ⅰ과 공격자 유형Ⅱ으로 나뉜다. 공격자 유형Ⅰ은 키 생성 기관의 마스터키 정보는 알 수 없지만 사용자의 공개키를 교체할 수 있고, 공격자 유형Ⅱ는 사용자들의 공개키를 교체할 수 없지만 키 생성 기관의 마스터키 정보를 획득할 수 있다. 두 공격자

유형이 어떠한 다항식 시간(polynomial-time) 안에 챌린저 B와의 게임에서 이길 확률이 무시할 수 있는(negligible) 값이고 선택 평문 공격(chosen plaintext attack)을 고려한 모델일 때, 무인증서 암호 시스템이 IND-CPA을 만족한다고 정의한다. 또한, 본 안전성 모델은 선택적인 ID 모델로 챌린지 아이디 ID^*를 설정 단계 이전에 공격자가 미리 정한다.

제안하는 안전성 모델은 기존의 무인증서 공개키 암호 시스템의 안전성 모델에서 키 질의에 대한 조건을 더욱 완화한 모델이다. 기존의 안전성 모델에서는 공개키가 교체된 이후나 챌린지 아이디에 대해서는 복호화 키 추출 질의를 할 수 없었다. 하지만, 제안하는 안전성 모델은 공개키가 교체된 이후에도 챌린지 아이디에 대한 부분 개인키와 공개키에 대한 비밀값을 얻을 수 없다면, 챌린지 아이디에 대해서도 복호화 키 추출 질의가 가능하다. 즉, 제안하는 안전성 모델의 공격자 유형Ⅰ과 공격자 유형Ⅱ은 기존과 다르게 공개키가 교체된 이후에도 챌린지 아이디에 대해 복호화 키 추출 질의가 가능한 것이다.

(1) 선택적인 ID 모델- 공격자 유형Ⅰ

공격자 유형Ⅰ A가 선택적인 IND-sID-CPA 안전성 모델은 다음과 같다.

- $Init$: A는 공격할 ID^*을 챌린저 B에게 전송한다.
- $Setup$: 챌린저 B는 보안 상수 1^k을 가지고 $Setup$ 알고리즘을 수행한다. 공개 파라미터를 공격자 A에게 전송하고 챌린저 B가 마스터키를 보관한다.
- $Phase\,1$: 공격자 A는 부분 개인키 추출 질의 $Extract-Partial-Private-Key$, 비밀값 추출 질의 $Extract-Secret-Value$, 복호화 키 추출 질의 $Extract-Decryption-Key$, 공개키 요청 질의 $Request-Public-Key$ 혹은 공개키 교체 질의 $Replace-Public-Key$가 가능하다. 챌린저 B는 공격자의 질의에 다음과 같이 답한다.

  (단, ID^*에 대한 부분 개인키 PSK_{ID^*}와 공개키 PK_{ID^*}에 대한 비밀값 S_{ID^*}는 동시에 획득할 수 없다.)

  • $Extract-Partial-Private-Key(ID)$: 공격자 A가 아이디에 해당하는 부분 개인키를 질의하면, 챌린저 B는 부분 개인키 추출 알고리즘을 수행한 후, 공격자가 요청한 사용자 아이디에 대한 부분 개인키 PSK_{ID}를 공격자 A에게 전송한다.

  (ID^*에 대한 부분 개인키 추출 질의를 할 수 없다.)

  • $Extract-Secret-Value(ID)$: 공격자 A가 비밀값을 질의하면 챌린저 B는 비밀값 설정 알고리즘을 수행한 후, 사용자 아이디에 해당하는 비밀값 S_{ID}을 출력 받고 이를 공격자 A에게 전송한다.

  • $Extract-Decryption-Key(ID)$: 공격자 A가 복호화 키를 질의하면 개인키 설정 알고리즘을 수행한 후, 사용자 아이디에 해당하는 복호화 키 DK_{ID}을 출력 받아 이를 공격자 A에게 전송한다.

  • $Request-Public-Key(ID)$: 공격자 A가 사용자 아이디에 해당하는 공개키를 요청하면 챌린저 B는 공개키 설정 알고리즘을 수행한 후, 해당하는 공개키 PK_{ID}를 출력 받아 이를 공격자 A에게 전송한다.

  • $Replace-Public-Key(ID,PK_{ID}')$: 공격자 A가 공개키 교체를 요청하면 챌린저 B는 사용자 아이디에 해당하는 공개키를 선택한 공개키로 교체한다.
- $Challenge$: 공격자 A는 길이가 같은 두 메시지 M_0, M_1을 선택하여 챌린저 B에게 전송한다. 챌린저는 $b \in \{0,1\}$를 임의로 뽑고, 선택한 ID^*에 해당하는 PK_{ID^*}을 이용해 M_b에 관한 암호문 C^*을 계산한다. 챌린저는 공격자에게 챌린지 암호문 C^*을 전송한다.
- $Phase\,2$: $Phase\,2$는 추가적으로 $Phase\,1$과 유사한 질의가 가능하다.
- $Guess$: A는 b에 대한 응답으로 $b' \in \{0,1\}$을 B에게 전송한다. 만약 $b = b'$이면 공격자가 게임에서 이긴다.

(2) 선택적인 ID 모델- 공격자 유형Ⅱ

공격자 유형Ⅱ A가 선택적인 IND-sID-CPA 안전성 모델은 다음과 같다.

- $Init$: A는 공격할 ID^*을 챌린저 B에게 전송한다.
- $Setup$: 챌린저 B는 보안 상수 1^k을 가지고 $Setup$ 알고리즘을 수행한다. 공개 파라미터와 마스터키를 공격자 A에게 전송한다.
- $Phase\,1$: 공격자 A는 비밀값 추출 질의 $Extract-Secret-Value$복호화 키 추출 질의 $Extract-Decryption-Key$ 혹은 공개키 요청 질의 $Request-Public-Key$가 가능하다. 챌린저 B는 공격자의 질의에 다음과 같이 답한다.

  (단, ID^*에 대한 부분 개인키 PSK_{ID^*}와 공개키 PK_{ID^*}에 대한 비밀값 S_{ID^*}는 동시에 획득할 수 없다.)

• $Extract-Secret-Value(ID)$: 공격자 A가 비밀값을 질의하면 챌린저 B는 비밀값 설정 알고리즘을 수행한 후, 사용자 아이디에 해당하는 비밀값 S_{ID}을 출력 받고 이를 공격자 A에게 전송한다.

• $Extract-Decryption-Key(ID)$: 공격자 A가 복호화 키를 질의하면 복호화 키 설정 알고리즘을 수행한 후, 사용자 아이디에 해당하는 복호화 키 DK_{ID}을 출력 받아 이를 공격자 A에게 전송한다.

