Journal of Korea Multimedia Society Vol. 12, No. 4, April 2009(pp. 530-539)

무인증서기반 프락시 재암호화 기법 및 다중 KGC 환경으로의 확장

서철[†], 정채덕[††], 박영호[†††], 이경현[††††]

요 약

본 논문에서는 프락시 재암호화 기술의 특성을 제공하면서 무인증서기반 암호 기술의 장점을 활용하기 위하여 무인증서기반 프락시 재암호화 기술의 개념을 소개하고, Bilinear Pairing을 이용한 안전한 무인증서기반 프락시 재암호화 기법을 설계한다. 제안 기법은 단방향성을 제공할 뿐만 아니라 기존의 무인증서 기반 암호 환경에 적합하도록 설계되었다. 또한, 제안 기법에 대하여 랜덤 오라클 모델에서 선택 암호문 공격에 대한 안전성을 증명한다. 마지막으로, 제안 기법을 다중 KGC 환경에 적합한 무인증서기반 단방향 프락시 재암호화 기법으로 확장한다.

Certificateless Proxy Re-Encryption Scheme and Its Extension to Multiple KGC Environment

Chul Sur[†], Chae Duk Jung[††], Youngho Park[†††], Kyung Hyune Rhee[††††]

ABSTRACT

In this paper we introduce the notion of certificateless proxy re-encryption which enjoys the advantages of certificateless cryptography while providing the functionalities of proxy re-encryption. We give precise definitions for secure certificateless proxy re-encryption schemes and also present a concrete scheme from bilinear pairing. Our scheme is unidirectional and compatible with current certificateless encryption deployments. In addition, we show that our scheme has chosen ciphertext security in the random oracle model. Finally, we extend the proposed scheme for appling multiple KGC environment.

Key words: Certificateless Encryption(무인증서기반 암호), Proxy Re-Encryption(프락시 재암호), Bilinear Pairing(Bilinear Pairing)

1. 서 론

프락시 재암호화 (PRE, Proxy Re-Encryption) 기술에서 프락시는 특정 사용자 (위임자)의 공개키로 암호화된 암호문을 다른 사용자 (피위임자)의 개인키로 복호화할 수 있도록 암호문을 변형하는 역할을 수행하지만, 이러한 변형 단계 (재암호화 단계)에서 프락시는 암호문 또는 재암호문으로부터 평문에 관한 정보를 알 수 없어야 한다. 이와 같은 프락시 재암호화 기술의 특성으로 인하여, 프락시 재암호화 기술은 분산 파일 시스템, 안전한 메일 포워딩 및 Interoperable DRM 시스템과 같은 서비스에서 효율

※ 교신저자(Corresponding Author) : 이경현, 주소 : 부산 남구 대연3동 599-1번지(608-737), 전화 : 051)626-4887, FAX : 051)626-4887, E-mail : khrhee@pknu.ac.kr
접수일 : 2008년 10월 27일, 완료일 : 2009년 1월 28일
[†] 부경대학교 전자계산학과 박사과정
(E-mail : kahlil@pknu.ac.kr)
[††] 준회원, 부경대학교 정보보호학 박사과정
(E-mail : jcd0205@pknu.ac.kr)
[†††] 준회원, 부경대학교 전자컴퓨터정보통신공학부 박사
(E-mail : pyhoya@pknu.ac.kr)
[††††] 종신회원, 부경대학교 전자컴퓨터정보통신공학부 교수
※ 본 연구는 ETRI 부설연구소 위탁과제 연구결과로 수행되었음.

적인 키 관리를 위한 암호학적 기반기술로 응용될 수 있다[1-3].

[1]에서 Blaze외 저자들은 처음으로 프락시 재암호화 기술의 개념을 소개하고, 프락시로부터 평문과 비밀키들의 정보를 노출시키지 않는 프락시 재암호화 기법을 설계하였다. 하지만, Blaze 등이 제안한 기법은 양방향 (Bidirectional) 특성으로 인하여, 위임자로부터 피위임자로의 암호문 변형뿐만 아니라, 역방향 (피위임자로부터 위임자)의 변형도 가능하다[2,3]. 그러나, 프락시 재암호화 기법의 설계시 단방향 (Unidirectional) 프락시 재암호화 기법을 두 번 사용함으로서 양방향 프락시 재암호화 기법과 동일한 기능을 제공할 수 있기 때문에, 원천기술로서 단방향 프락시 재암호화 기법을 설계하는 것이 보다 더 유용하다. 게다가, 양방향 프락시 재암호화 기법들은 피위임자와 프락시의 공모를 통하여 위임자의 개인키 정보를 유출하는 공모 공격 (Collusion Attack)에 취약성을 나타낸다. 이후, Ateniese외 저자들은 Bilinear Pairing을 이용하여 처음으로 단방향 프락시 재암호화 기법을 설계하였다[2]. 하지만, Ateniese 등이 제안한 기법은 공모 공격에는 안전하지만, 선택 평문 공격 (CPA, Chosen Plaintext Attack)에서만 안전성이 보장된다.

Canetti와 Hohenberger는 선택 평문 공격에만 안전한 기법은 다양한 응용기술로의 적용에 충분한 안정성을 제공하지 못함을 제시하고, 선택 암호문 공격 (CCA, Chosen Ciphertext Attack)에 안전한 프락시 재암호화 기법을 소개하였다[3]. 또한, [3]에서는 프락시 재암호화 기법의 정형화된 보안 모델을 제안하고 스탠다드 모델 (Standard Model)에서 제안 기법의 안전성을 증명하였다. 하지만, Canetti와 Hohenberger의 기법은 양방향 기법으로서 기존의 양방향 프락시 재암호화 기법들처럼 여전히 공모 공격에 취약성을 가지고 있다. 최근, Green과 Ateniese는 신원기반 프락시 재암호화 기법의 설계시 문제점을 지적하고, 랜덤 오라클 모델에서 선택 암호문 공격에 안전한 신원기반 단방향 프락시 재암호화 기법을 설계하였다[4]. 하지만, 제안 기법에 사용되는 재암호화 키 설정 기술의 특성으로 인하여 공모 공격에 취약성을 가지고 있다.

