情報保護學會論文誌
第 21 卷 第 4 號, 2011. 8

검증 능력이 제한된 검색 가능한 공개키 암호시스템*

엄 지 은[1†], 이 현 숙[2], 이 동 훈[1‡]
[1]고려대학교 정보보호대학원, [2]삼성전자

Public Key Encryption with Keyword Search for Restricted Testability*

Ji Eun Eom[1†], Hyun Sook Rhee[2], Dong Hoon Lee[1‡]
[1]Graduate School of Information Security, Korea University, [2]Samsung Electronics

요 약

공개키 기반의 키워드검색 시스템 (PEKS)은 암호화되어 저장된 데이터에 대한 효율적인 키워드 검색을 위해 Boneh 등에 의해 처음으로 제안되었다. 송신자는 메일내용과 키워드를 각각 수신자의 공개키로 암호화하여 서버에 전송하고, 수신자는 자신의 개인키로 키워드에 대한 트랩도어를 생성하여 키워드를 포함하는 메일을 검색할 수 있는 기법이다. 그러나 Byun 등은 PEKS 기법과 PEKS를 기반으로 한 몇 가지 기법들이 오프라인에서 키워드 추측 공격(keyword guessing attack)이 가능하다는 것을 보였다. 본 논문에서는 키워드 추측공격에 대한 안전성을 제공하는 검증 능력이 제한된 검색 가능한 공개키 암호시스템(Public Key Encryption with Keyword Search for Restricted Testability, PEKS-RT)을 제안한다.

ABSTRACT

To provide efficient keyword search on encrypted data, a public key encryption with keyword search (PEKS) was proposed by Boneh et al. A sender encrypts an e-mail and keywords with receiver's public key, respectively and uploads them on a server. Then a receiver generates a trapdoor of w with his secret key to search an e-mail related with some keyword w. However, Byun et al. showed that PEKS and some related schemes are not secure against keyword guessing attacks. In this paper, we propose a public key encryption with keyword search for restricted testability (PEKS-RT) scheme and show that our scheme is secure against keyword guessing attacks.

Keywords: Keyword search on encrypted data, Database security and privacy, Keyword guessing attack, Searchable encryption

I. 서 론

스마트 폰을 비롯한 통신기기의 발달로 인해 인터넷에 대한 접근이 용이해지면서 저장, 전송되는 정보의 양이 급격히 증가하고 있다. 특히 개인의 민감한 정보를 포함한 디지털 정보들은 개인의 컴퓨터뿐 아니라 서비스를 제공하는 웹서버, 이메일(e-mail) 서버 등의 외부서버에 저장되기도 한다. 그러나 최근 신세계몰 등 23여개의 인터넷 사이트에서 개인정보가 대량으로 유출된 사건[6]과 같이 외부공간에 저장된 정보는 해킹과 바이러스를 통해 유출될 위험이 있다. 데이터 누출로 인한 개인정보 유출은 개인의 프라이버시 침해를 초래하고, 기업이나 정부의 정보유출은 국가적인 측면에서도 상당한 손실이다.

접수일(2010년 9월 27일), 수정일(2011년 1월 13일), 게재확정일(2011년 4월 18일)
* 이 논문은 2010년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (한국연구재단-2010-0003388)
* 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (한국연구재단-2010-0029121)
† 주저자, eom_je@hanmail.net
‡ 교신저자, donghlee@korea.ac.kr

데이터 프라이버시를 제공하기 위한 기술로는 데이터를 암호화하여 저장하는 방법이 있다. 정당한 키를 가진 사용자만이 암호화된 데이터를 복구해볼 수 있기 때문에 키를 모르는 공격자는 저장된 암호문으로부터 그 내용에 대한 정보를 얻을 수 없어 안전성이 보장된다. 그러나 데이터를 암호화하여 저장하면 데이터베이스 관리가 어렵고, 검색성능이 떨어지는 문제가 발생한다. 이러한 문제점을 해결하기 위한 방법으로 2000년에 Song 등에 의해 웹하드나 블로그와 같이 공개된 DB에 적용 가능한 암호화된 데이터 상에서의 키워드(keyword)를 검색할 수 있는 시스템이 제안되었다[5]. Boneh 등이 제안한 검색 가능한 공개키 암호시스템은 (Public Key Encryption with Keyword Search, PEKS)은 이메일 시스템에서 메일내용과 키워드를 각각 수신자의 공개키로 암호화하고 함께 전송하여, 수신자가 자신의 개인키를 사용하여 키워드를 검색할 수 있는 기법이다[2]. 수신자는 검색하고자 하는 키워드와 자신의 개인키로 트랩도어(trapdoor)를 생성하여 서버에 전송하고, 서버는 트랩도어와 키워드 암호문을 검증(Test) 함수에 넣어 동일한 키워드인지 비교, 검색한다. 해당하는 키워드를 포함하고 있다면 메일을 암호화된 상태로 수신자에게 전송한다. 즉, 서버는 복호화 과정 없이 키워드를 검색할 수 있다. 이후 Byun 등은 검색 가능한 공개키 암호시스템(PEKS)과 PEKS를 기반으로 한 몇 가지 기법들이 오프라인에서 키워드 추측 공격(keyword guessing attack)이 가능하다는 것을 보였다[3]. 정당한 트랩도어를 얻은 임의의 공격자는 키워드를 추측하여 수신자의 공개키로 키워드 암호문을 생성할 수 있기 때문에 검증 함수를 이용하여 트랩도어에 사용된 키워드와 추측한 키워드가 동일한지 확인할 수 있다. 또한 PEKS 기반의 시스템은 동일한 키워드에 대해서는 송신자와 관계없이 하나의 트랩도어에 대한 정보가 드러나면 그에 대응하는 모든 키워드 암호문의 정보가 노출된다는 문제점이 있다.