• $Request-Public-Key(ID)$: 공격자 A가 사용자 아이디에 해당하는 공개키를 요청하면 챌린저 B는 공개키 설정 알고리즘을 수행한 후, 해당하는 공개키 PK_{ID}를 출력 받아 이를 공격자 A에게 전송한다.

- $Challenge$: 공격자 A는 길이가 같은 두 메시지 M_0, M_1을 선택하여 챌린저 B에게 전송한다. 챌린저는 $b \in \{0,1\}$를 임의로 뽑고, 선택한 ID^*에 해당하는 PK_{ID^*} 을 이용해 M_b에 관한 암호문 C^*을 계산한다. 챌린저는 공격자에게 챌린지 암호문 C^*을 전송한다.
- $Phase\,2$: $Phase\,2$는 추가적으로 $Phase\,1$과 유사한 질의가 가능하다.
- $Guess$: A는 b에 대한 응답으로 $b' \in \{0,1\}$을 B에게 전송한다. 만약 $b = b'$이면 공격자가 게임에서 이긴다.

Ⅲ. 기존 기법 안전성 분석

3.1. Baek, Safavi-Naini 와 Susilo의 기법 [11]과 Lai와 Kou의 기법 [12]

기존의 두 무인증서 암호 기법이 다음과 같은 안전성 모델에서 비밀값과 복호화 키 노출 공격에 대해 IND-sID-CPA 안전성을 제공하지 못함을 보인다.

정리. 만약 공격자가 공개키 교체 후 복호화 키 추출 질의하여 교체 이전의 복호화 키를 획득할 수 있다면 메시지 M_0 또는 M_1 중, 어떠한 메시지로부터 암호화된 챌린지 암호문인지를 유의미한 확률로 구별할 수 있다.

증명. 챌린저 B가 공격자 A와 상호작용하며 진행하는 게임과정은 다음과 같다.

- $Init$: A가 공격할 ID^*를 선택하여 B에게 전송한다.
- $Setup$: 보안 상수 k를 입력 받아 공개 파라미터를 생성하고 공격자 A에게 전송한다.
- $Phase1$: 공격자 A는 다음과 같이 수행한다.

1. 공격자 A는 복호화 키 추출 질의를 통해 다음과 같은 복호화 키를 얻을 수 있다. (이때, $z \neq z^*$이다.)

$$DK_{ID^*} = < S_{ID^*} = z, \quad PSK_{ID^*} = s + xH_1(ID^*, w) > \tag{1}$$

2. A는 교체할 공개키를 획득하기 위해 공개키 설정 알고리즘을 통해 ID^*에 해당하는 공개키 PK'_{ID^*}를 획득할 수 있다. (현재 시점에 공개키는 PK_{ID^*}이다.)

3. A는 공개키 교체 질의를 통해 ID^*에 대한 현재 공개키를 공개키 PK'_{ID^*}로 교체한다.

4. A는 비밀값 설정 알고리즘을 수행하여 비밀값 $S'_{ID^*} = z^*$을 획득한다.

- $Challenge$: A는 ID^*와 길이가 같은 두 메시지 M_0, M_1을 선택하여 B에게 전송하면, B는 $b \in \{0,1\}$를 임의로 뽑아 ID^*에 해당하는 PK'_{ID^*}을 이용해 M_b에 관한 챌린지 암호문 C^*을 계산하여 A에게 전송한다. C^*의 형태는 다음과 같다.

$$C^* = < C_0 = g^r, \quad C_1 = H_3(\mu^r, (wy^{H_1(ID^*, w)})^r) \oplus (M \| \sigma) > \tag{2}$$

where $\mu = z^*$

- $Guess$: 공격자 A는 다음과 같이 수행한다.

• A는 교체 이전의 복호화 키를 획득했다면, 복호화 키 구성요소에 부분 개인키와 비밀값의 형태가 그대로 유지된다. 이때, 부분 개인키 PSK_{ID^*}를 추출하고 ID^*에 해당하는 현재 복호화 키 DK'_{ID^*}를 획득하는 과정은 다음과 같다.

1. $Phase1$ 단계에서 획득한 공개키 교체 이전의 복호화 키는 1.-(1)과 같다.

$$DK_{ID^*} = < S_{ID^*} = z, \quad PSK_{ID^*} = s + xH_1(ID^*, w) > \tag{3}$$

2. 형태가 그대로 노출된 부분 개인키 PSK_{ID^*}와 $Phase1$ 단계 4.에서 획득한 비밀값 S'_{ID^*}으로 정당한 복호화 키를 계산한다.

$$PSK'_{ID^*} = PSK_{ID^*} \text{ 값이 같으므로,} \quad DK'_{ID^*} = < S'_{ID^*} = z^*, \quad PSK'_{ID^*} = s + xH_1(ID^*, w) > \tag{4}$$

• 복호화 키 DK'_{ID^*}를 이용하여 챌린지 암호문 C^*을 복호화 한다. 결과적으로, A가 획득한 메시지 M_b의 b 값을 찾아 이를 출력한다.

증명을 통해 A의 행위는 정당하며 무시할 수 없는 확률로 챌린지 메시지 M_b를 구별할 수 있음을 보였다. 이로써 A가 항상 게임에서 승리한다는 사실은 명백하다.

Lai와 Kou의 기법[12]의 키 구조는 위의 Baek 등의 키 구조와 유사하기 때문에 위의 방법과 같이 공격이 가능하다. 자세한 증명 과정은 생략한다.

3.2. Dent, Libert와 Paterson의 기법 [8]

기존의 무인증서 암호 기법이 다음과 같은 안전성 모델에서 비밀값과 복호화 키 노출 공격에 대해 IND-sID-CPA 안전성을 제공하지 못함을 보인다.

정리. 만약 공격자가 공개키 교체 후 복호화 키 추출 질의와 비밀값 추출 질의를 통해 교체하기 이전의 복호화 키와 비밀값을 획득할 수 있다면 메시지 M_0 또는 M_1 중, 어떠한 메시지로부터 암호화된 챌린지 암호문인지를 유의미한 확률로 구별할 수 있다.

증명. 챌린저 B가 공격자 A와 상호작용하며 진행하는 게임과정은 다음과 같다.

- $Init$: A가 공격할 ID^*를 선택하여 B에게 전송한다.
- $Setup$: 보안 상수 1^k를 입력 받아 공개 파라미터를 생성하고 공격자 A에게 전송한다.
- $Phase1$: 공격자 A는 다음과 같이 수행한다.

