정규논문 (Regular Paper)
방송공학회논문지 제18권 제5호, 2013년 9월 (JBE Vol. 18, No. 5, September 2013)
http://dx.doi.org/10.5909/JBE.2013.18.5.758
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

복호화 위임을 제공하는 효율적인 브로드캐스트 암호시스템

한 수 민[a], 박 승 환[a], 박 종 환[b] ,이 동 훈[a]‡

An Efficient Broadcast Encryption System with Delegation of Decryption

Su Min Han[a], Seung Hwan Park[a], Jong Hwan Park[b], and Dong Hoon Lee[a]‡

요 약

브로드캐스트 암호시스템은 송신자가 수신자 집합을 지정하여 생성한 암호문을 공개된 채널을 통해 전송하면, 수신자 집합에 속하는 정당한 사용자만이 메시지를 복호화 할 수 있는 암호 기법이다. 2005년도에는 공모공격에 안전하며 상수크기의 암호문과 비밀키를 가지는 페어링 기반의 기법이 Boneh 등에 의해 제안되었다. 일반적으로 페어링 기반의 기법은 사용자로부터 많은 연산량을 요구하기 때문에 리소스에 제한이 있는 기기에 적용하기에는 어려움이 있었다. 본 논문에서는 Boneh 등의 기법을 기반으로 브로드캐스트 암호시스템에서 암호문을 효율적으로 복호화 하는 기법(BEWD)을 제안한다. 제안하는 기법에서는 복호화 시에 요구되는 페어링연산과 다른 사용자들의 공개키가 쓰이는 연산을 제 3자인 프록시 서버에 위임함으로써 사용자에게 요구되는 연산량을 줄인다. 또한 사용자는 서버의 올바른 계산을 확인을 할 수 있다. 제안하는 기법은 n-BDHE가정 하에 선택적인 IND-RCCA에 안전한 기법이다.

Abstract

In a Broadcast Encryption System, a sender sends an encrypted message to a large set of receivers at once over an insecure channel and it enables only users in a target set to decrypt the message with their private keys. In 2005, Boneh et al. proposed a fully collusion-resistant public key broadcast encryption in which the ciphertext and the privatekey sizes are constant. In general, pairing-based broadcast encryption system is efficient in bandwidth and storing aspects than non-pairing based broadcast encryption system, however, it requires many computational costs that resource-constrained devices is not suit to be applied. In this paper, we propose a Broadcast Encryption scheme(called BEWD) that user can decrypt a ciphertext more efficiently. The scheme is based on Boneh et al.scheme. More precisely, it reduces receiver's computational costs by delegating pairing computation to a proxy server which computation is required to receiver in Boneh et al.scheme. Furthermore, the scheme enables a user to check if the proxy server compute correctly. We show that our scheme is secure against selective IND-RCCA adversaries under l-BDHE assumption.

Keyword : Broadcast Encryption System, outsourcing, RCCA

a) 고려대학교 정보보호대학원(Graduate School of Information Security, Korea University)
b) 상명대학교 소프트웨어대학 컴퓨터과학과(Division of Computer Science. College of Computer Software and Media Technology. Sangmyung University)
‡ Corresponding Author : 이동훈(Dong Hoon Lee)
E-mail: donghlee@kroea.ac.kr
Tel: +82-2-3290-4259

※ 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2010-0029121).이 논문은 2013년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2013R1A1A2009524). 본 연구는 2013년도 상명대학교 교내연구비를 지원받아 수행하였음.
· Manuscript received 8, August 2013 Revised 25, September 2013 Accepted 25, September 2013

Ⅰ. 서 론

1. 개요

공개키 암호시스템은 암/복호화 과정에서 서로 다른 키를 사용하여 안전한 통신을 하는 암호 기법으로 대칭키 암호시스템의 키 공유문제를 해결하기 위해 1976년 Diffie와 Hellman에 의해 처음 개념이 제시되었다[1]. 송신자는 수신자의 공개키를 이용하여 메시지를 암호화하고 수신자는 자신의 개인키를 이용하여 암호문을 복호화 한다. 일반적으로 공개키 암호시스템에서는 송/수신자간 일대일로 메시지 교환을 한다. 하지만, 송신자가 다수의 수신자에게 같은 메시지를 보내야하는 상황에서 기존의 공개키 암호시스템을 이용하는 경우 송신자는 수신자 각각의 공개키를 이용하여 다수의 암호문을 생성해야 했다. 이는 전체 암호문의 크기가 사용자의 수에 비례하기 때문에 전송량이 커지는 문제가 발생한다. 다른 방법으로는 수신자 집합에게 새로운 키를 부여하여 복호화 키로 사용하도록 하는 것이 있는데 수신자 집합이 설정될 때마다 새로운 키를 생성하기 때문에 사용자가 키를 관리하는데 어려움이 있다.

이처럼 공개키 암호시스템에서 임의의 사용자 집합이 수신자가 될 때 발생하는 전송량 또는 키 관리 문제를 해결하기 위하여 1994년 Fiat와 Naor에 의해 브로드캐스트 암호시스템(Broadcast Encryption)[2]이 처음으로 제안되었다. 브로드캐스트 암호시스템은 송신자가 수신자 집합을 지정하여 생성한 하나의 암호문에 대해 수신자 집합에 속하는 정당한 사용자만이 메시지를 얻을 수 있는 암호 기법이다. 브로드캐스트 암호시스템에서 암호문은 풀헤더와 바디로 구성되어있다. 송신자가 공개키를 이용하여 대칭키를 암호화한 부분인 헤더 Hdr와 수신자 집합 S를 브로드캐스트 암호문의 풀헤더라 하며, 대칭키를 이용하여 메시지 M을 암호화 한 암호문 C_M을 브로드캐스트 암호문의 바디라고 한다. 송신자가 공개된 채널을 통해 암호문 $CT=(S, Hdr, C_M)$을 전송하면 복호화 권한을 가지는 사용자만이 비밀키를 이용해 대칭키를 구하여 메시지 M을 얻는 구조를 가진다. 일반적으로 효율적인 브로드캐스트 암호시스템은 페어링을 기반으로 한다. 페어링 기반의 브로드캐스트 암호시스템은 다른 공개키 브로드 캐스트 암호시스템과 보다 전송량과 저장량 측면에서 좀 더 효율적이지만 사용자로부터 많은 연산량을 요구한다는 문제를 지닌다[3]. [4]에 따르면 타원곡선위에서의 페어링연산은 지수연산보다 약 10배 정도의 연산량을 요구한다. 이에 제한된 리소스를 가져 페어링과 같은 무거운 연산을 하는데 한계를 가지는 기기들의 연산량을 줄이는 방법으로 페어링 위임 프로토콜개념이 제시되었다[5,6]. 페어링 위임 프로토콜이란 페어링 연산을 필요로 하는 사용자가 제 3자인 프록시 서버를 이용하여 페어링 연산을 아웃소싱하는 방법으로, 사용자는 프록시 서버가 연산을 올바르게 하였는지 확인할 수 있으며 프록시 서버는 아웃소싱을 하는 동안 비밀정보(실제 페어링하려고 하는 값)에 대해서 아무런 정보를 얻을 수 없기 때문에 프록시 서버가 낮은 신뢰도를 갖더라도 안전하게 된다.