트랩도어에 대응하는 키워드 암호문을 생성할 수 있는 능력을 제한한다면 키워드 추측 공격에 대한 안전성을 제공할 수 있게 된다. 더 나아가 이러한 구조는 앞에서 언급한 트랩도어에 대응하는 키워드 암호문에 대한 안전성의 문제도 해결하게 된다. 트랩도어에 대응하는 암호문을 생성할 수 있는 능력을 제한하기 위해서 키워드와 수신자의 개인키, 그리고 송신자의 공개키를 이용해서 트랩도어를 생성하고, 키워드와 수신자의 공개키, 그리고 송신자의 개인키를 이용해서 키워드 암호문을 생성하는 방법을 고려하였다.

키워드를 암호화하는 과정에서 송신자의 개인키를 사용하기 때문에 개인키를 모르는 공격자는 정당한 트랩도어를 얻는다 해도 대응하는 키워드 암호문을 생성할 수 없다. 따라서 이 시스템은 임의의 공격자에게 검증 능력이 제한되어 키워드 추측 공격에 안전하고, 동시에 암호문에 대한 안전성을 보장한다.

검색측면에서 보면, 트랩도어를 생성할 때 송신자의 공개키를 사용하기 때문에 수신자가 특정 송신자를 지정하여 검색할 수 있다. 즉, 수신자가 서버로부터 전송받는 암호문은 선택한 키워드를 포함하면서 수신자가 지정한 송신자로부터 생성된 메일 암호문이다. 만일 기본적인 이메일 시스템에서와 같이 송신자의 정보가 알려져 있거나 송신자별로 메시지가 분류되어 있다면, 수신자가 트랩도어를 생성할 때 검색하고자 하는 송신자의 범위를 제한하여 서버에 전송하게 되는 경우 서버의 검색범위를 줄여서 Test단계에서의 계산량을 줄일 수 있다.

본 논문에서는 검증 능력이 제한된 검색 가능한 공개키 암호(Public Key Encryption with Keyword Search for Restricted Testability, PEKS-RT) 기법을 제안하고, 그에 대한 안전성을 BDH 가정(Bilinear Diffie-Hellman Assumption)에 기반을 두고 랜덤 오라클(random oracle) 모델에서 증명한다.

1.1 관련연구

암호화된 데이터 상에서 효율적인 검색능력을 제공하기 위해 2000년에 Song 등에 의해 공개된 DB에 적용 가능한 암호화된 데이터 상에서의 키워드를 검색할 수 있는 시스템이 제안되었다. 이후 많은 키워드 검색 시스템이 제안되었고, 20004년 Boneh 등은 겹선형 함수를 사용한 효율적인 검색 가능한 공개키 암호 시스템(Public Key Encryption with Keyword Search, PEKS)을 제안하였다. 메일내용과 키워드를 각각 수신자의 공개키로 암호화하고 함께 전송하여, 수신자가 자신의 개인키를 사용하여 키워드를 검색할 수 있는 시스템으로 암호화된 메일을 복호화과정 없이 검색하는 기능을 제공한다. 송신자와 수신자, 그리고 서버로 구성되며, 송신자가 특정 수신자의 공개키로 메일내용(M)과 키워드(W)를 암호화하여 다음과 같은 형태로 서버에 전송한다.