1. 공격자 A는 복호화 키 추출 질의를 통해 다음과 같은 복호화 키를 획득한다.

$$DK_{ID^*} =< D_0 = g_2^{\alpha S_{ID^*}} F_u(ID^*)^t, D_1 = g^t > \quad (5)$$

2. A는 비밀값 추출 질의를 통해 ID^*에 대한 비밀값 $S_{ID^*} = s$를 획득한다.

3. A는 교체할 공개키를 획득하기 위해 공개키 설정 알고리즘을 통해 ID^*에 해당하는 공개키 PK'_{ID^*}를 획득할 수 있다. (현재 시점에 공개키는 PK_{ID^*}이다.)

4. A는 공개키 교체 질의를 통해 ID^*에 대한 현재 공개키를 공개키 PK'_{ID^*}로 교체한다.

5. A는 비밀값 설정 알고리즘을 통해 비밀값 $S'_{ID^*} = s^*$을 획득한다.

- $Challenge$: 공격자 A는 챌린지 아이디 ID^*와 길이가 같은 두 메시지 M_0, M_1을 선택하여 B에게 전송하면 B는 $b \in \{0,1\}$를 임의로 뽑아 ID^*에 해당하는 PK'_{ID^*}을 이용해 M_b에 관한 챌린지 암호문 C^*을 계산하여 A에게 전송한다. C^*의 형태는 다음과 같다.

$$C^* =< C_0 = Me(g^{\alpha S'_{ID^*}}, g_2)^s, \\ C_1 = g^s, C_2 = F_u(ID^*)^s, C_3 = F_v(w)^s > \quad (6)$$

- $Guess$: 공격자 A는 다음과 같이 수행한다.

• A는 공개키 교체한 후 교체하기 이전의 복호화 키 DK_{ID^*}와 비밀값 S_{ID^*}을 획득했다면, 부분 개인키 PSK_{ID^*}와 공개키 교체 후의 ID^*에 해당하는 복호화 키 DK'_{ID^*}를 획득하는 과정은 다음과 같다.

1. $Phase1$ 단계에서 획득한 공개키 교체 이전의 비밀값은 $S_{ID^*} = s$이고 복호화 키는 1.-(5)와 같다.

$$DK_{ID^*} =< D_0 = g_2^{\alpha S_{ID^*}} F_u(ID^*)^t, D_1 = g^t > \quad (7)$$

2. 이를 연산에 이용해 부분 개인키를 획득한다.

$$D_0^{\frac{1}{S_{ID^*}}} = g_2^{\alpha} F_u(ID^*)^{\frac{t}{S_{ID^*}}}, D_1^{\frac{1}{S_{ID^*}}} = g^{\frac{t}{S_{ID^*}}} \quad (8)$$

$$\therefore PSK_{ID^*} =< g_2^{\alpha} F_u(ID^*)^{\frac{t}{S_{ID^*}}}, g^{\frac{t}{S_{ID^*}}} >$$

3. 연산을 통해 얻은 부분 개인키와 $Phase1$ 단계 5.에서 획득한 비밀값 S'_{ID^*}으로 정당한 복호화 키를 계산할 수 있다. 이때, 난수 $\bar{t} \in Z_p^*$를 선택한다.

$$DK''_{ID^*} =< (g_2^{\alpha} F_u(ID^*)^{\frac{t}{S_{ID^*}}})^{S'_{ID^*}} g^{\bar{t}}, (g^{\frac{t}{S_{ID^*}}})^{S'_{ID^*}} g^{\bar{t}} >$$

$$DK'_{ID^*} =< D'_0 = g_2^{\alpha S'_{ID^*}} F_u(ID^*)^{t'}, D_1 = g^{t'} > \quad (9)$$

(이때, $t' = S'_{ID^*}(t/S_{ID^*}) + \bar{t}$ 이다.)

• 복호화 키 DK'_{ID^*}를 이용하여 챌린지 암호문 C^*을 복호화 한다. 결과적으로, A가 획득한 메시지 M_b의 b 값을 찾아 이를 출력한다.

증명을 통해 A의 행위는 정당하며 무시할 수 없는 확률로 챌린지 메시지 M_b를 구별할 수 있음을 보였다. 이로써 A가 항상 게임에서 승리한다는 사실은 명백하다.

Ⅳ. 무인증서 공개키 암호

4.1. 제안하는 기법

무인증서 공개키 암호 기법을 정의하기 위한 7가지 알고리즘은 다음과 같다. 이때, $ID \in \{0,1\}^*$, $T_{ID} \in \{0,1\}^*$, $ID \| T_{ID} \in \{0,1\}^*$이고 실제 구현 시, 충돌 저항 해시 함수 $H:\{0,1\}^* \rightarrow Z_p^*$를 사용한다. 본 논문에서는 간단한 표기를 위해 해시 함수 사용 시, 해시 함수 H의 입력값인 ID 또는 $ID \| T_{ID}$로 표기한다.

- $Setup(k)$: 설정 알고리즘은 보안 상수 k를 입력 받아 다음과 같은 공개 파라미터를 생성한다.
  - 위수가 소수 p인 군 G_1, G_2과 생성자 $g \in G_1$, 겹선형 함수 $e : G_1 \times G_1 \rightarrow G_2$를 선택한다.
  - 임의의 난수 $\alpha \in Z_p^*$와 $g_1, h_1, h_2 \in G_1$를 선택한다. 마스터 키는 g^{α}이고 공개 파라미터 PP는 다음과 같다.

$$PP = < e, g, g_1, h_1, h_2, \Omega = e(g,g)^{\alpha} > \quad (1)$$

- $Extract-Partial-Private-Key(PP, MK, ID)$: 부분 개인키 추출 알고리즘은 마스터 키와 사용자 아이디를 입력 받아 부분 개인키를 출력한다.
  - 임의의 난수 $r_1 \in Z_p^*$을 선택한다. 이때, 부분 개인키 PSK_{ID}는 다음과 같다.

$$PSK_{ID} = < K_0 = g^{\alpha}(g_1^{ID} h_1)^{r_1}, K_1 = g^{r_1} > \quad (2)$$

- $Set-Secret-Value(PP, ID)$: 비밀값 설정 알고리즘은 임의의 난수 $\beta \in Z_p^*, r_2 \in Z_p^*$를 선택하고, 현재 시간정보 T_{ID}를 이용한다. 비밀값 S_{ID}은 다음과 같다.

$$S_{ID} = < S_0 = g^{\beta}(g_1^{ID \| T_{ID}} h_2)^{r_2}, \; S_1 = g^{r_2}, S_2 = T_{ID} > \quad (3)$$

(난수 β와 시간 T_{ID}는 공개키 생성을 위해 저장한다.)

- $Set-Public-Key(PP, ID)$: 공개키 설정 알고리즘은 공개 파라미터, 사용자 아이디를 입력으로 받고, 공개키를 출력한다. 이때, 공개키 PK_{ID}는 다음과 같다.

$$PK_{ID} = < P_0 = \Omega \cdot e(g,g)^{\beta}, P_1 = T_{ID} > \quad (4)$$

$$\therefore PK_{ID} = < P_0 = e(g,g)^{\alpha+\beta}, P_1 = T_{ID} >$$

(비밀값 설정 알고리즘 수행 시, 저장해둔 난수 β와 시간정보 T_{ID}를 호출하여 이용한다.)