비록, 최근까지 많은 프락시 재암호화 기법들이 제안되었지만, 기 제안된 모든 프락시 재암호화 기법들은 전통적인 공개키기반 암호 기법[1-3] 또는 신원기반 암호 기법[4]에 기반하여 설계되었다. 하지만, 전통적인 공개키기반 암호 기법은 인증서 관리 문제의 어려움을 내포하고 있으며, 신원기반 암호 기법은 키 위탁문제를 가지고 있다. 위와 같은 전통적인 공개키기반 및 신원기반 암호 기법의 문제점을 해결하기 위하여, Al-Riyami와 Paterson은 전통적인 공개키기반 암호 기법의 장점 (키 위탁문제 해결) 및 신원기반 암호 기법의 장점 (묵시적인 인증)을 결합시킨 무인증서기반 공개키 암호 (CL-PKC, Certificateless Public Key Cryptography) 기술의 개념을 소개하였다[5]. 위와 같은 장점으로 인하여, 무인증서기반 암호 기법에 관한 연구가 최근까지 활발히 진행되고 있지만[6-8], 아직까지 무인증서기반 프락시 재암호화 기술은 소개되지 않았다.

본 논문에서는 프락시의 재암호화 기능을 제공하면서 전통적인 공개키 암호 기법 및 신원기반 암호 기법의 장점을 유지할 수 있는, 무인증서기반 프락시 재암호화 기법의 개념을 소개한다. 또한, 안전한 무인증서기반 프락시 재암호화 기법의 설계를 위하여, 무인증서기반 암호 기법과 프락시 재암호화 기법의 보안 요소를 고려한 정형화된 보안 모델을 정의한 후, Bilinear Pairing을 이용하여 무인증서기반 프락시 재암호화 기법을 설계한다. 제안 기법은 단반향성을 제공하는 동시에 기존의 무인증서기반 암호 환경에 적합하도록 설계되었다. 또한, 제안 기법에 대하여 랜덤 오라클 모델에서 선택 암호문 공격에 대한 안전성을 증명한다. 마지막으로, 제안 기법을 다중 키 생성 센터 (KGC, Key Generation Center) 환경에 적합한 무인증서기반 단방향 프락시 재암호화 기법으로 확장한다.

2. 무인증서기반 프락시 재암호화 기법

본 장에서는 무인증서기반 프락시 재암호화 기법 (CL-PRE, Certificateless Proxy Re-Encryption)의 정의와 안전한 기법 설계를 위하여 보안 모델을 제시한다.

2.1 정의

무인증서기반 단방향 프락시 재암호화 (CL-PRE, Certificateless Proxy Re-Encryption) 기법의 정형

화된 모델은 다음과 같다.

• ***Definition 1 (CL-PRE).*** *무인증서기반 단방향 프락시 재암호화 기법은 아래와 같이 9가지의 알고리즘으로 구성된다.*

- **Setup**(k): 보안 매개변수 k를 입력 값으로, 마스터 키 mk와 시스템 변수 $params$를 생성한다.
- **Partial-Private-Key-Extract**(mk, ID_A): 마스터 키와 사용자 A의 신원정보 ID_A를 입력 값으로, 사용자 A에 대한 부분 개인키 d_A를 생성한다.
- **Set-Secret-Key**($params$): 시스템 변수를 입력 값으로, 사용자 A에 대한 임의의 비밀 값 x_A를 생성한다.
- **Set-Private-Key**($params$, d_A, x_A): 시스템 변수, 사용자 A의 부분 개인키와 비밀 값을 입력 값으로, 사용자 A에 대한 개인키 sk_A를 생성한다.
- **Set-Public-Key**($params$, x_A): 시스템 변수와 사용자 A의 비밀 값을 입력 값으로, 사용자 A의 공개키 pk_A를 생성한다.
- **Encrypt**(m, $params$, ID_A, pk_A): 메시지 m, 시스템 변수, 사용자 A의 신원정보와 공개키를 입력 값으로, m에 대한 암호문 C_A를 생성하거나 $\perp$을 출력한다.
- **Set-Proxy-Re-Encryption-Key**($params$, ID_A, pk_A, sk_A, ID_B, pk_B): 시스템 변수, 사용자 A의 신원정보 및 공개키/개인키 쌍과 사용자 B의 신원정보와 공개키를 입력 값으로, 단방향 재암호화 키 $rk_{A\rightarrow B}$를 생성한다.
- **Re-Encrypt**($params$, $rk_{A\rightarrow B}$, C_A): 시스템 변수, 재암호화 키와 사용자 A의 공개키로 암호화된 암호문을 입력 값으로, 사용자 B를 위해 재암호화된 암호문 C_B를 생성하거나 $\perp$을 출력한다.
- **Decrypt**($params$, sk_{ID}, C_{ID}): 시스템 변수, 사용자 ID에 대한 암호문과 개인키를 입력 값으로, 메시지 m을 출력하거나 $\perp$을 출력한다.

제안 모델의 완전성 (Completeness)으로서, 아래와 같은 두 가지 경우에만 암호문에 대한 정확한 메시지 m이 출력된다.

- **Decrypt**($params$, sk_A, **Encrypt**(m, $params$, ID_A, pk_A)) = m
- **Decrypt**($params$, sk_B, **Re-Encrypt**($params$, $rk_{A\rightarrow B}$, C_A) = m

여기서,

sk_{ID}←Set-Private-Key($params$, d_{ID}, x_{ID}),

$rk_{A\rightarrow B}$←Set-Proxy-Re-Encryption-Key ($params$, ID_A, pk_A, sk_A, ID_B, pk_B)이다.