$$Enc_{pk_A}(M) \| PEKS(pk_A, W_1) \| \cdots \| PEKS(pk_A, W_m)$$

서버는 저장된 키워드 암호문들과 수신자로부터 전송받은 검색요청 데이터인 트랩도어(trapdoor)를 검증 함수에 넣어 동일성 검사를 한 뒤, 동일한 키워드가 검색되는 경우 해당하는 메일을 암호화된 상태로 수신자에게 전송한다. 이와 같이 서버는 복호화 과정 없이 수신자가 원하는 메일을 검색하여 전송할 수 있다. 그러나 Byun 등은 검색 가능한 공개키 암호시스템(PEKS)과 PEKS를 기반으로 한 몇 가지 기법들이 오프라인 상에서 키워드 추측 공격(keyword guessing attack)이 가능하다는 것을 보였다. 정당한 트랩도어를 얻은 임의의 공격자는 키워드를 추측하여 수신자의 공개키로 키워드 암호문을 생성할 수 있기 때문에 검증 함수를 이용하여 트랩도어에 사용된 키워드와 추측한 키워드가 동일한지 확인할 수 있다. 이후 Baek 등은 수신자와 서버 사이에 안전한 채널(secure channel)이 필요 없도록 수신자와 서버의 공개키로 키워드를 암호화하여 그 서버만이 검증할 수 있는 시스템을 제안하였다[1]. 트랩도어가 드러나도 키워드 암호문과의 비교는 서버만 할 수 있기 때문에 암호문에 대한 안전성이 보장된다. 그러나 이 시스템에서 트랩도어는 기존 PEKS와 같은 구성으로 되어있기 때문에 PEKS와 동일한 구조로 키워드 암호문을 생성하고 검증한다면 여전히 키워드 추측 공격이 가능하다. 이에 대해 Rhee 등은 수신자와 서버의 공개키로 키워드를 암호화하여 지정된 서버만이 검증할 수 있고, 트랩도어가 드러나도 키워드 암호문과의 관련성을 추측할 수 없는 키워드 추측 공격에 안전한 PEKS 기법을 제안하였다[4]. 그러나 이러한 PEKS 기반의 시스템은 동일한 키워드에 대해서는 송신자와 관계없이 언제나 하나의 트랩도어로 검색하는 것이 가능하기 때문에 하나의 트랩도어에 대한 정보가 드러나면 그에 대응하는 모든 키워드 암호문의 정보가 노출된다. 이처럼 암호화된 데이터에서의 검색기술에 대한 연구가 활발하게 이루어지고 있고, 이러한 연구는 암호화된 데이터에서 다양한 쿼리(query)가 가능하도록 진화되고 있다.

본 논문의 구성은 다음과 같다. 제 2장에서는 기본적인 배경지식을 설명하고, 제 3장에서는 PEKS-RT 기법에 대한 안전성 모델을 정의한다. 제 4장에서는 PEKS-RT 기법을 제안하고, 그에 대한 안전성을 증명한다. 마지막으로 제 5장에서는 결론을 맺는다.

II. 배경지식

본 장에서는 제안하는 기법의 구성 및 안전성 증명에 필요한 겹선형 함수(bilinear map) 및 BDH 가정(Bilinear Diffie-Hellman Assumption)을 설명한다.

2.1 겹선형 함수

겹선형 함수 (Bilinear Maps) : G_1과 G_2를 위수가 소수 p인 순환군(cyclic group)이라 하고 g는 G_1의 생성원(generator)이라 하자. 이때, 군 G_1과 G_2에서 이산대수문제(DLP)는 어렵다고 가정한다. 아래와 같은 조건을 만족하는 함수 $e: G_1 \times G_1 \rightarrow G_2$를 유효한 겹선형 함수(admissible bilinear map)라고 한다. 겹선형 함수(bilinear map) $e: G_1 \times G_1 \rightarrow G_2$와 그룹 G_2가 존재하면 G_1을 겹선형 그룹(bilinear group)이라고 부른다.

- **겹선형성(bilinearity)**: 임의의 $g \in G_1$와 $x, y \in Z_p^*$에 대하여 $e(g^x, g^y) = e(g, g)^{xy}$를 만족한다.
- **비소실성(non-degeneracy)**: $e(g, g) \neq 1$을 만족하는 $g \in G_1$이 존재한다.
- **계산 가능성(computability)**: 임의의 $g_1, g_2 \in G_1$에 대하여 $e(g_1, g_2)$를 계산하는 효율적인 알고리즘이 존재한다.

$e(,)$는 $e(g^x, g^y) = e(g, g)^{xy} = e(g^y, g^x)$를 만족하고 이를 페어링 연산의 대칭성(symmetric)이라 부른다.

2.2 BDH 가정

Hardness 가정 (Hardness Assumption) : 본 논문에서 제안한 기법은 BDH 가정 (Bilinear Diffie-Hellman Assumption)의 안전성에 기반을 둔다.

BDH 가정(Bilinear Diffie-Hellman Assumption) : g를 G_1의 생성원이라 하자. 주어진 입력 값 $g, g^a, g^b, g^c \in G_1$에 대하여 $e(g, g)^{abc} \in G_2$를 계산하는 문제를 BDH 문제라고 정의한다. 이 때, BDH 문제를 푸는데 있어서 의미 있는 확률로 효율적으로 계산할 수 있는 알고리즘 A가 존재하지 않는다면 BDH문제는 풀기 어렵다(intractable)고 정의한다. 알고리즘 A의 이점(advantage)은 다음과 같은 확률 값으로 정의된다.

$$\Pr[A(g, g^a, g^b, g^c) = e(g, g^{abc})] \geq \epsilon$$

III. 형식적 정의 및 안전성 모델

본 장에서는 PEKS-RT 시스템의 형식적 정의와 증명에 필요한 안전성 모델에 대하여 정의한다. PEKS-RT 시스템은 송신자의 정보가 사용되는 점을 제외하고 Boneh 등이 제안한 검색 가능한 공개키 암호 시스템과 유사하게 구성된다.