- $Set-Decryption-Key(PP, ID, PSK_{ID}, S_{ID})$: 복호화 키 설정 알고리즘은 공개 파라미터, 부분 개인키와 비밀값을 입력받아, 복호화 키 DK_{ID}을 출력한다.
  - 임의의 난수 $r'_1, r'_2 \in Z_p^*$을 선택하고 시간정보 T_{ID}는 비밀값 구성요소인 S_2를 이용한다. 이때, 복호화 키 DK_{ID}는 다음과 같다.

$$DK_{ID} = < D_0 = K_0 \cdot S_0 \cdot (g_1^{ID} h_1)^{r'_1} \cdot (g_1^{ID \| T_{ID}} h_2)^{r'_2}, \; D_1 = K_1 \cdot g^{r'_1}, D_2 = S_1 \cdot g^{r'_2} >$$

$$\therefore DK_{ID} = < D_0 = g^{\alpha+\beta}(g_1^{ID} h_1)^{\hat{r_1}}(g_1^{ID \| T_{ID}} h_2)^{\hat{r_2}}, \; D_1 = g^{\hat{r_1}}, D_2 = g^{\hat{r_2}} > \quad (5)$$

(이때, 난수는 $\hat{r_1} = r_1 + r'_1$과 $\hat{r_2} = r_2 + r'_2$이 된다.)

- $Encrypt(PP, ID, PK_{ID}, M)$: 암호화 알고리즘은 사용자 아이디에 대한 공개키를 이용해 메시지를 암호화한 후 암호문을 출력한다.
  - 난수 $s \in Z_p^*$을 선택하고 암호문은 다음과 같다.

$$C = < C_0 = M \cdot P^s, C_1 = (g_1^{ID} h_1)^s, \; C_2 = (g_1^{ID \| T_{ID}} h_2)^s, C_3 = g^s > \quad (6)$$

- $Decrypt(PP, DK_{ID}, C)$: 복호화 알고리즘은 복호화 키를 이용하여 메시지를 구한다.

$$C_0 \frac{e(C_1, D_1) e(C_2, D_2)}{e(D_0, C_3)} = M \quad (7)$$

4.2. 정확성

복호화 알고리즘을 수행하여 메시지를 구하는 과정은 다음과 같다.

$$C_0 \frac{e(C_1, D_1) e(C_2, D_2)}{e(D_0, C_3)} \quad (8)$$

$$= M \cdot P^s \frac{e((g_1^{ID} h_1)^s, g^{\hat{r_1}}) e((g_1^{ID \| T_{ID}} h_2)^s, g^{\hat{r_2}})}{e(g^{\alpha+\beta}(g_1^{ID} h_1)^{\hat{r_1}}(g_1^{ID \| T_{ID}} h_2)^{\hat{r_2}}, g^s)}$$

$$= M \cdot P^s \frac{e((g_1^{ID} h_1)^s, g^{\hat{r_1}}) e((g_1^{ID \| T_{ID}} h_2)^s, g^{\hat{r_2}})}{e(g^{\alpha+\beta}, g^s) e((g_1^{ID} h_1)^{\hat{r_1}}, g^s) e((g_1^{ID \| T_{ID}} h_2)^{\hat{r_2}}, g^s)}$$

$$= M \cdot P^s \frac{e(g_1^{ID} h_1, g)^{\hat{r_1} s} e(g_1^{ID \| T_{ID}} h_2, g)^{\hat{r_2} s}}{e(g^{\alpha+\beta}, g^s) e(g_1^{ID} h_1, g)^{\hat{r_1} s} e(g_1^{ID \| T_{ID}} h_2, g)^{\hat{r_2} s}}$$

$$= M \bullet (e(g,g)^{\alpha} e(g,g)^{\beta})^{s} \frac{1}{e(g^{\alpha+\beta}, g^{s})}$$
$$= M \bullet e(g,g)^{s(\alpha+\beta)} \frac{1}{e(g,g)^{s(\alpha+\beta)}}$$
$$= M$$

4.3. 안전성 증명

4.3.1. 선택적인 ID 모델- 공격자 유형 I

정리 1. 제안하는 무인증서 암호 기법은 선택적인 ID 모델에서의 공격자 유형 I 으로부터 DBDH 문제가 어렵다는 가정 하에 IND-sID-CPA 관점에서 안전하다.

증명. 제안하는 무인증서 암호 기법의 IND-sID-CPA 안전성을 의미 있는(non-negligible) 확률로 깰 수 있는 공격자 A가 존재한다고 가정하자. 이 공격자 A는 DBDH 문제를 효율적으로 해결할 수 있는 챌린저 B를 이용하여 게임을 수행한다. 챌린저 B는 임의로 선택된 $a,b,c \in Z_p^*$와 $g \in G_1$를 이용하여 DBDH 문제 $< g, g^a, g^b, g^c, G_1, T >$를 입력 받고, $T = e(g,g)^{abc}$를 결정한다. 챌린저가 공격자와 상호작용하며 진행하는 선택적인 ID 게임과정은 다음과 같다.

- $init$: 공격자 A는 공격할 ID^*를 선택하여 챌린저 B에게 전송한다. B는 챌린지 암호문의 공개키 시간정보 $T_{ID^*}^* \in Time$를 추측한다.

(이때, $1/|T|$ 확률의 안전성 손실이 발생한다.)

- $Setup$: 챌린저 B는 공개 파라미터 PP를 생성하기 위해 임의로 $\alpha_1, \alpha_2 \in Z_p^*$를 선택하여 $h_1 = g_1^{-ID^*} g^{\alpha_1}$과 $h_2 = g_1^{-(ID^* \| T_{ID^*}^*)} g^{\alpha_2}$를 정의한다. $g_1 = g^a$로 설정하고 B는 A에게 $PP = < e, g, g_1, h_1, h_2, \Omega = e(g^a, g^b) >$를 전송한다. 이때, 마스터키 $MK = g^{ab}$로 설정되지만 챌린저가 마스터키를 계산할 수는 없다.
- $Phase\,1$: 챌린저 B는 공격자 A의 질의들에 대해 다음과 같이 응답한다.