2. 관련연구

Fiat와 Naor가 제안한 브로드캐스트 암호시스템[2]은 정당하지 않은 사용자들이 공모공격을 하는 경우, 총 공모자의 수가 어떤 임계값(threshold)보다 적을 때 안전성이 증명된다. 이후에 Naor등에 의해 공모자의 수의 제한 없이 공모공격에 안전한 기법[7]이 제시되었으나 이 기법은 암호문과 비밀키가 폐지된 사용자의 집합의 크기에 비례하기 때문에 폐지된 사용자의 집합의 크기가 작아야하는 제한이 있었다. 그 후 Dodis와 Fazio는 Naor 등의 기법을 기반으로 최초로 선택 암호문 공격(CCA)에 안전한 공개키 기반의 브로드캐스트 기법[8]을 제안하였으나 이 기법은 폐지된 사용자의 수가 어떤 임계값 t보다 작아야하고 암호문의 크기가 $O(t)$에 비례하였다. 2005년도에는 Boneh 등에 의해 암호문과 비밀키가 상수 크기를 가지며 공모자 수에 관계없이 공모공격에 안전한 기법(BGW기법)이 최초로 제안되었다[9]. Boneh 등은 표준 가정(standard assumption)보다 강한 가정인 비표준 가정 n-BDHE문제의 어려움을 기반으로 선택적인 선택 평문 공격(SCPA)에 안전한 기법과 선택적인 선택 암호문 공격(SCCA)에 안전한 기법을 제시하였다. 이때 암호문과 비밀키가 상수 크기를 가지는 기법은 공개키가 총

사용자수에 비례하는 크기를 가졌다. 2007년도에는 최초로 아이디 기반의 브로드캐스트 암호 기법[10]이 제안되었다. 이 기법은 BGW기법 보다 짧은 길이의 공개키를 가지나 SCPA에 안전성이 증명되었다. 이후 어댑티브모델로 안전성을 증명한 기법[11]이 제시되었으나 이 기법에서도 공개키의 크기가 총 사용자수에 비례하였고 표준 가정을 사용하지 않았다. 2010년도에는 Lewko등은 듀얼시스템 암호를 이용하여 어댑티브 모델에서 DBDH(Decisional Bilinear Diffie-Hellman)문제와 D-Lin문제(Decisional Linear problem)의 어려움을 가정하여 안전성을 증명한 기법[12]을 제안하였으나 암호문의 길이가 폐지된 사용자의 수에 비례하였다. 2012년도에는 Phan과 Pointcheval 등이 어댑티브 CCA(ACCA)에 안전하며 암호문이 상수 크기를 가지는 동적 브로드캐스트 암호 기법[13]을 제시하였다. 이 기법은 표준 가정(standard assumption)으로 안전성이 증명되지 않았으며 공개키의 크기가 총사용자의 수에 비례하였다. 최근에는 BGW기법을 기반으로 비대칭 페어링을 이용하여 전송량을 줄인 기법[14]이 제시되었으나 BGW기법과 마찬가지로 총 사용자수에 비례하는 크기를 가지는 공개키를 가진다.

앞에서 언급한 효율적인 브로드캐스트 암호시스템들은 모두 페어링을 기반으로 한다. 따라서 리소스가 적은 기기를 사용하는 사용자에게 페어링 기반의 브로드캐스트 암호시스템에 적용하는 데에는 어려움이 있다. 이와 같은 문제를 지니는 페어링기반의 아이디기반 암호시스템(Identity Based Encryption, IBE)과 속성기반 암호시스템(Attribute Based Encryption, ABE)에서도 같은 문제를 해결하기 위해 다음과 같은 활발한 연구가 있었다. 2011년도 속성기반 암호시스템에서는 암호문의 전송량과 사용자의 연산량을 줄이기 위해 Green등이 CPA에 안전한 CP-ABE 아웃소싱 기법[15]을 제안하였다. 이 기법은 Waters의 속성기반 암호 기법[16]이 SCPA에 안전할 때 SCPA와 재사용 가능한 선택 암호문 공격(RCCA)에 안전하게 된다. 이외에도 페어링 위임 프로토콜[5,6] 개념을 이용한 아이디 기반의 아웃소싱기법[17]이 제안되었는데, 이 기법은 Boneh 등의 IBE기법[18]이 IND-ID-CCA에 안전할 때 IND-ID-CCA에 안전한 기법이다.

3. 기여도

본 논문에서는 암호문과 비밀키 크기측면에서 효율적인 BGW기법에 제 3자인 프록시 서버를 이용하여 브로드캐스트 암호문을 효율적으로 복호화 하는 기법을 제안한다. 본 논문에서 제안하는 기법은 BGW기법과 달리 두 개의 변환키를 생성하고 이 두 변환키를 이용하여 복호화를 한다. 제안하는 기법에서 추가적으로 생성되는 첫 번째 키는 프록시 서버가 기존의 BGW기법의 암호문을 엘가말 타입의 암호문[19]으로 변환시킬 때 사용하는 키로 본 논문에서는 공개 변환키(TK_P)라고 부른다. 두 번째 키는 사용자가 엘가말 타입의 암호문을 복호화 할 때 쓰이는 키로 본 논문에서 비밀 변환키(TK_S)라고 부른다. 두 변환키를 생성하기 위해 비밀키에 난수값을 지수승하여 변환키를 생성하였던 [15,17]기법의 방법을 그대로 BGW기법에 적용하는 경우 수신자가 다른 사용자의 공개키를 가지고 그 외의 연산을 수행해야 했다. 이러한 경우 사용자에게 필요한 공개키를 송신자가 암호문과 함께 전송하는 방법과 사용자가 공개키를 미리 저장하는 방법이 있으나 암호문의 오버헤드측면과 사용자의 저장량측면에서 문제가 발생한다. 따라서 제안하는 기법에서는 [15,17]기법과는 달리 겹선형 함수의 성질을 이용해 그룹 G에서의 난수값을 마스킹 값으로 사용하여 두 변환키를 생성한다. 그리고 $Setup$단계에서 총 사용자수에 비례하는 크기를 가지는 공개키 PK를 서버에 저장하고, 상수개의 공개키를 사용자에게 전달한다. 제안하는 기법의 복호화 과정은 다음과 같다. $Transform$단계에서 공개 변환키 TK_P를 받은 서버는 저장된 공개키와 함께 페어링연산과 다른 사용자들의 공개키가 사용되는 연산을 수행한 뒤 부분 복호화된 암호문을 사용자에게 전달하고, 사용자는 $Decrypt$단계에서 상수개의 공개키와 비밀 변환키 TK_S를 이용하여 지수연산과 같은 단순한 연산들을 통해 메시지를 얻는다. 본 기법에서는 연산량이 좋은 프록시 서버를 이용하여 기존의 BGW기법에서 복호화 시에 요구되는 페어링연산과 다른 사용자들의 공개키가 사용되는 연산을 서버에게 위임하기 때문에 사용자의 연산량 측면에서 효율적인 기법이다.

본 논문에서 제안하는 기법은 사용자가 $O(1)$크기의 비밀키와 공캐키만을 가지고 페어링연산과 같이 무거운 연산을 수행하지 않고도 쉽게 복호화를 할 수 있도록 한다. 따라서 스마트 폰과 같이 제한된 리소스를 가지는 기기로도 클라우드 서비스에서 제공하는 컨텐츠를 손쉽게 사용할 수 있다. 기기들은 자신이 원하는 컨텐츠가 있을 때 클라우드 서버에 공개 변환키를 전송하고, 클라우드 서버로부터 받은 $O(1)$크기의 암호문과 기기에 저장된 $O(1)$크기의 비밀키와 공개키를 가지고 $O(1)$정도의 지수연산을 통해 메시지를 얻어볼 수 있다. 또한 제안하는 기법에서 사용자는 부분 복호화된 암호문이 서버의 올바른 계산을 통해 나온 값이 맞는지 확인 할 수 있으며, 서버는 위임연산을 하는 동안 사용자의 비밀키, 비밀 변환키, 메시지와 같은 비밀정보에 대해서 아무런 정보를 얻을 수 없기 때문에 안전하다. 본 논문에서 제안하는 기법을 이용하면 CPA에 안전한 브로드캐스트 암호시스템 존재 하에 RCCA에 안전한 브로드캐스트 암호시스템의 설계가 가능하다. 본 논문에서는 2005년도에 제안된 BGW기법을 기반으로 랜덤오라클 모델에서 안전성을 증명한다.