3.1 검증 능력이 제한된 검색 가능한 공개키 암호시스템의 형식적 정의

송신자(Sender, S)는 메시지 M에 대하여 다음과 같은 형태의 암호문을 생성하여 전송한다.

$$Enc_{pk_R}(M)\|PEKS-RT(pk_R, sk_S, W_1)\| \\ \ldots \|PEKS-RT(pk_R, sk_S, W_n)$$

수신자(Receiver, R)의 공개키로 암호화하는 일반적인 공개키 암호 시스템을 사용하여 메시지를 암호화하고, 검색이 가능하도록 키워드 암호문을 생성하여 함께 전송한다. 본 논문에서는 키워드 암호문을 생성하는 기법을 제안한다. 검증 능력이 제한된 검색 가능한 공개키 암호시스템(PEKS-RT)은 다음의 다항식 시간 알고리즘들로 구성된다.

- **송신자(S) 키 생성 알고리즘** $KeyGen_S(1^\lambda)$: 보안 상수(security parameter) λ를 입력받은 후, 송신자의 공개키/ 개인키 쌍 $[pk_S, sk_S]$을 생성한다.
- **수신자(R) 키 생성 알고리즘** $KeyGen_R(1^\lambda)$: 보안상수 λ를 입력받은 후, 수신자의 공개키/ 개인키 쌍 $[pk_R, sk_R]$을 생성한다.
- **암호화된 검색정보 알고리즘** $PEKS-RT(pk_R, sk_S, W)$: 수신자의 공개키 pk_R와 송신자의 개인키 sk_S, 그리고 키워드 W를 입력받은 후, 암호화된 검색정보 $C=PEKS-RT(pk_R, sk_S, W)$를 생성한다.
- **트랩도어 알고리즘** $Trapdoor(pk_S, sk_R, W)$: 송신자의 공개키 pk_S와, 수신자의 개인키 sk_R, 그리고 키워드 W를 입력받은 후, 암호화된 키워드인 트랩도어 T_W를 생성한다.
- **검증 알고리즘** $Test(C, T_W)$: 암호화된 검색정보 $C=PEKS-RT(pk_R, sk_S, W)$와 트랩도어 $T_{W'}$를 입력받은 후, $W=W'$이면 "yes"를 $W\neq W'$이면 "no"를 출력한다.

3.2 검증 능력이 제한된 검색 가능한 공개키 암호화 시스템의 안전성 모델

본 논문에서 PEKS-RT 기법에 대한 안전성의 정의는 암호문이 트랩도어(trapdoor) 없이는 정보를 노출하지 않는다는 것을 보장한다. 이때, 공격자는 자신이 선택한 키워드 W에 대한 트랩도어 T_W를 얻는 것이 가능한 능동적인(active) 공격자 A를 가정한다. 공격자 A의 목적은 트랩도어를 얻을 수 없는 두 개의 키워드 W_0, W_1 중 하나에 대한 암호문이 주어졌을 때, 어떠한 키워드에 대한 암호문인지를 결정하는 것이다. PEKS-RT 기법에서 정의된 안전성은 다음의 게임을 이용한다.

- **셋업(Setup)** : 챌린저(challenger)는 키 생성 알고리즘을 이용하여 송신자와 수신자의 공개키/ 개인키 쌍 $[pk_S, sk_S]$, $[pk_R, sk_R]$을 생성하고, 공개키 pk_S와 pk_R을 공격자 A에게 전송한다.
- **질의 1단계(트랩도어 쿼리)**: 공격자 A는 자신이 선택한 키워드 W에 대한 트랩도어 값을 질의하고 챌린저는 T_W를 반환한다.
- **챌린지(Challenge)** : 공격자 A는 두 개의 챌린지 키워드 W_0, W_1을 선택한다. 이때, 챌린지 키워드는 질의 1단계에서 트랩도어 질의를 하지 않았던 것으로 선택해야한다. 챌린저는 임의로 $b\in\{0,1\}$를 선택하여 $C=PEKS-RT(pk_R, sk_S, W)$ 값을 계산하여 공격자에게 전송한다.
- **질의 2단계(트랩도어 쿼리)** : 공격자 A는 $W\neq W_0, W_1$인 키워드 W를 선택하여 그에 대한 트랩도어 값을 질의하고, 챌린저는 T_W를 반환한다.
- **추측(Guess)** : 공격자 A는 $b'\in\{0,1\}$을 추측하여 결과를 낸다. 이때, $b=b'$이면 공격자 A는 게임에서 이긴다.

PEKS-RT 기법에서 공격자 A의 이점(advantage)은 다음과 같이 정의한다.

$$Adv_A(\lambda)=|\Pr[b=b']-\frac{1}{2}|$$

정의 1. 임의의 다항식 시간 공격자 A에 대해서 $Adv_A(\lambda)$가 negligible하다면 PEKS-RT 기법은 선택한 키워드 공격(adaptive chosen keyword at-

tack)에 대해서 안전(semantically secure)하다고 정의한다.

IV. 제안 기법

본 장에서는 검증 능력이 제한된 검색 가능한 공개키 암호 기법(PEKS-RT)을 제안하고 제안된 기법의 안전성을 랜덤 오라클 모델에서 분석한다. 제안된 기법의 안전성은 BDH(Bilinear Diffie-Hellman) 문제의 어려움에 기반을 둔다.