  • $Extract-Partial-Private-Key(ID)$: 공격자 A가 $ID(\neq ID^*)$에 해당하는 부분 개인키를 요청할 때, 챌린저 B는 임의의 난수 $r_1 \in Z_p^*$를 선택하고 마스터키와 공개 파라미터를 이용해 다음과 같이 부분 개인키 PSK_{ID}를 생성한다.

$$PSK_{ID} = < K_0 = g^{ab} (g_1^{ID} h_1)^{r_1}, K_1 = g^{r_1} > \quad (9)$$

챌린저 B가 마스터키를 연산하지 않고 정당한 부분 개인키를 생성하기 위해 임의의 난수 $\tilde{r_1} \in Z_p^*$를 선택한다. 이때, 부분 개인키 PSK_{ID}는 다음과 같다.

$$\begin{aligned} K_0 &= g^{ab} (g_1^{ID} h_1)^{r_1} = g^{ab} g^{a(ID-ID^*)r_1} g^{\alpha_1 r_1} \\ &= g^{ab} g^{-ab} g^{-\alpha_1 b/(ID-ID^*)} g^{a(ID-ID^*)\tilde{r_1}} g^{\alpha_1 \tilde{r_1}} \\ &= (g^b)^{-\alpha_1/(ID-ID^*)} ((g^a)^{(ID-ID^*)} g^{\alpha_1})^{\tilde{r_1}} \\ K_1 &= (g^b)^{-1/(ID-ID^*)+\tilde{r_1}} \end{aligned} \quad (10)$$

($r_1 = -b/(ID-ID^*) + \tilde{r_1}$식을 암묵적으로 정의한다.)

• $Extract-Secret-Value(ID)$: A가 비밀값을 요청할 때, B는 임의의 난수 $\beta, r_2 \in Z_p^*$를 선택하고 공개 파라미터를 이용해 다음과 같이 비밀값 S_{ID}를 생성한다. g^{ab} 연산이 필요하지 않아 쉽게 생성 가능하다.

$$\begin{aligned} S_{ID} = < S_0 &= g^{\beta} (g_1^{ID \| T_{ID}} h_2)^{r_2}, \\ S_1 &= g^{r_2}, S_2 = T_{ID} > \end{aligned} \quad (11)$$

• $Request-Public-Key(ID)$: B는 공개키 설정 알고리즘으로부터 사용자 아이디에 대한 공개키 PK_{ID}을 출력 받아 A에게 전송한다.

$$\begin{aligned} PK_{ID} &= < P_0 = e(g,g)^{\alpha+\beta}, P_1 = T_{ID} > \\ &= < P_0 = e(g^a, g^b) e(g,g)^{\beta}, P_1 = T_{ID} > \end{aligned} \quad (12)$$

• $Extract-Decryption-Key(ID)$: A가 사용자 아이디에 대한 복호화 키를 요청할 때, B는 임의의 난수 $\beta, \hat{r_1}, \hat{r_2} \in Z_p^*$를 선택하여 다음과 같이 복호화 키 DK_{ID}를 생성한다.

$$\begin{aligned} DK_{ID} = < D_0 &= g^{\alpha+\beta} (g_1^{ID} h_1)^{\hat{r_1}} (g_1^{ID \| T_{ID}} h_2)^{\hat{r_2}}, \\ D_1 &= g^{\hat{r_1}}, D_2 = g^{\hat{r_2}} > \end{aligned} \quad (13)$$

B가 $MK(= g^{ab})$를 연산하지 않고 정당한 복호화 키를 생성해주어야 하므로, 만약 $ID \neq ID^*$이면 임의의 난수 $r'_1, r_2 \in Z_p^*$를 선택하여 복호화 키를 생성한다.

$$\begin{aligned} D_0 &= g^{ab+\beta} \bullet (g_1^{ID} h_1)^{\hat{r_1}} \bullet (g_1^{ID \| T_{ID}} h_2)^{r_2} \\ &= g^{ab} g^{\beta} \bullet g^{a(ID-ID^*)\hat{r_1}} g^{\alpha_1 \hat{r_1}} \bullet (g_1^{ID \| T_{ID}} h_2)^{r_2} \end{aligned} \quad (14)$$

$= g^{ab}g^{\beta} \cdot g^{-ab+ar'_1(ID-ID^*)}$
$\cdot\ g^{-b\alpha_1/(ID-ID^*)+\alpha_1 r'_1} \cdot (g_1^{ID\|T_{ID}}h_2)^{r_2}$
$= g^{\beta}(g^a)^{r'_1(ID-ID^*)}(g^b)^{-\alpha_1/(ID-ID^*)}g^{\alpha_1 r'_1} \cdot (g_1^{ID\|T_{ID}}h_2)^{r_2}$

$D_1 = (g^b)^{-1/(ID-ID^*)+r'_1}$, $D_2 = g^{r_2}$

($\hat{r_1} = -b/(ID-ID^*)+r'_1$식을 암묵적으로 정의한다.)

혹은 $ID = ID^*$ 이면, 임의의 난수 $r_1, r'_2 \in Z_p^*$를 선택한다. 이때, 정당한 복호화 키 DK_{ID}는 다음과 같다.

$$D_0 = g^{ab+\beta} \cdot (g_1^{ID^*}h_1)^{r_1} \cdot (g_1^{ID^*\|T_{ID^*}}h_2)^{\hat{r_2}}$$
$$= g^{ab}g^{\beta} \cdot (g_1^{ID^*}h_1)^{r_1} \cdot g^{a(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})\hat{r_2}}g^{\alpha_2\hat{r_2}} \quad (15)$$
$$= g^{ab}g^{\beta} \cdot (g_1^{ID^*}h_1)^{r_1} \cdot g^{-ab+ar'_2(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})}$$
$$g^{-b\alpha_2/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})+\alpha_2 r'_2}$$
$$= g^{\beta} \cdot (g_1^{ID^*}h_1)^{r_1} \cdot (g^a)^{r'_2(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})}$$
$$(g^b)^{-\alpha_2/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})}g^{\alpha_2 r'_2}$$

$D_1 = g^{r_1}$, $D_2 = (g^b)^{-1/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})+r'_2}$

(이때, 식 $\hat{r_2} = -b/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})+r'_2$을 암묵적으로 정의한다.)

• $Replace-Public-Key(ID, PK_{ID}')$: A는 사용자 $ID(\neq ID^*)$에 해당하는 공개키를 선택한 공개키 $PK_{ID(\neq ID^*)}'$로 교체할 수 있다.

- $Challenge$: A는 길이가 같은 두 메시지 M_0, M_1을 선택하여 B에게 전송하면 B는 $b \in \{0,1\}$를 임의로 뽑아 선택한 ID^*에 대한 PK_{ID^*}을 이용해 M_b에 관한 챌린지 암호문을 계산하여 A에게 전송한다. 이때, C_0, C_1과 C_2의 연산과정은 다음과 같다.

$$C^* = < C_0 = M_b \cdot T \cdot e(g,g^c)^{\beta}, C_1 = (g^c)^{\alpha_1}, \quad (16)$$
$$C_2 = (g^c)^{\alpha_2}, C_3 = g^c >$$
$$C_0 = M_b e(g,g)^{(ab+\beta)c}$$
$$= M_b e(g,g)^{abc}e(g,g)^{\beta c} = M_b T e(g,g^c)^{\beta},$$
$$C_1 = (g_1^{ID^*}h_1)^c = (g^{a(ID^*-ID^*)}g^{\alpha_1})^c = (g^c)^{\alpha_1},$$
$$C_2 = (g_1^{ID^*\|T^*_{ID^*}}h_2)^c = (g^{a(ID^*\|T^*_{ID^*}-ID^*\|T^*_{ID^*})}g^{\alpha_2})^c = (g^c)^{\alpha_2}$$

- $Phase\,2$: $Phase\,1$에서 요청하지 않았던 질의들에 대해 $Phase\,1$과 같은 방식으로 응답한다.
- $Guess$: A는 b에 대한 응답으로 $b' \in \{0,1\}$을 챌린저 B에게 전송한다. 만약 $b = b'$ 이면 $T = e(g,g)^{abc}$를 의미하므로 공격자가 게임에서 이긴다.