이후 논문의 구성은 다음과 같다. Ⅱ장에서는 기법의 이해에 필요한 배경지식을 설명한다. Ⅲ장에서는 RCCA에 안전성이 증명된 기법을 제안하며 Ⅳ장에서는 제안한 기법의 안전성 증명을 한다. Ⅴ장에서는 제안한 기법에 대한 분석을 하고, 마지막으로 Ⅵ장에서 결론을 맺는다.

Ⅱ. 배경지식

1. 복호화 위임을 제공하는 브로드캐스트 암호 시스템의 정의

복호화 위임을 제공하는 브로드캐스트 암호시스템(Broadcast Encryption with Outsoucing the Decryption, BEWD)은 다음과 같이 5가지 알고리즘으로 구성된다. 브로드캐스트 암호시스템에서 S는 수신자 집합을, n은 총 사용자의 수를 의미한다. 그리고 $i\in\{1,...,n\}$는 i번째 사용자를 나타낸다. BEWD의 $Setup$ 알고리즘과 $Encrypt$ 알고리즘은 BGW기법[9]에서의 알고리즘과 동일하다.

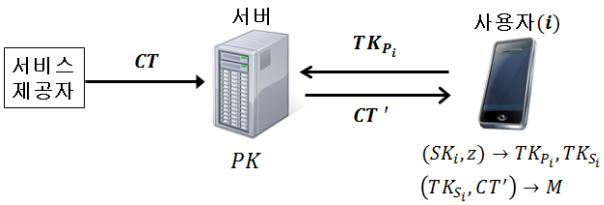

그림 1. 복호화 위임을 제공하는 브로드캐스트 암호시스템(BEWD)
Fig. 1. Broadcast Encryption System with delegation of decryption (BEWD)

- $Setup(n)$: 알고리즘은 총 사용자 수 n을 입력 받고, 공개키 PK와 모든 사용자에 대한 비밀키 $SK_i\in G$를 출력한다. $Setup$단계가 끝나면 사용자는 자신의 비밀키 $SK_i\in G$와 $TKgen$알고리즘에 필요한 공개키, 서버의 올바른 연산을 검증할 때 쓰이는 공개키를 저장한다, 총 사용자수에 비례하는 크기를 가지는 공개키 PK는 서버에 저장하며 송신자는 암호화 시 공개키를 서버에서 받아온다.
- $Encrypt(S,PK,M)$: 암호 알고리즘은 수신자 집합 $S\subseteq\{1,...,n\}$, 공개키 PK와 메시지 M을 입력받으면, 암호문 CT을 출력한다.
- $TKgen(i,SK_i)$: 알고리즘은 사용자를 나타내는 i와 비밀키 SK_i를 입력받으면, 사용자 i의 두 변환키 TK_{P_i}와 TK_{S_i}를 출력한다. 여기서 TK_{P_i}는 서버와 사용자간에 사용하는 공개키이며 TK_{S_i}는 사용자가 비밀로 유지하는 값이다. 본 논문에서는 TK_{P_i}를 공개 변환키라 부르며 TK_{S_i}를 비밀 변환키라 부른다.
- $Transform(i,TK_{P_i},CT,PK)$: 알고리즘은 i와 사용자 i의 공개 변환키 TK_{P_i}, 브로드 캐스트 암호문 CT와 공개키 PK를 입력받는다. 알고리즘은 $i\notin S$인 경우 $\perp$을 출력하고, 그렇지 않은 경우 TK_{P_i}와 PK를 이용하여 부분적으로 복호화한 엘가말 형태의 암호문 CT'을 출력한다.
- $Decrypt(i,TK_{S_i},CT')$: 알고리즘은 i와 엘가말 형태 암호문 CT', 비밀 변환키 TK_{S_i}를 입력받는다. 알고리즘

은 CT'가 부분적으로 복호화 되어있지 않다면 $Transform$알고리즘을 수행한다. 만약 $Transform$알고리즘의 출력 값이 $\perp$이면 $Decrypt$알고리즘도 마찬가지로 $\perp$을 출력하고, 그렇지 않다면 알고리즘은 TK_{S_i}와 검증에 필요한 공개키들을 이용해 CT'을 복호화 하여 메시지 M을 구한다. 이때 알고리즘은 서버의 연산이 올바르게 되었는지 검증을 한다. 만약 검증에 통과한 경우 M을 출력하며 그렇지 않은 경우, $\perp$을 출력한다.

2. 복호화 위임을 제공하는 브로드캐스트 암호 시스템의 안전성모델

반복 불가능한 선택 암호문 공격에 의한 평문 구분 불가능성(IND-RCCA)을 만족하는 BEWD의 안전성 모델은 다음과 같다. 어떠한 다항식 시간(polynomial-time)안에 공격자 A가 챌린저 B와의 게임에서 이길 확률이 무시할 수 있는(negligible) 값일 때, 그 브로드캐스트 암호시스템은 IND-RCCA을 만족한다고 정의한다. 총 사용자의 수 n은 공격자 A와 챌린저 B에게 주어진다. 다음은 선택적인 IND-RCCA에 대한 안전성 모델이다.

- $init$: 공격자 A는 공격할 수신자 집합 $S^* \subseteq \{1,...,n\}$을 챌린저 B에게 전송한다.
- $Setup$: 챌린저 B는 $Setup$알고리즘을 수행하고, 공개키 PK와 집합 S^*에 속하지 않은 모든 사용자에 대한 비밀키 $SK_i \in G$ 와 $TKgen$알고리즘에 필요한 공개키를 공격자 A에게 전송한다.
- $Phase1$: 공격자 A는 변환키 질의 $TKquery$와 복호질의 $Decrypt$가 가능하다. 챌린저 B는 테이블 T를 초기화 하고 공격자의 질의를 다음과 같이 답한다.
  - $TKquery(i)$: 공격자 A가 변환키 질의를 하면 챌린저 B는 $TKgen$알고리즘을 수행한 뒤 공격자가 요청한 사용자 i에 대한 변환키 TK_{P_i}를 공격자 A에게 전송하고, (i, SK_i, TK_i)를 테이블 T의 i번째 행에 저장한다. ($TKgen$알고리즘을 수행하여 나온 두 변환키 중 사용자가 비밀로 유지해야 하는 값인 TK_{S_i}는 전송하지 않는다)
  - $Decrypt(i, CT')$: 공격자 A는 사용자 i와 부분 복호화된 암호문 CT'를 입력한다. 챌린저 B는 테이블 T의 i번째 행에 (i, SK, TK)가 저장되어 있지 않은 경우 $\perp$을 출력하고, 그렇지 않은 경우 메시지 M을 출력한다.
- $Challenge$: 공격자 A는 길이가 같은 두 메시지 M_0, M_1을 선택하여 챌린저 B에게 전송한다. 챌린저 B는 $b \in \{0,1\}$를 임의로 뽑은 뒤 M_b에 관한 브로드캐스트 암호문 CT를 계산한다. B는 공격자 A에게 챌린지로 (CT, M_0, M_1)을 전송한다.
- $Phase2$: $Phase2$는 $Phase1$과 유사하다.
  - $TKquery(i)$: $Phase1$과 같이 공격자 A는 사용자 i에 대하여 변환키 질의를 하고, 공개 변환키 TK_{P_i}를 받는다.
  - $Decrypt(i, CT')$: 공격자 A가 $Challenge$에서 내놓은 M_0 또는 M_1을 메시지로 갖는 암호문에 대한 복호질의를 할 수 없는 것을 제외하면, $Phase1$와 같다.
- $Guess$: 공격자 A는 b에 대한 응답으로 $b' \in \{0,1\}$을 챌린저 B에게 전송한다.