2.2 보안 모델

본 절에서는 제안 기법에 대하여 선택 암호문 공격에 대한 안전성을 증명하기 위한 무인증서기반 프락시 재암호화 기법의 보안 모델을 소개한다. 제안 기법에 대하여 명확한 보안 개념을 확립하기 위하여, 기 제안된 무인증서기반 암호 기법[5,6] 및 프락시 재암호화 기법[3,4]의 보안 개념을 고려하였다. [5,6]의 보안 모델에 따라, 제안 기법의 보안 모델에서도 공격자는 타입 I (A_I)과 타입 II (A_{II}) 공격자로 나눠지며 각각 악의적인 사용자의 능동적 공격과 키 생성 센터 (KGC, Key Generation Center)의 도청 공격을 의미한다. 추가적으로, 도전 (Challenge)하는 신원정보 ID^* 및 암호문 C^*에 대하여 재암호화 질의 및 재암호화 키 질의를 통한 공격자의 공격을 예방하기 위하여, 제안하는 보안 모델에 Challenge derivative의 개념[3,4]을 추가하였으며, 선택 암호문 공격에서 선택-ID (sID, selective-ID) 모델[8-10]을 적용하였다. 따라서, 공격자는 공격을 시작하기 전에 공격하고자 하는 사용자의 신원정보 ID^*를 지정하게 된다. 하지만, 공격자가 선택한 암호문 및 신원정보에 대한 질의는 적응적 (Adaptively)으로 허용한다.

지금부터 두 공격자 A_I와 A_{II}에 대한 선택-ID, 선택 암호문 공격 (IND-sID-CCA) 게임을 정의한다. 공격자 A_I와 도전자 (Challenger)에 대한 IND-sID-CCA 게임의 정의는 다음과 같다.

• **Init**: A_I는 도전하고자 하는 사용자의 신원정보 ID^*를 출력한다.

• **Setup**: 도전자는 보안 매개변수 k를 이용하여 **Setup** 알고리즘을 수행한 후, A_I에게 시스템 변수 $params$를 전달하고 마스터 키 mk는 안전하게 보관한다.

• **Phase 1**: A_I는 아래와 같은 질의들 $q_1, \cdots, q_m$을 생성하고 질의에 대한 응답을 도전자에게 요청한다.

- **Extraction query** on $ID_i \neq ID^*$: 도전자는 **Partial-Private-Key-Extract** 알고리즘을 수행하여 ID_i에 대한 부분 개인키 d_{ID_i}를 생성한 후 A_I에게 전송한다.
- **Private Key query** on $ID_i \neq ID^*$: 만약 ID_i의 공개키가 대체되지 않았다면, 도전자는 **Set-Private-Key** 알고리즘을 수행하여 ID_i에 대한 개인키 sk_{ID_i}를 생성한 후 A_I에게 전송한다.
- **Public Key query** on ID_i: 도전자는 **Set-Public-Key** 알고리즘을 수행하여 ID_i의 공개키를 생성한 후 A_I에게 전송한다.
- **Replace Public Key query** on the public key for ID_i: 도전자는 A_I가 임의로 선택한 공개키 pk'_{ID_i}로 ID_i의 공개키를 대체한다.
- **Re-Encryption Key query** on $(ID_i \neq ID^*, ID_j)$: 도전자는 **Set-Proxy-Re-Encryption-Key** 알고리즘을 수행하여 재암호화 키 $rk_{i\rightarrow j}$를 생성한 후 A_I에게 전송한다.
- **Re-Encryption query** on $(ID_i, ID_j, C_i, rk_{i\rightarrow j})$: A_I가 선택한 신원정보 (ID_i, ID_j), 암호문 C_i과 재암호화 키 $rk_{i\rightarrow j}$에 대하여, 도전자는 **Re-Encrypt** 알고리즘을 수행하여 C_i를 C_j로 변형한 후 A_I에게 전송한다.
- **Decryption query** on (ID_i, C_i): A_I가 선택한 신원정보 ID_i와 암호문 C_i에 대하여, 도전자는 개인키 sk_i를 이용하여 **Decrypt** 알고리즘을 수행한 후 출력된 결과를 A_I에게 전송한다.

• **Challenge**: A_I는 **Phase 1** 단계를 종료한 후, 동일한 길이의 평문 $m_0, m_1 \in M$을 생성하여 도전자에게 전송한다. 도전자는 임의의 한 비트 $b \in \{0,1\}$을 선택하고 도전 암호문 C^*=**Encrypt**$(m_b, params, ID^*, pk_{ID^*})$을 생성한 후 A_I에게 전송한다.

• **Phase 2**: A_I는 아래와 같은 질의들 $q_{m+1}, \cdots, q_n$을 추가적으로 생성하고 질의에 대한 응답을 도전자에게 요청한다.

- **Decryption query** on (ID_i, C_i): 만약 (ID_i, C_i)가 Challenge derivative에 포함되지 않으면, 도전자는 **Phase 1**과 동일하게 수행한다. Challenge derivative의 정의는 다음과 같다 [3,4].

***Challenge derivative* 정의**

1. (ID^*, C^*)는 *Challenge derivative*이다.
2. (ID_i, C_i)가 *Challenge derivative*에 포함되고 A_I가 C_i와 (ID_i, ID_j)에 대한 **Re-Encryption query**를 통하여 C_j을 획득하였다면, (ID_j, C_j)도 *Challenge derivative*에 포함된다.
3. (ID_i, C_i)가 *Challenge derivative*에 포함되고 A_I가 (ID_i, ID_j)에 대한 **Re-Encryption Key query**를 통하여 $rk_{i\rightarrow j}$와 C_j=**Re-Encrypt**$(params, rk_{i\rightarrow j}, C_i)$를 획득하였다면, (ID_j, C_j)도 *Challenge derivative*에 포함된다.