공격자가 게임에서 이겼다는 것은 챌린저가 공격자를 이용해 DBDH 문제를 해결했다는 것이다. 다시 말해 DBDH 문제를 푸는 챌린저 B의 이점이 게임에서 공격자 A의 이점보다 크거나 같다.

$$Adv_B^{DBDH}(k) \geq 1/|T| \cdot Adv_{A_1}^{CLE}(k)$$
$$= 1/|T| \cdot \left|\Pr[b' = b] - \frac{1}{2}\right| \quad (17)$$

4.3.2. 선택적인 ID 모델- 공격자 유형Ⅱ

정리 2. 제안하는 무인증서 암호 기법은 선택적인 ID 모델에서 공격자 유형Ⅱ로부터 DBDH 문제가 어렵다는 가정 하에 IND-sID-CPA 관점에서 안전하다.

증명. 제안하는 무인증서 암호 기법의 IND-sID-CPA 안전성을 의미 있는(non-negligible) 확률로 깰 수 있는 공격자 A가 존재한다고 가정하자. 이 공격자 A는 DBDH 문제를 효율적으로 해결할 수 있는 챌린저 B를 이용하여 게임을 수행한다. 챌린저 B는 임의로 선택된 $a,b,c \in Z_p^*$와 $g \in G_1$를 이용하여 DBDH 문제 $< g, g^a, g^b, g^c, G_1, T >$를 입력 받고, $T = e(g,g)^{abc}$를 결정한다. 챌린저가 공격자와 상호작용하며 진행하는 선택적인 ID 게임과정은 다음과 같다.

- $init$: 공격자 A는 공격할 ID^*를 선택하여 챌린저 B에게 전송한다. B는 챌린지 암호문의 공개키 시간정보 $T^*_{ID^*} \in Time$를 추측한다.
  (이때, $1/|T|$ 확률의 안전성 손실이 발생한다.)
- $Setup$: 챌린저 B는 공개 파라미터 PP를 생성하기 위해 임의의 난수로 $\alpha, \alpha_1, \alpha_2 \in Z_p^*$를 선택하여 $h_1 = g^{-aID^*}g^{\alpha_1}$, $h_2 = g^{-a(ID^*\|T^*_{ID^*})}g^{\alpha_2}$를 정의한다. 이때, $g_1 = g^a$를 설정하고, 마스터 키는 $MK = g^{\alpha}$로 설정한 후, $PP = < e, g, g_1, h_1, h_2, \Omega = e(g,g)^{\alpha} >$와 마스터키를 공격자에게 전송한다.
- $Phase\,1$: 챌린저 B는 공격자 A의 질의들에 대해 다음과 같이 응답한다. 공격자 유형Ⅱ는 공개키 교체는 할 수 없지만 마스터키를 알고 있기 때문에 부분 개인키 추출 질의를 할 필요가 없다.
  • $Extract-Secret-Value(ID)$: A가 비밀값을 요청할 때, B는 임의의 난수 $\beta, r_2 \in Z_p^*$를 선택하고 공

개 파라미터를 이용해 다음과 같이 비밀값 S_{ID}를 생성한다. g^{ab} 연산이 필요하지 않아 쉽게 생성 가능하다.

$$S_{ID} =< S_0 = g^{\beta}(g_1^{ID\|T_{ID}}h_2)^{r_2}, \quad S_1 = g^{r_2}, S_2 = T_{ID} > \tag{18}$$

• $Request-Public-Key(ID)$: B는 공개키 설정 알고리즘으로부터 사용자 아이디에 대한 공개키 PK_{ID}을 출력 받아 A에게 전송한다. 만약, $T_{ID} = T^*_{ID^*}$ 이면, 다음과 같이 계산한다.

$$PK_{ID} =< P_0 = e(g,g)^{\alpha+\beta}, P_1 = T_{ID} > \tag{19}$$
$$=< P_0 = e(g,g)^{\alpha}e(g^a,g^b), P_1 = T_{ID} >$$

그렇지 않다면, 임의의 난수 $\beta \in Z_p^*$를 선택하여 다음과 같이 계산한다.

$$PK_{ID} =< P_0 = e(g,g)^{\alpha+\beta}, P_1 = T_{ID} > \tag{20}$$

• $Extract-Decryption-Key(ID)$: A가 사용자 아이디에 대한 복호화 키를 요청하면, B는 난수 $\hat{r_1}, \hat{r_2} \in Z_p^*$를 선택하여 복호화 키 DK_{ID}를 생성한다.

$$DK_{ID} =< D_0 = g^{\alpha+ab}(g_1^{ID}h_1)^{\hat{r_1}}(g_1^{(ID\|T_{ID})}h_2)^{\hat{r_2}}, \quad D_1 = g^{\hat{r_1}}, D_2 = g^{\hat{r_2}} > \tag{21}$$

B가 $MK(=g^{ab})$를 연산하지 않고 정당한 복호화 키를 생성해주어야 하므로, 만약 $ID \neq ID^*$이면 임의의 난수 $r'_1, r_2 \in Z_p^*$를 선택하여 복호화 키를 생성한다.

$$D_0 = g^{\alpha+ab} \cdot (g_1^{ID}h_1)^{\hat{r_1}} \cdot (g_1^{ID\|T_{ID}}h_2)^{r_2} \tag{22}$$
$$= g^{\alpha}g^{ab} \cdot g^{a(ID-ID^*)(-b/(ID-ID^*)+r'_1)} \cdot g^{\alpha_1(-b/(ID-ID^*)+r'_1)} \cdot (g_1^{ID\|T_{ID}}h_2)^{r_2}$$
$$= g^{\alpha}g^{ab} \cdot g^{-ab+ar'_1(ID-ID^*)}g^{-b\alpha_1/(ID-ID^*)+\alpha_1 r'_1} \cdot (g_1^{ID\|T_{ID}}h_2)^{r_2}$$
$$= g^{\alpha}(g^a)^{r'_1(ID-ID^*)}(g^b)^{-\alpha_1/(ID-ID^*)}g^{\alpha_1 r'_1} \cdot (g_1^{ID\|T_{ID}}h_2)^{r_2}$$

$D_1 = (g^b)^{-1/(ID-ID^*)+r'_1}$, $D_2 = g^{r_2}$

($\hat{r_1} = -b/(ID-ID^*)+r'_1$식을 암묵적으로 정의한다.)