4.1 PEKS-RT 기법

G_1과 G_2는 위수가 소수 p인 순환군(cyclic group)이고, g는 G_1의 생성원(generator)이라 하자. 함수 $e: G_1 \times G_1 \rightarrow G_2$를 겹선형 함수라고 가정하자. 이때, $e(g, g)$는 G_2의 생성원이다. 두 암호학적 해쉬 함수를 $H_1: \{0,1\}^* \rightarrow G_1$ 과 $H_2: G_2 \rightarrow \{0,1\}^{\log p}$와 같이 정의한다. PEKS-RT 기법은 다음 4개의 다항식 시간 알고리즘으로 구성된다.

- $KeyGen_S(1^\lambda)$: 임의의 랜덤 값 $\alpha \in Z_p^*$를 선택하여, $pk_S = g^\alpha$를 계산한다. 공개키/ 개인키 쌍 $[pk_S = g^\alpha, sk_S = \alpha]$를 생성하여 송신자에게 안전하게 전송하고 pk_S를 공개한다.
- $KeyGen_R(1^\lambda)$: 임의의 랜덤 값 $\beta \in Z_p^*$를 선택하여 $pk_R = g^\beta$를 계산한다. 공개키/ 개인키 쌍 $[pk_R = g^\beta, sk_R = \beta]$를 생성하여 수신자에게 안전하게 전송하고 pk_R을 공개한다.
- $PEKS-RT(pk_R, sk_S, W)$: 수신자의 공개키 $pk_R = g^\beta$와 송신자의 개인키 $sk_S = \alpha$, 그리고 키워드 W를 입력받은 후, 다음을 계산한다.

1. 임의의 $\gamma \in Z_p^*$를 선택하여 $C_1 = (g^\alpha)^\gamma$를 생성한다.
2. $t = e(g^\alpha \cdot H_1(W), (g^\beta)^\alpha)^\gamma$를 계산하여, $C_2 = H_2(t)$를 생성한다.
3. 암호문 $C = [C_1, C_2]$를 생성한다.

- $Trapdoor(pk_S, sk_R, W)$: 송신자의 공개키 $pk_S = g^\alpha$와 수신자의 개인키 $sk_R = \beta$, 그리고 키워드 W를 입력받은 후, 트랩도어 $T_W = (g^\alpha \cdot H_1(W))^\beta$를 출력한다.
- $Test(C, T_W)$: 암호문 $C = [C_1, C_2]$와 트랩도어 T_W를 입력받은 후, $C_2 = H_2(e(T_W, C_1))$의 등호가 성립하면 "yes"를, 성립하지 않으면 "no"를 출력한다.

정확성(Correctness). 위에서 제안한 기법은 정확성을 가짐을 다음과 같이 쉽게 보일 수 있다.

- 트랩도어 검증 과정(Test). 저장된 암호문 $C = [C_1, C_2]$와 질의한 트랩도어 $T_W = (g^\alpha \cdot H_1(W))^\beta$에 대한 검증은 다음과 같은 과정을 통하여 확인할 수 있다.

$$t = e(g^\alpha \cdot H_1(W), (g^\beta)^\alpha)^\gamma = e(g^\alpha \cdot H_1(W), g^\alpha)^{\gamma\beta}$$
$$= e((g^\alpha \cdot H_1(W))^\beta, (g^\alpha)^\gamma) = e(T_W, C_1)$$

4.2 안전성 분석

본 절에서 PEKS-RT 기법의 안전성을 BDH 문제에 기반하여 증명한다.

정리. 두 해쉬 함수 H_1과 H_2를 랜덤 오라클(random oracle)이라고 가정하자. PEKS-RT 기법은 랜덤 오라클 모델에서 BDH(Bilinear Diffie-Hellman) 가정아래 선택한 키워드 공격(Chosen Keyword Attack)에 대하여 안전하다.

증명. 공격자 A를 PEKS-RT 기법에 대한 공격의 이점(advantage) ϵ을 갖는 공격자라고 가정하자. 이때, A는 최대 q_{H_2}개의 H_2 해쉬 쿼리와 최대 q_T개의 트랩도어 쿼리를 만들 수 있다고 가정하자. BDH 문제를 푸는데 있어서 $\epsilon' = \epsilon / e q_T q_{H_2}$의 이점을 갖는 알고리즘 B가 존재한다는 것을 보임으로써 기법에 대한 안전성을 증명한다. 여기서 e는 자연 상수(natural logarithm)이다.

g는 G_1의 생성원이고, BDH 문제의 입력 값으로 $g, u_1 = g^\alpha, u_2 = g^\beta, u_3 = g^\gamma \in G_1$이 주어졌다고 가정하자. B의 목표는 $v = e(g, g)^{\alpha\beta\gamma}$를 계산하는 것이다. B는 A를 하위 루틴으로 실행하고 A의 공격환경을 다음과 같이 시뮬레이션 한다.

- **셋업** : B는 임의의 $r \in Z_p^*$를 선택하여 $pk_S = g^r$를 계산하고, A에게 공개키 pk_S와 $pk_R = u_1$를 전송한다.
- H_1-해쉬쿼리 : A는 랜덤 오라클 H_1에 언제든지 질의할 수 있다. 질의에 응답하기 위해서 B는

$< W_i, h_i, a_i, c_i >$의 리스트를 L_{H_1}에 저장하고 관리한다. 초기단계의 L_{H_1}는 공집합(empty set)이다. A가 $W_i \in \{0,1\}^*$에 대한 H_1 질의를 요청할 때, B는 다음과 같이 시뮬레이션 한다.