- **Extraction query** on $ID_i \neq ID^*$: (ID_i, C_i)가 *Challenge derivative*에 포함되지 않으면, 도전자는 **Phase 1**과 동일하게 수행한다.
- **Private Key query** on $ID_i \neq ID^*$: (ID_i, C_i)가 *Challenge derivative*에 포함되지 않고 ID_i의 공개키가 대체되지 않았으면, 도전자는 **Phase 1**과 동일하게 수행한다.
- **Public Key query** on ID_i: 도전자는 **Phase 1**과 동일하게 수행한다.
- **Replace Public Key query** on public key for ID_i: 도전자는 **Phase 1**과 동일하게 수행한다.
- **Re-Encryption Key query** on $(ID_i \neq ID^*, ID_j)$: A_I가 ID_i에 대한 **Private Key query** 질의를 생성하였고 (ID_i, C_i)가 *Challenge derivative*에 포함되는 경우를 제외하고, 도전자는 **Phase 1**과 동일하게 수행한다.
- **Re-Encryption query** $(ID_i, ID_j, C_i, rk_{i\rightarrow j})$: A_I가 ID_j에 대한 **Private Key query** 질의를 요청하였고 (ID_i, C_i)가 *Challenge derivative*에 포함되는 경우를 제외하고, 도전자는 **Phase 1**과 동일하게 수행한다.

• **Guess**: 마지막으로, A_I는 $b' \in \{0,1\}$을 출력하고, 만약 $b'=b$이면 A_I는 공격에 성공한다.

무인증서기반 프락시 재암호화 기법을 공격자 A_I가 손상시킬 수 있는 이점 (Advantage)은 다음과 같이 정의한다.

$$Adv(A_I)=\left|\Pr[b=b']-\frac{1}{2}\right|$$

• ***Definition 2.*** *만약 임의의 t-시간 동안 공격자 A_I에 대하여 $Adv(A_I)$가 negligible한 확률 ϵ을 가진다면 무인증서기반 단방향 프락시 재암호화 기법은 타입 I 공격자에 대하여 (t,ϵ)-선택-ID, 적응적 선택 암호문 공격에 안전하다고 정의한다.*

A_{II}와 도전자에 대한 선택-ID, 선택 암호문 공격 (IND-sID-CCA) 게임의 정의는 다음과 같다.

• **Init**: A_{II}는 도전하고자 하는 사용자의 신원정보 ID^*를 출력한다.

• **Setup**: 도전자는 보안 매개변수 k를 이용하여 **Setup** 알고리즘을 수행한 후, A_{II}에게 시스템 변수 $params$와 마스터 키 mk를 전송한다.

• **Phase 1**: A_{II}는 아래와 같은 질의들 $q_1,\cdots,q_m$을 생성하고 질의에 대한 응답을 도전자에게 요청한다.

- **Private Key query** on $ID_i \neq ID^*$: 도전자는 **Set-Private-Key** 알고리즘을 수행하여 ID_i에 대한 개인키 sk_{ID_i}를 생성한 후 A_{II}에게 전송한다.
- **Public Key query** on ID_i: 도전자는 **Set-Public-Key** 알고리즘을 수행하여 ID_i의 공개키 pk_{ID_i}를 생성한 후 A_{II}에게 전송한다.
- **Re-Encryption Key query** on $(ID_i \neq ID^*, ID_j)$: 도전자는 **Set-Proxy-Re-Encryption-Key** 알고리즘을 수행하여 재암호화 키 $rk_{i\rightarrow j}$를 생성한 후 A_{II}에게 전송한다.
- **Re-Encryption query** on $(ID_i, ID_j, C_i, rk_{i\rightarrow j})$: A_{II}가 선택한 신원정보 (ID_i, ID_j), 암호문 C_i과 재암호화 키 $rk_{i\rightarrow j}$에 대하여, 도전자는 **Re-Encrypt** 알고리즘을 수행하여 C_i를 C_j로 변형한 후 A_{II}에게 전송한다.
- **Decryption query** on (ID_i, C_i): A_I가 선택한 신원정보 ID_i와 암호문 C_i에 대하여, 도전자는 개인키 sk_i를 이용하여 **Decrypt** 알고리즘을 수행한 후 출력된 결과를 A_{II}에게 전송한다.

• **Challenge**: A_{II}는 **Phase 1** 단계를 종료한 후, 동일한 길이의 평문 $m_0, m_1 \in M$을 생성하여 도전자에게 전송한다. 도전자는 임의의 한 비트 $b \in \{0,1\}$을 선택하고, 도전암호문 C^*=**Encrypt**$(m_b, params, ID^*, pk_{ID^*})$을 생성하여 A_{II}에게 전송한다.

• **Phase 2**: A_{II}는 아래와 같은 질의들 $q_{m+1},\cdots,q_n$을 추가적으로 생성하고 질의에 대한 응답을 도전자에게 요청한다.

- **Decryption query** on (ID_i, C_i): 만약 (ID_i, C_i)가 *Challenge derivative*에 포함되지 않으면, 도전자는 **Phase 1**과 동일하게 수행한다.
- **Private Key query** on $ID_i \neq ID^*$: 만약 (ID_i, C_i)가 *Challenge derivative*에 포함되지 않으면, 도전자는 **Phase 1**과 동일하게 수행한다.
- **Public Key query** on ID_i: 도전자는 **Phase 1**과 동일하게 수행한다.
- **Re-Encryption Key query** on $(ID_i \neq ID^*, ID_j)$: A_{II}가 ID_j에 대하여 **Private Key query** 질의를 생성하였고, (ID_i, C_i)가 *Challenge derivative*에 포함되는 경우를 제외하고, 도전자는 **Phase 1**과 동일하게 수행한다.
- **Re-Encryption query** on $(ID_i, ID_j, C_i, rk_{i\rightarrow j})$: A_{II}가 ID_j에 대한 **Private Key query** 질의를 생성하였고 (ID_i, C_i)가 *Challenge derivative*에 포함되는 경우를 제외하고, 도전자는 **Phase 1**과 동일하게 수행한다.

• **Guess**: 마지막으로, A_{II}는 $b' \in \{0,1\}$을 출력하고, 만약 $b'=b$이면 A_{II}는 공격에 성공한다.