* 재사용 가능한 선택 암호문 공격에 대한 안전성[20]

일반적으로 선택 암호문 공격(CCA)에 대한 안전성은 공격자로부터 임의의 암호문에 대한 변형 불가능성(Non-malleability)을 제공해야한다. 이는 CCA에 안전한 기법에서의 공격자는 Challenge단계에서 내놓은 메시지와 같은 의미를 갖는 암호문을 생성할 수 없음을 의미한다. 이때 재사용 가능한 선택 암호문 공격(Replayable-CCA:RCCA)이란 공격자가 $Challenge$에서 내놓은 메시지와 같은 의미를 가지는 암호문에 대해서는 복호질의를 할 수 없는 공격이다. 재사용 가능한 선택 암호문 공격은 선택 평문공격(CPA)보다는 강력하지만 CCA보다는 약간의 제약이 존재함을 알 수 있다.

3. 겹선형 함수

위수를 소수 p로 갖는 곱셈 군 G와 G_T가 있고, G의 생성원(generator)이 g라고 가정하자. 군 G와 G_T에서 모두 이산

대수문제(discrete logarithm problem)가 어렵다고 가정한다면, 다음과 같은 조건을 만족하는 함수 $e : G \times G \rightarrow G_T$를 겹선형 함수(bilinear maps)라 한다.

1) 겹선형성(bilinearity): 임의의 두 원소 $g, h \in G$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 $g \in G$가 존재한다.
3) 계산 가능성(computability): 임의의 $g, h \in G$에 대해서 $e(g,h)$를 효율적으로 계산할 수 있는 알고리즘이 존재한다.

4. BDHE-가정

본 논문에서 제안하는 브로드캐스트 암호시스템의 안전성은 결정적 BDHE(Decisional Bilinear Diffie-Hellman Exponent)가정에 기반 한다.

정의 1. BDHE-가정(Bilinear Diffie-Hellman Exponent assumption)

군 G는 소수 p를 위수로 갖는다고 가정하자. l-BDHE가정이란 다음과 같은 집합 $(g, h, g_i = g^{\alpha^i} \in G \mid i \in \{1,2,...,l, l+2,...,2l\})$이 주어졌을 때 $e(g,h)^{\alpha^{l+1}} \in G_T$을 의미 있는(non-negligible) 확률로 계산할 수 있는 알고리즘 A가 존재하지 않음을 의미한다. 알고리즘 A가 $e(g,h)^{\alpha^{l+1}}$를 계산할 확률은 다음과 같다.

$$p_r\left[A(g,h,g_1,...,g_l,g_{l+2},...g_{2l}) = e(g,h)^{\alpha^{l+1}}\right] \leq \epsilon$$

정의 2. 결정적 BDHE-가정(Decisional Bilinear Diffie-Hellman Exponent assumption)

이 가정은 군 G에서 결정적(decisional) BDHE문제이다. 이때 $\overrightarrow{y_{g,\alpha,l}} = (g_1,...,g_l,g_{l+2},...g_{2l})$이라고 가정한다. l-DBDHE가정이란 다음과 같은 집합 $\left(g,h,\overrightarrow{y_{g,\alpha,l}}, T\right)$이 주어졌을 때, T값이 $e(g,h)^{\alpha^{l+1}}$인지 임의의 난수인지를 의미 있는 확률로 판단할 수 있는 알고리즘 A가 존재하지 않음을 의미한다. 알고리즘 A는 l-BDHE문제를 푸는데 다음과 같은 이점(advantage)을 가진다.

$$Adv(A) = \left|\left[A\left(g,h,\overrightarrow{y_{g,\alpha,l}}, e(g,h)^{\alpha^{l+1}}\right)=1\right] - \left[A(g,h,\overrightarrow{y_{g,\alpha,l}}, T)=1\right]\right| \leq \epsilon$$

III. 제안하는 기법

본 절에서는 BGW기법 중 비밀키와 암호문이 상수크기를 가지며 CPA에 안전한 기법에 Fujisaki와 Okamoto의 변환기법[21]을 이용하여 랜덤 오라클 모델에서 RCCA에 안전한 기법으로 변환한다. 제안하는 기법은 BGW의 기법과 달리, 공개키 파라미터에 두 개의 해쉬 함수 $H_1 : \{0,1\}^* \rightarrow Z_p, H_2 : \{0,1\}^* \rightarrow \{0,1\}^k$와 검증에 필요한 공개값 $e(g,g), e(g_n,g_1)$이 추가되었으며, $TKgen$알고리즘과 $Transform$알고리즘이 추가되었다.

- $Setup(n)$: 그룹 G를 위수를 소수 p로 갖는 군이라 가정한다. 알고리즘은 총 사용자 수 n을 입력 받으면, 생성원 $g \in G$와 $\alpha \in Z_p$를 임의로 선택한다. 그리고 $e(g,g)$와 $e(g_n,g_1)$를 계산하며, $i = \{1,...,n,n+2,...,2n\}$에 대하여 $g_i = g^{\alpha^i}$를 계산하고 임의의 값 $\gamma \in Z_P$을 뽑아 $v = g^\gamma$으로 설정하며 임의의 해쉬 함수 $H_1 : \{0,1\}^* \rightarrow Z_p$, $H_2 : \{0,1\}^* \rightarrow \{0,1\}^k$를 선택한다. 알고리즘은 공개키를 다음과 같이 $PK = (g, g_1,...,g_n, g_{n+2},...,g_{2n}, v, H_1, H_2, e(g,g), e(g_n,g_1)) \in G^{2n+1}$으로 설정하고, 사용자 $i \in \{1,...,n\}$에 대한 비밀키를 $SK_i = g_i^\gamma = v^{\alpha^i} \in G$으로 설정한다. 알고리즘은 공개키 PK와 n명의 사용자에 대한 비밀키 $SK_i \in G$ 들을 출력한다. $Setup$단계가 끝나면 사용자는 자신의 비밀키 $SK_i \in G$와 값 $g, e(g,g), e(g_n,g_1)$를 저장하며, 공개키 PK는 서버에 저장한다. 송신자는 암호화 시 공개키를 서버에서 받아온다.
- $Encrypt(S,PK,M)$: 그룹 G는 위수를 소수 p로 갖는 겹선형 군이라 가정한다. 알고리즘은 수신자 집합

$S \subseteq \{1,...,n\}$, 공개키 PK와 메시지 M을 입력받는다, 알고리즘은 G에서 임의의 값 R을 뽑은 뒤, 해쉬 함수를 이용하여 두 난수 값 $H_1(R,M)=t$, $H_2(R)=r$을 얻는다. 알고리즘은 메시지를 암/복호 하는데 사용되는 대칭키 $K=e(g_{n+1},g)^t$을 계산한다. 값 $e(g_{n+1},g)$는 공개키 g_n,g_1을 이용하여 계산한다. 알고리즘은 헤더 Hdr를 다음과 같이 $Hdr=\left(g^t,\left(v\prod_{j\in S}g_{n+1-j}\right)^t\right)\in G^2$계산하고 대칭키 K를 이용하여 R을 $E_K(R)$와 같이 암호화하고, 메시지 M에 대한 암호문을 $C_M=(E_K(R),M\oplus r)$으로 설정한다. 알고리즘은 브로드 캐스트 암호문 $CT=(S,Hdr,C_M)$을 출력한다,