1. W_i가 이미 $< W_i, h_i, a_i, c_i >$의 형태로 L_{H_1}에 존재하면 B는 $H_1(W_i) = h_i \in G_1$를 반환한다.
2. 그렇지 않으면 B는 $\Pr[c_i = 0] = 1/(q_T+1)$의 확률을 만족하는 임의의 코인(random coin) $c_i \in \{0,1\}$을 생성한다.
3. B는 임의의 $a_i \in Z_p^*$를 선택하여 $c_i = 0$이면 $h_i = u_2 \cdot g^{a_i} \in G_1$을 계산하고 $c_i = 1$이면 $h_i = g^{a_i} \in G_1$을 계산하여 $H_1(W_i) = h_i$를 반환한 후 $< W_i, h_i, a_i, c_i >$를 L_{H_1}에 추가한다.

- H_2-해쉬쿼리 : H_1 오라클 구성과 유사한 방법으로 A는 랜덤오라클 H_2에 언제든지 질의할 수 있다. A가 $t \in G_2$에 대한 H_2 질의를 요청할 때, 만약 t가 이미 L_{H_2}에 (t, V)의 형태로 존재하면 B는 $H_2(t) = V$를 반환한다. 그렇지 않으면 임의의 $V \in \{0,1\}^{\log p}$를 선택하여 $H_2(t) = V$를 반환하고, (t, V)를 L_{H_2}에 추가한다. 초기단계의 L_{H_2}는 공집합(empty set)이다.
- 질의 1단계(트랩도어 쿼리) : A가 키워드 W_i에 대응하는 트랩도어 값을 질의할 때, B는 다음과 같이 시뮬레이션 한다.

1. B는 $H_1(W_i) = h_i \in G_1$를 만족하는 $h_i \in G_1$을 얻기 위해 H_1함수와 L_{H_1}를 검색한다. L_{H_1}에 키워드 W_i에 대응하는 $< W_i, h_i, a_i, c_i >$이 저장되어 있다고 하자. 만약 $c_i = 0$이면 B는 시뮬레이션을 중단한다.
2. $c_i = 1$이면 $h_i = g^{a_i} \in G_1$이므로 B는 A에게 키워드 W_i에 대응하는 정당한 트랩도어 $T_i = u_1^{r+a_i} = (g^r \cdot g^{a_i})^{\alpha} = (g^r \cdot H_1(W_i))^{\alpha}$를 생성하여 T_i를 반환한다.

- **챌린지** : A는 두 개의 키워드 W_0, W_1를 선택하여 B에게 전송한다. B는 다음과 같이 시뮬레이션 한다.

1. B는 $H_1(W_0) = h_0$와 $H_1(W_1) = h_1$을 만족하는 $h_0, h_1 \in G_1$을 얻기 위해서 H_1에 질의한다. $b \in \{0,1\}$에 대해 키워드 W_b에 대응하는 $< W_b, h_b, a_b, c_b >$이 L_{H_1}에 저장되어 있다고 하자. 만약 $c_0 = c_1 = 1$이면 B는 시뮬레이션을 중단한다.
2. $c_0 = 0$이거나 $c_1 = 0$이면 B는 $c_b = 0$인 $b \in \{0,1\}$를 임의로 선택한다. (하나의 값만 0이라면, $c_b = 0$인 b를 선택한다.)
3. B는 임의의 $J \in \{0,1\}^{\log p}$를 선택하여 키워드 W_b에 대한 암호문 $C = [C_1, C_2] = [u_3^{\beta}, J]$를 생성하여 반환한다. $J = H_2(e(g^r \cdot H_1(W_b), u_1^r)^{\gamma})$라 하면 C는 W_b에 대한 정당한 암호문이 되고 다음이 성립한다.

$$\begin{aligned} J &= H_2(e(g^r \cdot H_1(W_b), u_1^r)^{\gamma}) \\ &= H_2(e(g^r \cdot u_2 g^{a_b}, g^{\alpha r})^{\gamma}) \\ &= H_2(e(g^{r+a_b}, g^{\alpha r})^{\gamma} \cdot e(u_2, g^{\alpha r})^{\gamma}) \\ &= H_2(e(g^{\gamma}, g^{\alpha})^{r(r+a_b)} \cdot (e(g,g)^{\alpha\beta\gamma})^r) \end{aligned}$$