무인증서기반 프락시 재암호화 기법을 공격자 A_{II}가 손상시킬 수 있는 이점은 다음과 같이 정의한다.

$$Adv(A_{II})=\left|\Pr[b=b']-\frac{1}{2}\right|$$

• ***Definition 3.*** *만약 임의의 t-시간 동안 공격자 A_{II}에 대하여 $Adv(A_{II})$가 negligible한 확률 ϵ을 가진다면 무인증서기반 단방향 프락시 재암호화 기법은 타입 II 공격자에 대하여 (t,ϵ)-선택-ID, 적응적 선택 암호문 공격에 안전하다고 정의한다.*

선택-ID, 선택 평문 공격 (IND-sID-CPA)에 대한 보안 모델은 IND-sID-CCA의 보안 모델에서 *Decryption* 질의를 허용하지 않을 뿐, 그 밖에 질의 및 단계는 IND-sID-CCA의 보안 모델과 동일하다.

3. Bilinear Pairing을 이용한 무인증서기반 프락시 재암호화 기법

본 장에서는 Bilinear Pairing을 이용하여 신원기반 프락시 재암호화 기법에서의 키 위탁 문제와 전통적인 공개키기반 프락시 재암호화 기법의 인증서 관리 문제를 해결한 무인증서기반 프락시 재암호화 기법을 제안하고, 제안 기법에 대하여 랜덤 오라클 모델에서 선택-ID, 선택 암호문 공격 (IND-sID-CCA)에 대한 안전성을 증명한다.

3.1 Bilinear Pairing 및 Complexity Assumptions

본 절에서는 무인증서기반 프락시 재암호화 기법을 설계하기 위한 기반기술로서 Bilinear Pairing 및 관련 문제들을 소개한다. 제안 기법에 적용할 Bilinear Pairing을 다음과 같이 설정한다[9,11].

1. G_1과 G_2는 차수 (order)가 소수 q인 곱셈 순환군 (multiplicative cyclic groups)이다.
2. g는 G_1의 생성자 (generator)이다.
3. $e: G_1 \times G_1 \rightarrow G_2$를 Bilinear Pairing이라 한다.

위와 같은 Bilinear Pairing은 아래와 같은 두 가지 특성을 가진다.

1. Bilinear: 임의의 $u, v \in G_1$과 $a, b \in Z_q^*$에 대하여 식 $e(u^a, v^b) = e(u, v)^{ab}$을 만족한다.
2. Non-degenerate: $e(g, g) \neq 1_{G_2}$

또한, 식 $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$을 만족하기 때문에, $e(,)$은 대칭성 (Symmetric)을 가진다.

제안 기법과 관련된 Bilinear Pairing기반의 어려운 문제들은 다음과 같다.

• ***Definition 4 (DBDH).*** *Decision Bilinear Diffie Hellman (DBDH) 문제는 $(g, g^a, g^b, g^c, T) \in G_1^4 \times G_2$이 주어졌을 때, 식 $T = e(g, g)^{abc}$를 결정하는 문제이다. 만약 $b \in \{0,1\}$를 출력하는 알고리즘 B가 아래 식을 만족한다면 알고리즘 B는 DBDH 문제를 해결하기 위해 ϵ만큼의 이점을 가진다.*

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

• ***Definition 5 (p-DBDHI).*** *Decision p-Bilinear Diffie Hellman Inversion (p-DBDHI) 문제는 $(g, g^\alpha, \cdots, g^{\alpha^p}, T) \in G_1^{p+1} \times G_2$이 주어졌을 때, 식 $T = e(g, g)^{1/\alpha}$를 결정하는 문제이다. 만약 $b \in \{0,1\}$를 출력하는 알고리즘 B가 아래 식을 만족한다면 알고리즘 B는 p-DBDHI 문제를 해결하기 위해 ϵ만큼의 이점을 가진다.*

$$\left| \begin{array}{l} \Pr[B(g, g^\alpha, \cdots, g^{\alpha^p}, e(g,g)^{1/\alpha}) = 0] \\ -\Pr[B(g, g^\alpha, \cdots, g^{\alpha^p}, T) = 0] \end{array} \right| \geq \epsilon$$

3.2 제안 기법

제안하는 무인증서기반 단방향 프락시 재암호화 기법은 아래와 같은 9가지의 알고리즘으로 구성되어 있으며, 각 알고리즘의 자세한 설명은 다음과 같다.

• **Setup**: 보안 매개변수 k를 입력 값으로, 키 생성 센터 (KGC)는 아래와 같은 절차를 수행한다.

1. k-bit인 소수 q를 선택하고, 위수 q를 갖는 Bilinear map group (G_1, G_2)과 G_1의 생성자 g를 선택한다.
2. KGC의 마스터 키로 임의의 $\alpha \in Z_q^*$를 선택하고, 공개키 $g_1 = g^\alpha \in G_1$를 계산한다.
3. 암호학적 해쉬 함수 $H_1: \{0,1\}^* \rightarrow G_1$, $H_2: \{0,1\}^* \rightarrow Z_q^*$, $H_3: \{0,1\}^* \rightarrow Z_q^*$, $H_4: G_2 \rightarrow \{0,1\}^n$, $H_5: \{0,1\}^* \rightarrow G_1$, $H_6: \{0,1\}^* \rightarrow G_1$, $H_7: \{0,1\}^* \rightarrow G_1$를 선택하고, 그룹 원소 $g_2 = e(g,g) \in G_2$를 계산한다.

시스템 변수는 $params = \{q, G_1, G_2, e, g, g_1, g_2, H_1, H_2, H_3, H_4, H_5, H_6, H_7\}$으로 구성되며, 메시지 공간은 $M := \{0,1\}^n$이다.

• **Partial-Private-Key-Extract**: 사용자 A의 신원정보 ID_A를 입력 값으로, A의 부분 개인키 $d_A = H_1(ID_A)^\alpha \in G_1$를 출력한다.