- $TKgen(i,SK_i)$: 알고리즘은 사용자를 나타내는 i와 비밀키 SK_i를 입력받으면, Z_p에서 임의의 값 z_i을 선택하여 비밀 변환키를 $TK_{S_i}=z_i$로, 공개 변환키를 $TK_{P_i}=SK_i\times g^{zi}$으로 설정한 뒤 TK_{P_i}와 TK_{S_i}를 출력한다. 여기서 TK_{P_i}는 서버와 사용자간에 사용하는 공개키이며 TK_{S_i}는 사용자가 비밀로 유지하는 값이다.
- $Transform(i,TK_{P_i},CT,PK)$: 알고리즘은 사용자를 나타내는 i와 공개 변환키 TK_{P_i} 그리고 수신자 집합이 S인 브로드 캐스트 암호문 CT와 공개키 PK를 입력받는다. 알고리즘은 $i\notin S$인 경우 $\perp$을 출력하고, 그렇지 않은 경우 브로드 캐스트 암호문을 $CT=(S,Hdr,E_K(R),M\oplus r)=(S,C_0,C_1,C_2,C_3)$이라 할 때 다음과 같이 T_0,T_1을 계산한다.

$$T_0=\frac{e(g_i,C_1)}{e\left(TK_{P_i}\times\prod_{j\in S_a,j\neq b}g_{n+1-j+b},C_0\right)}=\frac{K}{e\left(g^{z_i},C_0\right)},$$

$$T_1=e(g,C_0)=e(g,g^t)$$

알고리즘은 부분적으로 복호화된 엘가말 형태의 암호문 $CT'=(T_0,T_1,T_2=C_2,T_3=C_3)$을 출력한다.

- $Decrypt(i,TK_{S_i},CT')$: $Decrypt$알고리즘은 사용자를 나타내는 i와 비밀 변환키 TK_{S_i}, 엘가말 형태 암호문 $CT'=(T_0,T_1,T_2,T_3)$를 입력받는다. 만약 CT'가 부분적으로 복호화 되어있지 않았다면 $Transform$알고리즘을 수행한다. $Transform$알고리즘의 출력 값이 $\perp$이라면 알고리즘도 마찬가지로 $\perp$을 출력하고, 그렇지 않다면 다음을 수행한다. 알고리즘은 TK_{S_i}와 공개키 $e(g,g)$, $e(g_n,g_1)$를 이용하여 $K=T_0(T_1)^{TK_{S_i}}$, $R=D_K(T_2)$, $M=T_3\oplus H_2(R)$,$t=H_1(R,M)$을 계산하고 두 식 $T_0=\frac{e(g_n,g_1)^t}{e(g,g)^{tz_i}}$, $T_1=e(g,g)^t$을 만족하는지 검증한다. 만약 검증에 통과하는 경우 부분 복호화된 암호문은 올바른 공개 변환키를 통해 부분 복호화된 것이므로 사용자는 메시지 M을 출력하며, 그렇지 않은 경우 $\perp$을 출력한다.

- 정확성($Correctness$)

$$T_0(T_1)^{TK_S}$$

$$=\left[e(g_i,C_2)/e\left(TK_P\times\prod_{j\in S_a,j\neq i}g_{n+1-j+i},C_1\right)\right]\times\left[e(g,g^t)\right]^{TK_s}$$

$$=\left[\frac{e\left(g^{\alpha^i},\left(v\times\prod_{j\in S_a}g_{n+1-j}\right)^t\right)}{e\left(v^{\alpha^i}g^z\times\left(\prod_{j\in S_a,j\neq i}g_{n+1-j+i}\right),g^t\right)}\right]\times\left[e(g,g^t)\right]^{TK_s}$$

$$=\left[\frac{e\left(g^{\alpha^{n+1}},g^t\right)e\left(g^{\alpha^i},\left(v\times\prod_{j\in S_a,j\neq i}g_{n+1-j}\right)^t\right)}{e(g^z,g^t)\,e\left(v^{\alpha^i}\prod_{j\in S_a,j\neq i}g_{n+1-j+i},g^t\right)}\right]\times\left[e(g,g^t)\right]^{TK_s}$$

$$=\left[\frac{e\left(g^{\alpha^{n+1}},g^t\right)}{e(g^z,g^t)}\right]\times\left[e(g,g^t)\right]^{TK_s}$$

$$=\left[\frac{e\left(g^{\alpha^{n+1}},g^t\right)}{e(g^z,g^t)}\right]\times\left[e(g,g^t)\right]^{z}=e(g,g)^{\alpha^{n+1}\times t}=K$$

Ⅳ. 제안하는 기법의 안전성 증명

보조정리 1. l-BDHE 문제의 어려움을 가정할 때, BGW 기법은 선택적인 IND-CPA에 안전하다.

증명. BGW기법은 Decisional l-BDHE 가정 하에 선택적인 IND-CPA에 안전함이 증명되었다[9]. 본 논문에서 구체적인 증명은 생략한다. □

보조정리 2. BGW기법이 선택적인 IND-CPA에 대해 안전할 때, 본 논문에서 제안한 BEWD기법은 랜덤 오라클 모델에서 선택적인 IND-RCCA에 안전하다.

증명. 복호화 위임을 제공하는 브로드캐스트 암호시스템이 있을 때, 선택적인 RCCA 환경에서 다항식 시간 이내에 의미 있는 확률로 암호문을 구별할 수 있는 공격자 A가 존재한다고 가정한다. 이때 공격자 A의 이득이 ε라 한다면, BGW기법에 대하여 선택적인 RCCA환경에서 ε보다 약간 적은 이득을 갖는 시뮬레이터 B를 설계할 수 있다. 총 사용자의 수 n은 공격자 A와 시뮬레이터 B에게 주어진다고 가정한다.

- $init$: 시뮬레이터 B는 BGW 챌린저에게 공격할 사용자 집합을 내놓기 위해 공격자 A를 실행시킨다. 공격자 A는 공격할 수신자 집합 $S^* \subseteq \{1,...,n\}$을 시뮬레이터 B에게 전달하고, 시뮬레이터 B는 A로부터 받은 S^*을 BGW 챌린저에게 전송한다.
- $Setup$: 시뮬레이터 B는 BGW 챌린저로부터 공개키 $PK=\left(g,g_{1,...,}g_n,g_{n+2,...,}g_{2n},v\right)\in G^{2n+1}$와 집합 S^*에 속하지 않은 모든 사용자에 대한 비밀키 $SK_i \in G$ 를 전송 받는다. 시뮬레이터 B는 임의의 두 해쉬함수 $H_1 : \{0,1\}^* \rightarrow Z_p, H_2 : \{0,1\}^* \rightarrow \{0,1\}^k$를 선택하며 비밀키 $\left\{SK_i | i \notin S^*\right\}$들과 값 $g, e(g,g)$를 공격자인 A에게 전송한다.
- $Phase1$: 공격자 A는 변환키 질의 $TKquery$와 복호 질의 $Decrypt$가 가능하다. 챌린저 B는 테이블 T를 초기화하고 공격자의 질의를 다음과 같이 답한다.
  - $TKquery(i)$:
    공격자 A가 시뮬레이터 B에게 챌린지 사용자 집합 S^*에 속하는 사용자에 대하여 변환키 질의를 하면, B는 다음과 같이 "가짜의" 변환키를 생성한다. 시뮬레이터 B는 임의의 값 $z' \in Z_p$을 선택하여 사용자 i에 대한 비밀 변환키를 $TK_S = z'$로, 공개 변환키 TK_P는 다음과 같이 설정한다.

$$TK_P = g^{z'}\left(\prod_{j \in S_a, j \neq i} g_{n+1-j+i}\right)^{-1}$$

$$= g_i^{\gamma}(g_{n+1})^{-1}\left(\prod_{j \in S_a, j \neq i} g_{n+1-j+i}\right)^{-1} \times g^{-\alpha^i\gamma+\alpha^{n+1}+z'}$$

$$= g_i^{\gamma}\left(\prod_{j \in S_a} g_{n+1-j+i}\right)^{-1} \times g^{-\alpha^i\gamma+\alpha^{n+1}+z'}$$

$$= SK_i \times g^{-\alpha^i\gamma+\alpha^{n+1}+z'} = SK_i \times g^z$$

시뮬레이터 B는 공격자 A에게 공개 변환키 TK_P를 전송하고, $(i, SK, TK=(TK_P, TK_S))$를 테이블 T의 i번째 행에 저장한다. 위의 값에서 $-\alpha^i\gamma+\alpha^{n+1}+z'$은 TK 생성 시에 쓰이는 난수 z의 역할을 한다.