혹은 $ID = ID^*$이면, 임의의 난수 $r_1, r'_2 \in Z_p^*$를 선택한다. 이때, 정당한 복호화 키 DK_{ID}는 다음과 같다.

$$D_0 = g^{\alpha+ab} \cdot (g_1^{ID^*}h_1)^{r_1} \cdot (g_1^{ID^*\|T_{ID^*}}h_2)^{\hat{r_2}} \tag{23}$$
$$= g^{\alpha}g^{ab} \cdot (g_1^{ID^*}h_1)^{r_1} \cdot g^{a(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})(-b/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})+r'_2)} \cdot g^{\alpha_2(-b/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})+r'_2)}$$
$$= g^{\alpha}g^{ab} \cdot (g_1^{ID^*}h_1)^{r_1} \cdot g^{-ab+ar'_2(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})} g^{-b\alpha_2/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})+\alpha_2 r'_2}$$
$$= g^{\alpha} \cdot (g_1^{ID^*}h_1)^{r_1} \cdot (g^a)^{r'_2(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})} (g^b)^{-\alpha_2/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})} g^{\alpha_2 r'_2}$$

$D_1 = g^{r_1}$, $D_2 = (g^b)^{-1/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})+r'_2}$

(이때, 식 $\hat{r_2} = -b/(ID^*\|T_{ID^*}-ID^*\|T^*_{ID^*})+r'_2$을 암묵적으로 정의한다.)

- $Challenge$: A는 길이가 같은 두 메시지 M_0, M_1을 선택하여 B에게 전송하면 B는 $b \in \{0,1\}$를 임의로 뽑아 선택한 ID^*에 대한 PK_{ID^*}을 이용해 M_b에 관한 챌린지 암호문을 계산하여 A에게 전송한다. 이때, C_0, C_1과 C_2의 연산과정은 다음과 같다.

$$C^* =< C_0 = M_b \cdot T \cdot e(g^c,g)^{\alpha}, C_1 = (g^c)^{\alpha_1}, \tag{24}$$
$$C_2 = (g^c)^{\alpha_2}, C_3 = g^c >$$
$$C_0 = M_b e(g,g)^{(\alpha+ab)c}$$
$$= M_b e(g,g^{abc})e(g,g^{\alpha c}) = M_b T e(g^c,g)^{\alpha})$$
$$C_1 = (g^{\alpha ID^*}h_1)^c = (g^{\alpha(ID^*-ID^*)}g^{\alpha_1})^c = (g^c)^{\alpha_1}$$
$$C_2 = (g^{\alpha(ID^*\|T^*_{ID^*})}h_2)^c = (g^{\alpha(ID^*\|T^*_{ID^*}-ID^*\|T^*_{ID^*})}g^{\alpha_2})^c = (g^c)^{\alpha_2}$$

- $Phase\,2$: $Phase\,1$에서 요청하지 않았던 질의들에 대해 $Phase\,1$과 같은 방식으로 응답한다.
- $Guess$: A는 b에 대한 응답으로 $b' \in \{0,1\}$을 챌린저 B에게 전송한다. 만약 $b = b'$이면 $T = e(g,g)^{abc}$를 의미하므로 공격자가 게임에서 이긴다.

공격자가 게임에서 이겼다는 것은 챌린저가 공격자를 이용해 DBDH 문제를 해결했다는 것이다. 정리 1과 같이 공격자 A의 이점은 다음과 같이 정의한다.

$$Adv_B^{DBDH}(k) \geq 1/|T| \cdot Adv_{A_2}^{CLE}(k) = 1/|T| \cdot \left| \Pr[b' = b] - \frac{1}{2} \right| \tag{25}$$

정리 1., 정리 2.에 의해 제안된 무인증서 암호 기법은 제안된 무인증서 암호 안전성 모델에서 안전하다.

Ⅴ. 논 의

5.1. 계층적 무인증서 공개키 암호

본 논문에서 제안한 무인증서 공개키 암호 기법은 Boneh-Boyen의 계층적 아이디 기반 암호 기법(Hierarchical IBE, HIBE) [14]의 구조이며, 다음과 같이 부분 개인키 $PSK_{I_1 \cdots I_k}$와 암호문 $C_{I_1 \cdots I_{k'}}$의 구조로 계층적 무인증서 공개키 암호 기법으로 확장 가능하다.

$$PSK_{I_1 \cdots I_k} =< K_1 = g_2^{\alpha}(g_1^{I_1}h_1)^{r_1} \bullet \ldots \bullet (g_1^{I_k}h_1)^{r_k}, \quad (1)$$
$$K_{2,1} = g^{r_1}, \ldots, K_{2,k} = g^{r_k} >$$
$$C_{I_1 \cdots I_{k'}} =< C_0 = M \bullet P^s, C_{1,1} = (g_1^{I_1}h_1)^s, \ldots,$$
$$C_{1,k'} = (g_1^{I_{k'}}h_1)^s,\ C_2 = (g_1^{I_1 || \ldots || I_{k'} || T_{I_1}}h_2)^s, C_3 = g^s >$$

5.2. 선택 암호문 공격(CCA) 안전성

본 논문에서 제안한 무인증서 공개키 암호는 IND-sID-CPA 모델에서 안전성을 증명하였다. Dent 등 [8]이 제안한 CCA에 안전한 무인증서 공개키 암호의 설계 방법을 적용하면 본 논문에서 제안한 무인증서 공개키 암호 기법은 IND-sID-CCA에 안전한 무인증서 공개키 암호 기법으로 확장 가능하다. 충돌저항성의 성질을 갖는 해시 함수 H가 있다고 가정하자. CCA에 안전한 무인증서 암호기법의 암호문 C 구조는 다음과 같다.

$$C =< C_0 = M \bullet P^s, C_1 = (g_1^{ID}h_1)^s, \quad (2)$$
$$C_2 = (g_1^{ID || T_{ID}}h_2)^s, C_3 = (g_1^{w}h_3)^s,\ C_4 = g^s >$$
$$\text{where } w \leftarrow H(C_0, C_1, C_2, C_4, ID, PK_{ID})$$

위 기법의 안전성 증명은 IND-sID-CPA 모델에서의 증명에 추가적으로 암호문에 대한 복호화 질의를 시뮬레이션 해줄 수 있어야 한다. 챌린저는 복호화 키 $(g_1^{w}h_3)^{\hat{r_3}}$의 부분에서 w를 이용한 Boneh-Boyen의 마스터 키 소거 기술(cancelation technique)을 이용하여 정당한 복호화 키 생성이 가능하기 때문에 복호화 질의의 응답할 수 있다. 선택적인 ID 모델에서 챌린지 암호문은 미리 결정되며, w^*는 공개 파라미터 생성 시 계산되어 $h_3 = g_1^{-w^*}g^{\alpha_1}$으로 설정된다. 암호문의 w는 해시의 충돌저항성 성질에 의하여 w^*와 같을 수 없다.