- **질의 2단계(트랩도어 쿼리)** : A는 $W_i \neq W_0, W_1$인 키워드 W_i에 대응하는 트랩도어 값을 질의한다. B는 질의 1단계에서와 동일한 방법으로 시뮬레이션 한다.
- **추측** : A는 암호문 C가 W_0에 대한 암호문인지 W_1에 대한 암호문인지 결정하여 $b' \in \{0,1\}$을 결과로 출력한다. B는 L_{H_2}로부터 (t, V)를 임의로 선택하여 $x = t/e(u_1, u_3)^{r(r+a_b)}$를 계산하고, 결과 값으로 $\sqrt[\gamma]{x}$를 출력한다. 여기서 a_b는 챌린지 단계에서 선택된 값이다. A가 정확하게 추측하기 위해서는 $H_2(e(g^r \cdot H_1(W_0), u_1^r)^{\gamma})$와 $H_2(e(g^r \cdot H_1(W_1), u_1^r)^{\gamma})$ 중 적어도 하나는 질의를 했어야만 하므로 L_{H_2}는 1/2의 확률로 $t^* = e(g^r \cdot H_1(W_b), u_1^r)^{\gamma} = e(g,g)^{\alpha\gamma\{\beta + r(r+a_b)\}}$를 포함한다. 만약 B가 L_{H_2}로부터 (t^*, V)를 선택한다면 출력 값 $\sqrt[\gamma]{x^*}$은 $\sqrt[\gamma]{x^*} = \sqrt[\gamma]{t^*/e(u_1, u_3)^{r(r+a_b)}} = e(g,g)^{\alpha\beta\gamma}$ 을 만족한다.

B가 $e(g,g)^{\alpha\beta\gamma}$를 정확하게 출력할 확률이 ϵ'이라는 것을 증명하기 위해 다음 사건을 정의하여 시뮬레이션이 실패하지 않을 확률을 계산한다.

E_1: B는 A의 트랩도어 쿼리 단계에서 시뮬레이션을 중단하지 않을 사건.

E_2: B는 챌린지 단계에서 시뮬레이션을 중단하지 않을 사건.

E_3: A는 $H_2(e(g^r \cdot H_1(W_0), u_1^r)^{\gamma})$와 $H_2(e(g^r \cdot$

$H_1(W_1), u_1^r)^\gamma$) 중 적어도 하나는 질의할 사건.

보조정리 1: $\Pr[E_1] \geq 1/e$

증명. 일반성을 잃지 않고, A가 동일한 키워드에 대한 트랩도어를 질의하지 않는다고 가정한다. $\Pr[\neg E_1] \geq 1/(q_T+1)$임을 보이기 위해, W_i를 공격자가 i번째로 질의한 키워드라 하고, 그에 대응하는 $< W_b, h_b, a_b, c_b >$이 L_{H_1}에 있다고 가정하자. 트랩도어 값으로 반환된 값 중에 c_i에 종속인 값은 $H_1(W_i)$ 뿐이고, $H_1(W_i)$의 분포는 c_i의 분포와 같기 때문에 시뮬레이션이 중단될 확률은 최대 $1/(q_T+1)$이다. A는 최대 q_T번의 트랩토어 값을 질의하므로 트랩도어 쿼리에서 시뮬레이션이 중단되지 않을 확률은 $\Pr[E_1] = (1-1/(q_T+1))^{q_T} \geq 1/e$ 이다.

보조정리 2: $\Pr[E_2] \geq 1/q_T$

증명. $b=0,1$에 대해 W_b에 대응하는 $< W_b, h_b, a_b, c_b >$이 L_{H_1}에 있다고 가정하자. B가 챌린지 단계에서 시뮬레이션을 중단하는 경우는 L_{H_1}에서 W_0, W_1에 대응하는 c_b가 $c_0 = c_1 = 1$을 만족하는 경우이다. W_0와 W_1은 트랩도어 값을 질의하지 않았고, 두 확률 $\Pr[c_b = 0] = 1/(q_T+1)$, $b=0,1$은 서로 독립이기 때문에 시뮬레이션이 중단될 확률은 $\Pr[c_0 = c_1 = 1] = (1-1/(q_T-1))^2 \leq 1-1/q_T$ 를 만족한다. 따라서 챌린지 단계에서 시뮬레이션이 중단될 확률은 $\Pr[E_2] \geq 1/q_T$ 이다.

A는 챌린지 단계에서 질의하는 키워드 W_0와 W_1에 대해서는 트랩도어 값을 질의하지 않기 때문에 사건 E_1과 E_2는 서로 독립이다. 그러므로 $\Pr[E_1 \wedge E_2] \geq 1/(eq_T)$이 성립한다.

보조정리 3: $\Pr[E_3] \geq 2\epsilon$

증명. A가 $H_2(e(g^r \cdot H_1(W_0), u_1^r)^\gamma)$와 $H_2(e(g^r \cdot H_1(W_1), u_1^r)^\gamma)$를 질의하지 않는다고 가정하자. A에게 주어지는 암호문 C에 대해, A가 추측한 결과 값 $b' \in \{0,1\}$은 $b=b'$을 만족할 확률이 최대 1/2이다. A는 공격의 이점 ϵ을 갖는 공격자이므로 $|\Pr[b=b'] - 1/2| \geq \epsilon$이 성립한다.

$$\begin{aligned}
\Pr[b=b'] &= \Pr[b=b'|\neg E_3]\Pr[\neg E_3] + \Pr[b=b'|E_3]\Pr[E_3] \\
&\leq \Pr[b=b'|\neg E_3]\Pr[\neg E_3] + \Pr[E_3] \\
&= \frac{1}{2}\Pr[\neg E_3] + \Pr[E_3] \\
&= \frac{1}{2} + \frac{1}{2}\Pr[E_3] \\
\Pr[b=b'] &\geq \Pr[b=b'|\neg E_3]\Pr[\neg E_3] \\
&= \frac{1}{2}\Pr[\neg E_3] \\
&= \frac{1}{2} - \frac{1}{2}\Pr[E_3]
\end{aligned}$$

두 식에 의해 $\epsilon \leq |\Pr[b=b'] - 1/2| \leq \frac{1}{2}\Pr[E_3]$이 성립하고, 따라서 $\Pr[E_3] \geq 2\epsilon$이다.