• **Set-Secret-Value**: 시스템 변수 $params$와 사용자 A의 신원정보 ID_A를 입력 값으로, 임의의 $x_A \in Z_q^*$를 선택하고 사용자 A의 비밀 값으로 출력한다.

• **Set-Private-Key**: 시스템 변수 $params$, 사용자 A의 부분 개인키 d_A와 비밀 값 x_A를 입력 값으로, A의 개인키 $sk_A = (d_A, x_A)$를 출력한다.

• **Set-Public-Key**: 시스템 변수 $params$와 사용자 A의 비밀 값 x_A를 입력 값으로, 사용자 A의 공개키 $pk_A = g^{x_A} \in G_1$를 생성한다.

• **Encrypt**: 사용자 A의 신원정보 ID_A와 공개키 pk_A를 이용하여 메시지 $m \in M$에 대한 암호문 C를 아래와 같은 단계를 통하여 생성한다.

1. pk_A가 G_1의 원소인지 검사한 후, 만약 G_1의 원소가 아닌 경우에 $\perp$을 출력한다.
2. 임의의 $\sigma \in G_2$를 선택한다.
3. $r_1 = H_2(m\|\sigma\|ID_A\|pk_A)$와 $r_2 = H_3(m\|\sigma\|ID_A\|pk_A)$를 설정한다.
4. 아래와 같이 암호문 $C=(C_1, C_2, C_3, C_4, C_5, C_6)$를 생성한다.

$$C=(g^{r_1},\ pk_A^{r_2},\ \sigma \cdot e(H_1(ID_A), g_1)^{r_1} \cdot g_2^{r_2},\ m \oplus H_4(\sigma),\ u^{r_1},\ v^{r_2})$$

여기서,

$u = H_6(ID_A\|pk_A\|C_1\|C_2\|C_3\|C_4)$,

$v = H_7(ID_A\|pk_A\|C_1\|C_2\|C_3\|C_4)$이다.

• **Set-Proxy-Re-Encryption-Key**: 사용자 B의 공개키 pk_B와 사용자 A의 공개키 pk_A 및 개인키 sk_A를 입력 값으로, 다음과 같은 절차를 통하여 재암호화 키 $rk_{A\to B}$를 생성한다.

1. $\kappa = e(d_A, H_1(ID_B))$와 $\mu = H_5(\kappa\|ID_A\|pk_A\|ID_B\|pk_B)$를 계산한다.
2. 재암호화 키 $rk_{A\to B} = (\mu \cdot d_A, g^{x_B/x_A})$를 설정한다.

• **Re-Encrypt**: 재암호화 키 $rk_{A\to B}$와 사용자 A에 대한 암호문 C_A를 입력 값으로, 프락시는 아래와 같이 재암호화를 수행한 후, 사용자 B에 대한 암호문 C'을 출력한다.

1. $u' = H_6(ID_A\|pk_A\|C_1\|C_2\|C_3\|C_4)$와 $v' = H_7(ID_A\|pk_A\|C_1\|C_2\|C_3\|C_4)$ 설정한다.
2. 식 $e(g, C_5) \cdot e(pk_A, C_6) = e(u', C_1) \cdot e(v', C_2)$을 만족하지 않으면, $\perp$를 출력한다.
3. $C_2' = e(C_2, g^{x_B/x_A})$와 $C_3' = C_3/e(C_1, \mu \cdot d_A)$를 계산한다.
4. 새로운 암호문 $C' = (C_1, C_2', C_3', C_4, ID_A, pk_A)$를 출력한다.

• **Decrypt**:

- 재암호화되지 않은 암호문 $C=(C_1, C_2, C_3, C_4, C_5, C_6)$의 경우, 사용자 A는 아래와 같이 복호화 과정을 수행한다.
  1. $\sigma' = C_3/(e(d_A, C_1) \cdot e(g, C_2)^{1/x_A})$를 계산한다.
  2. $m' = C_4 \oplus H_4(\sigma')$를 계산한 후, $r'_1 = H_2(m'\|\sigma'\|ID_A\|pk_A)$와 $r'_2 = H_3(m'\|\sigma'\|ID_A\|pk_A)$를 설정한다.
  3. 만약 식 $C_1 = g^{r'_1}$와 식 $C_2 = pk_A^{r'_2}$을 만족하면, m'을 메시지로 출력하고 식을 만족하지 않은 경우에는 $\perp$를 출력한다.
- 재암호문 $C' = (C_1, C_2', C_3', C_4, ID_A, pk_A)$의 경우, 사용자 B는 아래와 같이 복호화 알고리즘을 수행한다.
  1. $\kappa' = e(H_1(ID_A), d_B)$와 $\mu' = H_5(\kappa'\|ID_A\|pk_A\|ID_B\|pk_B)$를 계산한다.
  2. $\sigma' = C_3' \cdot e(C_1, \mu')/C_2'^{1/x_B}$를 계산한다.
  3. $m' = C_4 \oplus H_4(\sigma')$를 계산한 후, $r'_1 = H_2(m'\|\sigma'\|ID_A\|pk_A)$와 $r'_2 = H_3(m'\|\sigma'\|ID_A\|pk_A)$를 설정한다.
  4. 만약 식 $C_1 = g^{r'_1}$와 식 $C_2 = e(g, pk_B)^{r'_2}$을 만족하면, m'을 메시지로 출력하고 식을 만족하지 않은 경우에는 $\perp$를 출력한다.

제안 기법의 일치성은 다음과 같다.