5.3. 키 노출 공격을 고려한 일반적인 무인증서 암호

본 논문에서는 공개키 교체 이전의 비밀값과 복호화 키가 노출되어도 안전한 무인증서 암호 기법을 설계하기 위해서 재랜덤화가 가능한 복호화키를 만들고, 비밀값에 시간정보를 결합하였다. 하지만, 시간정보의 사용은 기존의 무인증서 암호 기법과 차이가 발생한다. 그렇다 하더라도, 본 논문에서 고려하는 안전성 모델은 기존의 무인증서 암호에서 제공하지 못한 안전성을 보장함으로써 논의될 가치가 충분하고 유의미하다. 따라서, 키 노출 공격을 고려하면서 시간정보를 사용하지 않는 일반적인 무인증서 암호 기법의 설계가 필요하다.

5.4. 풀-모델 안전성

암호 기법을 설계할 때 풀-모델에서 안전성이 증명이 되는 기법을 설계하는 것은 매우 중요하다. 본 논문에서 제안한 무인증서 공개키 암호 기법은 선택적인 ID 모델에서 안전성이 증명된다. 본 논문에서 제안한 기법은 기본적으로 Boneh-Boyen의 ID기반 암호 기법 [14]의 구조를 따르기 때문이다. ID의 해시 구조를 Waters의 IBE 기법 구조로 변환하면 풀-모델에서 안전한 무인증서 공개키 암호 기법의 설계가 가능할 것으로 보인다. 하지만, 이에 대한 명확한 증명이 필요하다.

Ⅵ. 결 론

본 논문에서는 공개키가 교체된 이후 이전의 비밀값과 복호화 키의 노출을 고려한 새로운 안전성 모델을 제안하였고, 제안한 안전성 모델에서 비밀값과 복호화 키 노출 공격에 대해 기존의 무인증서 공개키 암호 기법들이 안전하지 않음을 보였다. 또한, 비밀값과 복호화 키가 노출되더라도 안전한 무인증서 공개키 암호 기법을 제안하고 DBDH 가정을 기반으로 안전성을 증명하였다. 향후에, 풀-모델에서 안전한 무인증서 공개키 암호 기법을 설계하고 증명하는 연구가 필요하다.

ACKNOWLEDGMENTS

This research was supported by the MSIP (Ministry of Science, ICT & Future Planning), Korea, under the "The Types of employment contract to support master's degree in Information Security" supervised by the KISA(Korea Internet Security Agency). This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (No.R0126-16-1090, A study of a public-key authentication framework for internet entities with hierarchical identities)

REFERENCES

[1] A. Shamir, “Identity-Based Cryptosystems and Signature Schemes,” in *Proceedings of CRYPTO*, vol. 196, pp. 47-53, 1985.

[2] S. S. Al-Riyami and K. G. Paterson, “Certificateless Public Key Cryptography,” in *Proceedings of ASIACRYPT*, vol. 2894, pp. 452-473, Dec. 2003.

[3] S. S. Al-Riyami and K. G. Paterson, “A Generic Construction and Efficient Schemes,” in *Proceedings of Public Key Cryptography*, vol. 3386, pp. 398-415, Jan. 2005.

[4] B. Libert and J. J. Quisquater, “On Constructing Certificateless Cryptosystems from Identity Based Encryption,” in *Proceedings of Public-Key Cryptography*, vol. 3958, pp. 474-490, Apr. 2006.

[5] Z. Zhang and D. Feng, “Key Replacement Attack on a Certificateless Signature Scheme,” in *Proceedings of IACR Cryptology ePrint Archive*, pp. 1-5, 2006.

[6] E. Fujisaki and T. Okamoto, “How to enhance the security of public-key encryption at minimum cost,” in *Proceedings of Public Key Cryptography*, vol. 1560, pp. 53-68, Mar. 1999.

[7] Y. Shi and J. Li, “Provable Efficient Certificateless Public Key Encryption,” in *Proceedings of IACR Cryptology ePrint Archive*, pp. 1-15, 2005.

[8] A. W. Dent, B. Libert and K. G. Paterson, “Certificateless Encryption Schemes Strongly Secure in the Standard Model,” in *Proceedings of Public-Key Cryptography*, vol. 4939, pp. 344-359, Mar. 2008.

[9] H. Xiong, T. H. Yuen, C. Zhang, S. M. Yiu and Y. -J. He, “Leakage-resilient certificateless public key encryption,” in *Proceedings of ACM AsiaCCS*, pp. 13-22, 2013.

[10] Y. Sun, F. Zhang, L. Shen and R. H. Deng, “Efficient revocable certificateless encryption against decryption key exposure,” in *Proceedings of IET information security*, vol. 9, no. 3, pp. 158-166, May 2015.

[11] J. Baek, R. Safavi-Naini and W. Susilo, “Certificateless Public Key Encryption Without Pairing,” in *Proceedings of the 8th International Conference on Information Security*, vol. 3650, pp. 134-148, Sep. 2005.

[12] J. Lai and W. Kou, “Self-Generated-Certificate Public Key Encryption Without Pairing,” in *Proceedings of Public-Key Cryptography*, vol. 4450, pp. 476-489, Apr. 2007.

[13] D. Boneh, “The Decision Diffie-Hellman problem,” in *Algorithmic number theory*, *vol. 1423*, pp. 48-63, Jun. 1998.

[14] D. Boneh and X. Boyen, “Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles,” in *Proceedings of EUROCRYPT*, vol. 3027, pp. 223-238, May 2004.

김송이(Songyi Kim)

2015.2 성신여자대학교 IT학부 학사
2015.3 - 현재 고려대학교 정보보호대학원 석사과정
※관심분야 : 금융보안, 암호 프로토콜, 공개키 암호

박승환(Seunghwan Park)

2009.2 숭실대학교 수학과 학사
2011.8 고려대학교 정보경영공학대학원 정보보호학과 석사
2011.9 - 현재 고려대학교 정보보호대학원 정보보호학과 박사과정
※관심분야 : 암호 프로토콜, 공개키 암호

이광수(Kwangsu Lee)

1998.2 연세대학교 컴퓨터과학사 학사
2000.2 한국과학기술원 전자전산학과 석사
2011.2 고려대학교 정보경영공학사 박사
2014.9 - 현재 고려대학교 정보보호대학원 조교수
※관심분야 : 암호 프로토콜, 공개키 암호