- $Decrypt(i)$:
  1) 시뮬레이터 B는 테이블 T에 저장된 값 $TK_S = z_i'$을 가지고 $\beta = e(g_i,v)^{-1}e(g_n,g)e(g^{z_i'},g)$을 계산한다.
  2) B는 테이블 T_1에 저장된 각각의 (R,M,t)에 대하여 $D_{e(g_n,g_1)^t}(T_2) = R$를 만족하는지 확인한다.
  3) 만약 2)에서 매칭되는 테이블 값이 존재하지 않다면, B는 공격자 A에게 $\perp$을 출력한다.
  4) 만약 2)에서 매칭되는 테이블 값이 두 쌍 이상 존재한다면, B는 시뮬레이션을 중단한다.
  5) 3)4)과 같은 경우가 아닌 경우 (R,M,t)는 유일하게 매칭되는 값이 되고, 시뮬레이터 B는 테이블 T_2에서 (R,r)를 찾는다. 값이 존재하지 않는 경우에는 공격자 A에게 $\perp$을 출력한다.
  6) B는 $(R,M,t),(R,r)$을 가지고 다음과 같이 검증을 한다. $T_0 = e(g_n,g_1)^t/\beta^t, T_1 = e(g,g)^t, T_2 = E_{e(g_n,g_1)^t}(R), T_3 = M \oplus r$
  7) 검증에 모두 통과한 경우 시뮬레이터 B는 공격자 A에게 메시지 M를 출력한다. 검증에 모두통과하지 않을 경우, A에게 $\perp$을 출력한다.

- $Challenge$: 공격자 A는 길이가 같은 두 메시지

M_0, M_1을 선택하여 시뮬레이터 B에게 전송하고, B는 다음과 같은 과정을 따른다.

1) 시뮬레이터 B는 길이가 같은 두 임의의 값 R_0, R_1을 선택하고 BGW 챌린저에게 보내면 챌린지 값 $CT=(S, Hdr, E_K(R_b))=(S, C_0, C_1, C_2)$을 전송받는다.
2) B는 M_0과 같은 길이를 갖는 C_3을 임의로 선택한다.
3) B는 2)에서 선택한 임의의 값을 이용하여 공격자 A에게 챌린지로 $CT=(S, C_0, C_1, C_2, C_3)$을 전송한다.

- $Phase\,2$: $Phase\,2$에서는 $Phase\,1$과 마찬가지로 $TK\,query(i)$에 대한 답을 하지만 $Decrypt(i, CT')$에서는 약간의 제한을 둔다.
  - $TK\,query(i)$: $Phase\,1$과 같이 변환키 질의를 하고, 사용자 i에 대한 공개 변환키 TK_{P_i}를 받는다.
  - $Decrypt(i, CT')$: 공격자 A는 $Challenge$에서 받은 M_0 또는 M_1을 메시지로 갖는 암호문에 대한 복호질의를 할 수 없는 것을 제외하면, $Phase\,1$와 동등하다.
- $Guess$: 최종적으로 공격자 A는 시뮬레이션을 중단하거나, B에게 한 비트를 출력한다. 시뮬레이터 B는 응답을 무시하고 테이블 T_1에서 R_0 또는 R_1이 있는지 찾는다. 테이블 T_1에 $(\bullet, R_b, \bullet)$형태의 질의가 존재하지 않는 경우 B는 임의의 한 비트를 출력하고, 하나의 R_b 값이 저장되어 있는 경우 b를 출력한다. □

정리 1. 결정적 n-BDHE 문제의 어려움을 가정할 때. 본 논문에서 제안한 BEWD기법은 선택적인 IND-RCCA에 안전하다.

증명. 본 절의 보조정리 1,2를 통해 다음과 같은 결과를 알 수 있다. 만약 의미 있는 확률로 선택적인 IND-RCCA에 안전한 BEWD기법의 안전성을 깰 수 있는 알고리즘이 존재한다면 선택적인 IND-CPA에 안전한 BGW기법을 깰 수 있는 효율적인 알고리즘이 존재한다고 할 수 있다. 그리고 BGW기법은 결정적 n-BDHE가정 하에 선택적인 IND-CPA에 안전함이 증명되었다. 따라서 본 논문에서 제안한 BEWD기법은 결정적 n-BDHE 문제의 어려움을 가정하였을 때 선택적인 IND-RCCA에 안전한 기법임이 증명되었다. □

표 1. 비밀키 사이즈가 $O(1)$인 브로드캐스트 암호시스템들의 암호문, 공개키 사이즈와 사용자에게 요구되는 연산량 및 안전성 비교 (n:총 사용자의수, S:수신자 집합, l:수신자 그룹의 최대 크기($|S| \le l$), r:탈퇴된 사용자의 수, $|\widetilde{PK}|$=사용자가 복호화 시에 사용하는 공개키 사이즈)

Table 1. Security and efficiency comparison between broadcast encryption schemes with constant-size private-key (n: number of total , S: receiver set, l: maximal size of receiver set($|S| \le l \le n$), r: number of revoked users, $|\widetilde{PK}|$=public-key size required to user when decrypting) (n-BDHE: n-Decisional Bilinear Diffie-Hellman Exponent assumption, SUF: Strongly Unforgeable, GDDHE: General Decisional Diffie-Hellman Exponent Assumption, PRF: Pseudo Random Function, DBDH: Decisional Bilinear Diffie-Hellman, D-Lin: Decisional Linear problem, GKEA,: Generalized Knowledge of Exponent Assumption, UOWHF: Universal One-Way Hash Function, SCPA: Selective Chosen Plaintext Attack, SCCA: Selective Chosen Ciphertext Attack, ACPA: Adaptive Chosen Plaintext Attack, ACCA: Adaptive Chosen Ciphertext Attack, RCCA: Replayable Chosen Ciphertext Attack)

Scheme	$\|C\|$	$\|PK\|$	$\|\widetilde{PK}\|$	페어링 연산	지수 연산	위임 여부	Security	ROM	Assumption
BGW05[9]	$O(1)$	$O(n)$	$O(\|S\|)$	$O(1)$	$O(1)$	NO	SCCA	NO	n-BDHE , SUF
DEL07[10]	$O(1)$	$O(l)$	$O(\|S\|)$ $(\|S\| \le l)$	$O(1)$	$O(\|S\|)$ $(\|S\| \le l)$	NO	SCPA	YES	GDDHE
GW09[11]	$O(\|S\|)$	$O(l)$	$O(l)$	$O(1)$	$O(1)$	NO	ACPA	YES	n-BDHES, PRF
LSW10[12]	$O(r)$	$O(1)$	$O(1)$	$O(1)$	$O(r)$	NO	ACPA	NO	DBDH, dLin
PPSS12[13]	$O(1)$	$O(n)$	$O(\|S\|)$	$O(1)$	$O(1)$	NO	ACCA	NO	n-BDHE,GKEA, UOWHF
DGB13[14]	$O(\sqrt{n})$	$O(\sqrt{n})$	$O(\|S\|)$	$O(1)$	0	NO	SCCA	NO	n-BDHE
제안하는 기법	$O(1)$	$O(n)$	$O(1)$	0	$O(1)$	YES	RCCA	YES	n-BDHE

V. 분 석

본 논문에서는 Boneh 등이 제안한 기법을 기반으로 브로드캐스트 암호시스템에서 수신자가 제 3자인 프록시 서버를 이용하여 효율적인 복호화를 할 수 있는 기법(BEWD)을 제안하였다. 제안하는 기법을 통해 사용자는 리소스가 적은 기기를 가지고 기기에 저장된 $O(1)$크기의 비밀키와 공개키로 간단한 연산을 거친 뒤 메시지를 얻을 수 있다. 이 때 사용자는 브로드캐스트 암호문을 복호화하기 위해 기본적으로 기기에 비밀키를 저장하고 있음을 가정하기 때문에 표 1에서는 비밀키가 $O(1)$크기를 가지는 브로드캐스트 암호시스템에 관하여 비교 및 분석을 한다.