- 재암호문이 아닌 경우, 사용자 A에 대하여

$$C_3/(e(d_A, C_1) \cdot e(g, C_2)^{1/x_A}) = \frac{\sigma \cdot e(H_1(ID_A), g^{\alpha})^{r_1} \cdot g_2^{r_2}}{e(H_1(ID_A)^{\alpha}, g^{r_1}) \cdot e(g, g^{x_A r_2})^{1/x_A}} = \frac{\sigma \cdot e(H_1(ID_A), g)^{\alpha r_1} \cdot g_2^{r_2}}{e(H_1(ID_A), g)^{\alpha r_1} \cdot g_2^{r_2}} = \sigma$$

- 재암호문의 경우, 사용자 B에 대하여

$$C_3' \cdot e(C_1, \mu')/C_2'^{1/x_B} = \frac{\sigma \cdot g_2^{r_2} \cdot e(g^{r_1}, \mu)}{e(g^{r_1}, \mu) \cdot g_2^{x_B r_2/x_B}} = \frac{\sigma \cdot g_2^{r_2} \cdot e(g, \mu)^{r_1}}{e(g, \mu)^{r_1} \cdot g_2^{r_2}} = \sigma$$

3.3 안전성

두 공격자 A_I와 A_{II}에 대하여 제안 기법의 안전성은 3.1절에서 소개한 **DBDH** 문제 및 *p*-**DBDHI** 가정과 랜덤 오라클 모델 하에서 아래와 같이 증명된다.

• ***Theorem 1.*** *3.2절에서 제안한 무인증서기반 단방향 프락시 재암호화 기법에 대하여, IND-sID-CCA 게임에서 공격자 A_I가 τ만큼의 실행 시간동안 ϵ만큼의 이점을 가진다고 가정하자. 그렇다면 제안 기법은 DBDH 가정과 랜덤 오라클 모델 하에서 안전하다.*

• **Theorem 2.** *3.2절에서 제안한 무인증서기반 단방향 프락시 재암호화 기법에 대하여, IND-sID-CCA 게임에서 공격자 A_{II}가 τ만큼의 실행 시간동안 ϵ만큼의 이점을 가진다고 가정하자. 그렇다면, 제안 기법은 p-DBDHI 가정과 랜덤 오라클 모델 하에서 안전하다.*

논문 매수 규정으로 인하여 Theorem 1과 Theorem 2에 대한 증명은 생략한다.

4. 다중 KGC 환경에 적합한 무인증서기반 프락시 재암호화 기법으로의 확장

무인증서기반 암호 기법은 전통적인 공개키 암호 기법보다는 소규모의 환경에 적합하지만 키 위탁문제를 내포하고 있는 신원기반 암호 기법보다 광범위한 환경에서 사용 가능하다. 이러한 무인증서기반 암호 기법의 특성으로 인하여, 서로 다른 KGC에 속한 사용자들 간의 안전하고 효율적인 무인증서기반 프락시 재암호화 기법에 대한 연구가 필요하다. 따라서, 본 장에서는 3장에서 소개한 무인증서기반 단방향 프락시 재암호화 기법을 다중 KGC 환경으로 확장하고자 한다.

제안 기법을 다중 KGC 환경으로 확장하기 위해 아래와 같은 환경을 가정한다.

• 두 신뢰기관 (KGC1, KGC2)은 서로 다른 마스터 키/공개키 쌍을 가지고 있다.

- KGC1: $(g^{\alpha_1} \in G_1,\ \alpha_1 \in Z_q^*)$
- KGC2: $(g^{\alpha_2} \in G_1,\ \alpha_2 \in Z_q^*)$

• 신뢰기관들은 마스터 키/공개키 쌍을 제외하고 동일한 시스템 변수를 가진다.

- $params = \{q, G_1, G_2, e, g, g_2, H_1, H_2, H_3, H_4, H_5, H_6, H_7\}$

• KGC1에 등록된 사용자 A는 부분 개인키 $d_A = H_1(ID_A)^{\alpha_1} \in G_1$를 KGC1을 통하여 획득하고 비밀 값 $x_A \in Z_q^*$을 선택한 후, 공개키 $pk_A = g^{x_A} \in G_1$를 생성하고 개인키로 $sk_A = (d_A, x_A)$를 설정한다.

• KGC2에 등록된 사용자 B는 부분 개인키 $d_B = H_1(ID_B)^{\alpha_2} \in G_1$를 KGC2을 통하여 획득하고 비밀 값 $x_B \in Z_q^*$을 선택한 후, 공개키 $pk_B = g^{x_B} \in G_1$를 생성하고 개인키로 $sk_B = (d_B, x_B)$를 설정한다.

3장에서 제안한 기법을 다중 KGC 환경에 적합한 기법으로 확장하기 위하여 Chen과 Kudla가 제안한 키 설정 기법 [12]을 사용하여 아래와 같이 **Set-Proxy-Re-Encryption-Key** 알고리즘과 **Decrypt** 알고리즘을 수정하여, 다중 KGC 환경에 적합한 무인증서기반 단방향 프락시 재암호화 기법을 설계한다.

• **Set-Proxy-Re-Encryption-Key**: 사용자 B의 공개키 pk_B와 사용자 A의 공개키 pk_A 및 개인키 sk_A를 입력 값으로, 다음과 같은 절차를 수행하여 재암호화 키 $rk_{A\to B}$를 생성한다.

1. $\kappa = e(d_A, pk_B) \cdot e(H_1(ID_B), (g^{\alpha_2})^{x_A})$와 $\mu = H_5(\kappa \| ID_A \| pk_A \| ID_B \| pk_B)$를 계산한다.
2. 재암호화 키 $rk_{A\to B} = (\mu \cdot d_A,\ g^{x_B/x_A})$를 설정한다.

• **Decrypt**:

- 재암호문 $C' = < C_1, C_2', C_3', C_4, ID_A, pk_A >$의 경우 사용자 B는 아래와 같이 복호화를 수행한다.

1. $\kappa' = e(d_B, pk_A) \cdot e(H_1(ID_A), (g^{\alpha_1})^{x_B})$와 $\mu' = H_5(\kappa' \| ID_A \| pk_A \| ID_B \| pk_B)$를 계산한다.
2. $\sigma' = C_3' \cdot e(C_1, \mu') / C_2'^{\,1/x_B}$를 계산한다.
3. $m' = C_4 \oplus H_4(\sigma')$를 계산한 후, $r'_1 = H_2(m' \| \sigma' \| ID_A \| pk_A)$와 $r'_2 = H_2(m' \| \sigma' \| ID_A \| pk_A)$를 설정한다.
4. 만약 식 $C_1 = g^{r'_1}$와 식 $C_2 = e(g, pk_B)^{r'_2}$을 만족하면, m'을 메시지로 출력하고 그렇지 않을 경우에는 $\perp$를 출력한다.