보조정리 3에 의해 L_{H_2}에는 적어도 ϵ의 확률로 $t^* = e(g^r \cdot H_1(W_b), u_1^r)^\gamma = e(g,g)^{\alpha\gamma\{\beta + r(r+a_b)\}}$를 만족하는 t^*가 존재한다. L_{H_2}에서 올바른 (t^*, V)를 선택할 확률은 최소 $1/q_{H_2}$이므로 B는 적어도 ϵ/q_{H_2}의 확률로 올바른 값을 얻을 수 있다. 그리고 보조정리 1과 보조정리 2에 의해 B는 적어도 $1/eq_T$의 확률로 시뮬레이션을 중단하지 않는다. 따라서 B는 적어도 $\epsilon/eq_Tq_{H_2}$의 확률로 BDH 문제를 해결할 수 있다. 위의 시뮬레이션을 통해 다음과 같은 결과를 얻을 수 있다. 의미 있는(non-negligible) 확률로 제안 기법의 안전성을 깰 수 있는 알고리즘이 존재한다면, BDH 문제를 해결할 수 있는 효율적인 알고리즘이 존재한다.

V. 결 론

본 논문에서는 검증 능력이 제한된 검색 가능한 공개키 암호시스템(PEKS-RT)을 제안하였다. PEKS-RT 기법에 대한 안전성 모델을 정의하고, 그에 따라 PEKS-RT에 대한 안전성을 랜덤오라클 모델에서 증명하였다. 기존에 검색 가능한 공개키 암호 시스템은 모두 키워드 추측 공격이 가능했다. 본 논문에서 제안한 기법은 임의의 공격자에게 검증 능력이 제한되어 서버를 지정하지 않아도 키워드 추측 공격에 안전하고, 동시에 암호문에 대한 안전성을 보장한다는 점에서 의미가 있다. PEKS-RT 기법에서는 송신자의 수가 증가함에 따라 트랩도어의 수도 함께 증가하는데 이 수를 줄이도록 하는 연구가 필요하다.

참고문헌

[1] J. Baek, R. Safavi-Naini, and W. Susilo, "Public key encryption with keyword search revisited," *Proc. ICCSA 2008, LNCS 5072*, pp. 1249 - 1259, 2008.

[2] D. Boneh, G.D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," *Proc. EUROCRYPT 2004, LNCS 3027*, pp. 506 - 522, 2004.

[3] J.W. Byun, H.S. Rhee, H.A. Park, and D.H. Lee, "Off-line keyword guessing attacks on recent keyword search schemes over encrypted data," *Proc. SDM 2006, LNCS 4165*, pp. 75 - 83, 2006.

[4] H.S. Rhee, J.H. Park, W. Susilo, and D.H. Lee, "Improved searchable Public key encryption with designated tester," *Proc. ASIACCS 2009*, pp. 376 - 379, March 2009.

[5] D. Song, D. Wagner, and A. Perrig, "Practical techniques for searching on encrypted data," *Pro. IEEE Symposium on Security and Privacy*, pp. 44-55, May 2000.

[6] 연합뉴스, "사상 최대 고객정보유출...'잠자는' 제도", 2010.03.12.(http://www.yonhapnews.co.kr/bulletin/2010/03/12/0200000000AKR20100312128000017.HTML)

〈著者紹介〉

엄 지 은 (Eom Ji Eun) 학생회원
2010년 2월: 고려대학교 수학과 학사 졸업
2010년 2월~현재: 고려대학교 정보보호대학원 정보보호학과 석사과정
〈관심분야〉 정보보호이론, 암호 프로토콜, 프라이버시향상기술(PET)

이 현 숙 (Hyun Sook Rhee) 정회원
1998년 2월: 단국대학교 수학과 학사 졸업
2000년 2월: 단국대학교 응용수학과 이학 석사 졸업
2008년 2월: 고려대학교 정보경영공학과 공학 박사 졸업
2008년 3월: 고려대학교 정보경영공학전문대학원, 박사후 연구원
2011년 1월: 삼성전자 책임, 인증 및 보안솔루션 개발
〈관심분야〉 정보보호 이론, IPTV 와 Smart Card 관련 보안기술, 프라이버시향상기술(PET)

이 동 훈 (Dong Hoon Lee) 정회원
1983년: 고려대학교 경제학과 학사 졸업
1987년: Oklahoma University 전산학 석사 졸업
1992년: Oklahoma University 전산학 박사 졸업
1993년~1997년: 고려대학교 전산학과 조교수
1997년~2001년: 고려대학교 전산학과 부교수
2001년~현재: 고려대학교 정보보호대학원 교수
〈관심분야〉 정보보호이론, 암호 프로토콜, USN, 키 교환, 프라이버시향상기술(PET), 익명성 연구