먼저 암호문과 비밀키가 상수크기를 갖는 효율적인 BGW기법[9]을 살펴보면, 이 기법은 브로드캐스트 암호시스템의 총 사용자의 수가 n명일 때, 암호문과 비밀키는 $O(1)$크기를 가지지만, 공개키는 $O(n)$크기를 가진다. 그리고 이 기법은 수신자 집합이 S일 때 $O(|S|)$크기의 공개키가 필요하며 페어링연산이 요구된다. 만약 수신자 집합이 커지는 경우 많은 양의 공개키가 복호화 하는데 쓰이게 되는데 이 공개키들을 송신자가 전송하는 경우 $|C| + |\widetilde{PK}|$크기의 암호문을 전송해야 하므로 암호문의 오버헤드가 너무 커진다. 다른 방법으로 사용자가 $|PK|$ 크기의 공개키를 미리 저장하는 경우 보통 $n \approx 2^{30}$이기 때문에 사용자의 기기에 많은 양의 저장량을 요구하게 되므로 리소스가 적은 기기에는 적합하지 않다. 그리고 BGW기법과 마찬가지로 복호화 시에 $O(|S|)$크기의 공개키가 쓰이는 PPSS기법과 DGB기법에서도 복호화 시에 페어링 연산이 요구되며 앞에서와 같이 암호문의 오버헤드 또는 저장량의 문제가 발생한다. 또한 DEL기법과 GW기법 같은 경우 공개키의 사이즈가 수신자 집합의 최대크기 $l(\leq n)$에 비례하기 때문에 수신자 집합의 최대 크기가 크게 설정된 경우 BGW기법과 같은 문제가 발생한다. 그리고 LSW기법과 같은 경우 공개키의 사이즈가 $O(1)$이며 표 1에 나와 있는 기법들 중에서 가장 좋은 가정을 사용하지만 실제 사용자는 복호화 시에 공개키를 사용하지 않고, 폐지된 사용자의 수에 비례하는 크기를 가지는 암호문을 가지고 복호화 한다. 따라서 폐지된 사용자의 수가 많아지는 경우 사용자에게 전달되는 암호문의 오버헤드가 커지는 문제가 발생한다. 표 1에 나와 있는 브로드캐스트 암호시스템들을 살펴보면 공통적으로 복호화 시에 페어링연산을 요구하며 공개키 또는 암호문의 사이즈로 인해 암호문의 오버헤드측면 또는 저장량 측면에서 문제가 발생함을 알 수 있다.

이를 해결하기 위해 제안하는 기법에서는 겹선형 함수의 성질을 이용해 두 개의 변환키를 생성하고 서버에 공개키 PK를 저장하여 복호화 시에 사용자들의 공개키가 필요한 연산과 페어링 연산을 서버에게 위임함으로서 사용자의 연산량을 줄였다. 수신자의 집합이 S일 때 암호문과 비밀키가 상수크기를 가지며 CCA에 안전한 BGW기법에서는 복호화 과정에서 2번의 페어링연산과 $O(|S|)$번의 곱 연산이 사용자에게 요구되는 반면 제안하는 기법에서는 사용자가 대칭키 K를 얻기 위해 한 번의 지수연산과 1번의 곱 연산을 수행하고, T_0, T_1을 만족하는지 검증하는데 세 번의 지수연산과 한 번의 곱 연산을 수행한다. 즉, 사용자는 페어링 연산없이 총 네 번의 지수연산과 2번의 곱 연산을 통해 올바른 메시지를 얻을 수 있다. 제안하는 기법은 다른 기법들과 달리 강력한 프록시 서버에 페어링 연산과 그 외 연산들을 아웃소싱을 함으로써 추가적으로 발생하는 통신오버헤드를 고려하더라도 사용자에게 $O(1)$정도의 저장량과 $O(1)$정도의 지수연산만이 요구되기 사용자의 효율적인 복호화가 가능하다. [15]에 의하면 100개의 속성을 포함하는 암호문에 대해 사용자 모바일 기기만을 가지고 복호화하는 경우 메시지를 얻는데 최소 30초 이상이 소요되었으며 기기의 많은 배터리가 사용되었지만, 프록시 서버를 사용하는 경우 프록시 서버가 부분 복호화를 하는데 2초가 소요되며 부분 복호화된 암호문을 가지고 사용자 기기 자체적으로 연산하는 데에는 0.06초만이 소요되었다. 일반적으로 프록시 서버는 다수의 사용자가 존재함에 따라 효율적인 연산을 위하여 α개의 프로세스를 가지고 처리하는 경우 n명의 사용자가 부분복호화를 요청 시 총 $2\times\left\lceil\frac{n}{\alpha}\right\rceil$ 초가 소요된다. 그리고 제안하는 기법은 BGW기법 중 SCPA에 안전한 기법을 기반으로 설계되었기 때문에 BGW와 같은 가정인 n-BDHE문제의 어려움에 기반 한다. 본 논문에서는

아웃소싱 기법을 통해 사용자는 제 3자인 프록시 서버로부터 부분 복호화된 암호문을 전송받고 간단한 연산을 통해 메시지를 얻고자 함을 목적으로 하기 때문에 [20]에서 제시한 재사용 가능한 선택 암호문 공격(RCCA)에 대해 안전성이 증명되며 안전성 증명 시에 랜덤오라클이 사용된다. 하지만 사용자가 복호화 시에 저장해야하는 공개키와 비밀키의 사이즈와 암호문의 전송량, 메시지를 얻는데 요구되는 연산량 측면에서 다른 기법들보다 매우 뛰어나다. 또한 사용자는 메시지를 얻는 과정에서 부분 복호화된 암호문이 프록시 서버로부터 올바른 계산을 통해 나온 값인지 검증을 할 수 있으며, 서버는 연산을 하는 동안 사용자의 비밀키, 메시지, 비밀 변환키에 대하여 아무런 정보를 알 수 없기 때문에 안전하다. 따라서 본 기법을 이용하면 연산력이 낮고 저장량이 충분하지 않은 기기에서도 브로드캐스트 암호기법을 적용시킬 수 있다.

Ⅵ. 결 론

본 논문에서는 BGW기법[9]을 기반으로 브로드캐스트 암호시스템에서 암호문을 효율적으로 복호화 하는 기법(BEWD)을 제안하였다. 디지털방송 콘텐츠의 보호를 위한 기술 중 하나인 CAS(Conditional Access System)는 방송망으로 통해 전송되는 콘텐츠에 대해 수신 권한을 제어하는 기술이다. CAS기술은 허가된 사용자에게만 수신권한을 부여한다는 점에서 BEWO기법과 유사하지만 송신자는 정당한 수신자에게 자격을 부여하기 위해 각 수신자와 공유하고 있는 비밀키를 가지고 EMM(Entitlement Management Message)을 생성해야한다. 즉, CAS에서는 송신자가 수신권한을 가지는 총 사용자수에 비례하는 EMM을 생성해야 하지만 BGWO기법에서는 송신자가 수신자 집합을 정하여 하나의 암호문을 생성함으로써 수신자 집합에 속하는 사용자가 자동으로 자격을 부여받게 되므로 효율적이라 할 수 있다.