제안 기법의 일치성은 다음과 같다.

$$
\begin{aligned}
- \ \kappa &= e(d_A, pk_B) \cdot e(H_1(ID_B), (g^{\alpha_2})^{x_A}) \\
&= e(H_1(ID_A)^{\alpha_1}, g^{x_B}) \cdot e(H_1(ID_B)^{\alpha_2}, g^{x_A}) \\
&= e(H_1(ID_A), (g^{\alpha_1})^{x_B}) \cdot e(d_B, pk_A) \\
&= \kappa'
\end{aligned}
$$

5. 결 론

본 논문에서는 무인증서기반 프락시 재암호화 기법의 개념을 소개하고 안전한 무인증서기반 프락시 재암호화 기법을 설계하기 위한 정형화된 모델을 제안하였다. 또한, Bilinear Pairing을 이용하여 무인증서기반 암호 기법과 프락시 재암호화 기법의 특성을 융합하는 무인증서기반 단방향 프락시 재암호화 기법을 설계하였다. 또한, 제안 기법을 Bilinear Pairing과 관련된 어려운 문제들을 기반하여 랜덤 오라클 모델 하에서 선택-ID, 선택 암호문 공격에 대한 안전성을 증명하였다. 마지막으로, 제안 기법의 보다 광범위한 활용을 위하여 다중 KGC 환경에 적합한 무인증서기반 단방향 프락시 재암호화 기법으로의 확장 방법을 제안하였다.

참 고 문 헌

[1] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," *Advances in Cryptology - Eurocrypt' 98*, LNCS 1403, pp. 127-144, 1998.

[2] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," *Cryptography ePrint Archive*, Report 2005/028, 2005.

[3] R. Canetti and S. Hohenberger, "Chosen-ciphertext secure proxy re-encryption," *Cryptography ePrint Archive*, Report 2007/171, 2007.

[4] M. Green and G. Ateniese, "Identity-based proxy re-encryption," *Cryptography ePrint Archive*, Report 2006/473, 2006.

[5] S. S. Al-Riyami and K. Paterson, "Certificateless public key cryptography," *Advances in Cryptology - Asiacrypt 2003*, LNCS 2894, pp. 452-473, 2003.

[6] S. S. Al-Riyami and K. Paterson, "CBE from CL-PKE: A generic construction and efficient scheme," *Public Key Cryptography - PKC 2005*, LNCS 3386, pp. 398-415, 2005.

[7] B. Libert and J. Quisquater, "On constructing certificateless cryptosystem from identity based encryption," *Public Key Cryptography - PKC 2006*, LNCS 3958, pp. 474-490, 2006.

[8] J. H. Park, K. Y. Choi, J. Y. Hwang, and D. H. Lee, "Certificateless public key encryption in the selective-id security model," *Pairing 2007*, LNCS 4575, pp. 60-82, 2007.

[9] D. Boneh and X. Boyen, "Efficient selective-id secure identity based encryption without random oracles," *Advances in Cryptology - Eurocrypt 2004*, LNCS 3027, pp. 223-238, 2004.

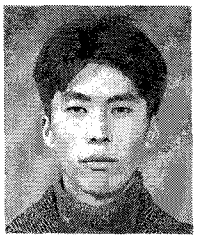
[10] R. Canetti, S. Halevi, and J. Katz, "A forward-secure public-key encryption scheme," *Advances in Cryptology - Eurocrypt 2003*, LNCS 2656, pp. 255-271, 2003.

[11] D. Boneh and M. Franklin, "Identity-based encryption from the weil paring," *Advances in Cryptology - Crypto 2001*, LNCS 2139, pp. 213-229, 2001.

[12] L. Chen and C. Kudla, "Identity based authenticated key agreement protocols from pairings," *Cryptography ePrint Archive*, Report 2002/184, 2002.

[13] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," *ACM CCS' 93*, pp. 62-73, 1993.

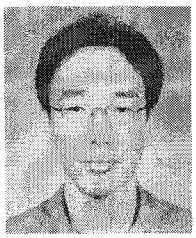
[14] E. Fujisaki and T. Okamoto, "Secure integration of asymmetric and symmetric encryption schemes," *Advances in Cryptology - Crypto' 99*, LNCS 1666, pp. 535-554, 1999.



서 철

2000년 부경대학교 전자계산학과 학사
2004년 부경대학교 전자계산학과 석사
2004년~현재 부경대학교 전자계산학과 박사과정

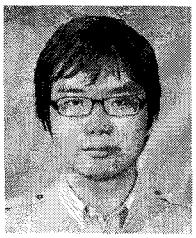
관심분야 : 암호 프로토콜, 공개키 암호, 신원기반 암호



박 영 호

2000년 부경대학교 전자계산학과 학사
2002년 부경대학교 전자계산학과 석사
2006년 부경대학교 정보보호학과 박사

관심분야 : 암호 프로토콜, 공개키 암호, 신원기반 암호



정 채 덕

2005년 동의대학교 수학과 학사
2007년 부경대학교 정보보호학과 석사
2007년~현재 부경대학교 정보보호학과 박사과정

관심분야 : 암호 프로토콜, 공개키 암호, 신원기반 암호

이 경 현

1982년 경북대학교 수학교육과 학사
1985년 한국과학기술원 응용수학과 석사
1992년 한국과학기술원 수학과 박사
1993년~현재 부경대학교 전자컴퓨터정보통신공학부 교수

관심분야 : 정보보호론, 공개키 암호, 신원기반 암호, 멀티미디어 정보보호, 그룹 키 관리