BEWD기법은 BGW기법에 페어링 위임 프로토콜[15,17] 개념을 이용하여 페어링연산과 다른 사용자들의 공개키를 사용하는 연산들을 프록시 서버에 아웃소싱하여 복호화 시에 높은 연산량을 요구했던 기존의 브로드캐스트 암호시스템의 문제점을 해결하였다. 사용자는 $O(1)$크기의 공개키와 비밀키를 저장하고 $O(1)$크기의 암호문을 받아 $O(1)$정도의 지수연산만으로 복호화가 가능하기 때문에 제한된 리소스를 가진 기기를 이용해서 클라우드 서비스 이용이 가능하다. 또한 수신자는 신뢰할 수 없는 서버로부터 연산이 잘 되었는지 확인이 가능하며 서버는 위임연산을 하는 동안 메시지 또는 비밀키에 관한 아무런 정보를 얻지 못한다는 장점이 있었다. 하지만 본 논문에서 제시한 기법은 랜덤 오라클을 사용하여 IND-RCCA에 안전성이 증명되었다. 따라서 향후에 랜덤 오라클을 사용하지 않고 IND-RCCA에 대해 안전성을 보이는 것은 좋은 연구 과제가 될 것이다.

참 고 문 헌

[1] W. Diffie and M. E. Hellman, “New directions in cryptography,” IEEE Transactions on Information Theory, vol. IT-22, no. 6, pp. 644-654, Nov. 1976

[2] A. Fiat and M. Naor, “Broadcast encryption,” Advances in Cryptology, CRYPTO'93, LNCS 773, pp. 480 - 491, 1994.

[3] Sherman S.M. Chowa, Man Ho Aub, Willy Susilob, “Server-aided signatures verification secure against collusion attack,” Information Security Technical Report, Volume 17, Issue 3, February 2013, Pages 46 - 57

[4] Xavier Boyen, “A Tapestry of Identity-Based Encryption: Practical Frameworks Compared,” International Journal of Applied Cryptography, volume 1, number 1, pages 3-21. Inderscience, 2008

[5] B. G. Kang, M. S. Lee, and J. H. Park. Efficient delegation of pairing computation.Cryptology ePrint Archive, Report 2005/259, 2005. http://eprint.iacr.org/.

[6] B. Chevallier-Mames, J.S. Coron, N. McCullagh, D. Naccache, and M. Scott. “Secure delegation of elliptic-curve pairing,” In CARDIS, LNCS 6035, pp. 24-35, Springer, 2010.

[7] D. Naor, M. Naor, and J. Lotspiech. “Revocation and tracing schemes for stateless receiver,” Advances in Cryptology, CRYPTO'01, LNCS 2139, pp. 41-62, 2001.

[8] Y. Dodis and N. Fazio, “Public key trace and revoke scheme secure against adaptive chosen ciphertext attack,” In Public Key Cryptography－PKC 2003. Springer Berlin Heidelberg, pp. 100-115. Jan. 2003.

[9] D. Boneh, C. Gentry, and B. Waters, “Collusion resistant broadcast encryption with short ciphertexts and private keys,” Advances in Cryptology, CRYPTO'05, LNCS 3621, pp. 258 - 275, 2005.

[10] C. Delerablee, “Identity-based broadcast encryption with constant size

ciphertexts and private keys," Advances in Cryptology, CRYPTO'07, LNCS 4833, pp. 200 – 215, 2007.
[11] C. Gentry and B. Waters, "Adaptive security in broadcast encryption systems (with short ciphertexts)," Advances in Cryptology, CRYPTO'01, LNCS 5479, pp. 171-188, 2009.
[12] A.B. Lewko, A. Sahai, and B. Waters, "Revocation systems with very small private keys," Proceedings of the 2010 IEEE Symposium on Security and Privacy, pp. 273-285, May 2010.
[13] D. H. Phan, D. Pointcheval, S. F. Shahandashti, and M. Streer, "Adaptive CCA broadcast encryption with constant size secret keys and ciphertexts," Information Security and Privacy. Springer Berlin Heidelberg, pp. 308-321, 2012.
[14] Renaud Dubois, Aurore Guillevic, Marine Sengelin Le Breton, "Improved Broadcast Encryption Scheme with Constant-Size Ciphertext," Pairing-Based Cryptography – Pairing 2012 Lecture Notes in Computer Science Volume 7708, 2013, pp 196-202
[15] M. Green, S. Hohenberger, B. Waters, Outsourcing the Decryption of ABE Ciphertexts". Proceedings of the 20th USENIX conference on Security. USENIX Association, pp. 34-34. 2011.
[16] B. Waters. "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," In Public Key Cryptography–PKC 2011. Springer Berlin Heidelberg, pp. 53 – 70. 2011
[17] J.K. Liu, C.K. Chu, J. Zhou, "Identity-Based Server-Aided Decryption," In ACISP 2011, LNCS, vol. 6812, pp. 337-352, Springer, 2011
[18] D. Boneh and M. Franklin, " Identity-Based Encryption from the Weil Pairing," Advances in Cryptology, CRYPTO'01, LNCS 2139, pp. 213-229, 2001.
[19] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," IEEE Transactions on Information Theory, vol. IT-31, no. 4, pp. 469-472, 1985.
[20] R. Canetti, H. Krawczyk, and J.B. Nielsen. "Relaxing chosen-ciphertext security." Advances in Cryptology, CRYPTO'03, LNCS 2729, pp. 565 – 582, 2003.
[21] E. Fujisaki and T. Okamoto, "Secure integration of asymmetric and symmetric encryption schemes," Advances in Cryptology, CRYPTO'99, LNCS 1666, pp. 537 – 554, 1999.

저 자 소 개

한 수 민

- 2012년 2월 : 고려대학교 정보수학과 (학사)
- 2012년 3월 ~ 현재 : 고려대학교 정보보호대학원 정보보호학과 석사과정
- 주관심분야 : 정보보호이론, 암호 프로토콜, 프라이버시향상기술(PET)

박 승 환

- 2009년 2월 : 숭실대학교 수학과 (학사)
- 2011년 8월 : 고려대학교 정보경영공학과 (석사)
- 2011년 9월 ~ 현재 : 고려대학교 정보보호대학원 정보보호학과 박사과정
- 주관심분야 : 정보보호이론, 암호 프로토콜, 프라이버시향상기술(PET)

박 종 환

- 1999년 2월 : 고려대학교 수학과 (학사)
- 2004년 2월 : 고려대학교 정보보호대학원 정보보호학과 (석사)
- 2008년 8월 : 고려대학교 정보경영공학전문대학원 정보보호학과 (박사)
- 2009년 6월 ~ 2011년 5월 : 경희대학교 응용과학대학 학술연구교수
- 2011년 6월 ~ 2013년 8월 : 고려대학교 BK21정보보호사업단 연구교수
- 2013년 8월 ~ 현재 : 상명대학교 컴퓨터과학과 조교수
- 주관심분야 : Pairing-based 암호, 브로드캐스트 암호, ID-based 암호, 전자서명 등

저 자 소 개

이 동 훈

- 1983년 8월 : 고려대학교 경제학과 (학사)
- 1987년 12월 : Oklahoma University 전산학 (석사)
- 1992년 5월 : Oklahoma University 전산학 (박사)
- 1992년 8월 : 단국대학교 전자계산학과 전임강사
- 1993년 3월 ~ 1997년 2월 : 고려대학교 전산학과 조교수
- 1997년 3월 ~ 2001년 2월 : 고려대학교 전산학과 부교수
- 2001년 2월 ~ 현재 : 고려대학교 정보보호대학원 교수
- 주관심분야 : 정보보호이론, 암호 프로토콜, USN, 키 교환, 프라이버시향상기술(PET), 익명성 연구