정규논문 (Regular Paper)
방송공학회논문지 제19권 제4호, 2014년 7월 (JBE Vol. 19, No. 4, July 2014)
http://dx.doi.org/10.5909/JBE.2014.19.4.502
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

2-SD 방식에 기반한 브로드캐스트 암호시스템의 안전성 분석

이 재 환[a)], 박 종 환[a)‡]

Security Analysis of Broadcast Encryption System Based on 2-Subset Difference Method

Jae Hwan Lee[a)] and Jong Hwan Park[a)‡]

요 약

브로드캐스트 암호시스템은 한명의 송신자가 다수의 수신자에게 메시지를 암호화하여 안전하게 전송하는 기법이다. 2001년 Naor, Naor, Lotspiech가 이진트리 하에서 Subset Difference(SD) 방식을 이용하여 제안한 브로드캐스트 암호시스템이 가장 효율적인 기법으로 알려지고 있다. 2006년 장지용, 양대헌, 송주석은 SD 방식을 변형한 2-SD 방식을 이용하여 새로운 브로드캐스트 암호시스템을 제안하였다. 장지용 등의 기법은 기존 Naor 등이 제안한 SD 방식에 비해 전송량을 거의 절반으로 줄일 수 있는 획기적인 기법이었으며, 2009년 대한민국에 특허로 등록되기까지 하였다(등록번호: 100879083). 그러나 본 논문에서는 장지용 등이 제안한 2-SD 방식의 브로드캐스트 암호시스템 안전성의 기본 전제인 공모공격(collusion attack)에 전혀 안전하지 않다는 것을 보인다.

Abstract

Broadcast encryption is a cryptographic primitive that allows a sender to securely transmit a message to a set of receivers. The most influential broadcast encryption system was proposed in 2001 by Naor, Naor, Lotspiech, based on binary trees and the Subset Difference (SD) method. In 2006, Jang, Nyang, and Song suggested a new broadcast encryption system that can reduce transmission rate by 50% compared to the SD method, by introducing the so-called ‘2-SD’ method. Their result was later given the registration of a patent in Korea (registration number: 100879083). Unfortunately, however, this paper shows that Jang et. al.‘s broadcast encryption system is not secure against collusion attacks that are considered as being the basic security requirement in designing broadcast encryption.

Keyword : broadcast encryption, collusion attack, subset difference method

a) 상명대학교 소프트웨어대학 컴퓨터과학과 (Department of Computer Science, College of Software, Sangmyung University)
‡ Corresponding Author : 박종환(Jong Hwan Park)
E-mail: jhpark@smu.ac.kr
Tel: +82-2-781-7589
※본 연구는 2013년도 상명대학교 교내연구비를 지원받아 수행하였음.
‡ Manuscript received May 13, 2014 Revised July 16, 2014 Accepted July 28, 2014

Ⅰ. 서 론

브로드캐스트 암호시스템[1]은 한명의 송신자가 대규모의 수신자에게 공통된 메시지를 암호화하여 전송하는 시스템으로, 대규모 수신자 중에서 권한이 있는 수신자들만이 메시지를 볼 수 있도록 하는 것을 가능하게 한다. 주요 응용

환경으로는 유선 케이블 TV, 위성 방송, 또는 각종 유료 콘텐츠의 배포 시스템 등이 있다.

브로드캐스트 암호시스템의 효율성은 수신자의 비밀키 저장량, 복호화 시 요구되는 계산량, 암호문 전송량 측면에서 분석된다. 브로드캐스트 암호시스템 설계의 목표는 세 가지 요소가 모두 가능한 한 작은 값을 취하도록 하는 것인데, 기존에 제안된 대부분의 브로드캐스트 암호시스템은 세 가지 효율성 요소 간의 trade-off 관계를 보여주는 경우가 많다. 현재까지 제안된 기법 중 가장 효율적인 것으로 인정받는 것은 Naor, Naor, Lotspiech[2]가 제안한 것으로 이진트리 위에서 Subset Difference (SD) 방식에 기반한 암호시스템(이하에서는 'SD 기법'이라 명명함)이다. SD 기법은 전체 사용자 수를 n, 탈퇴자 수를 r이라 할 때, $O(\log n)$ 계산량, $O(r)$ 전송량, 그리고 $O(\log^2 n)$ 저장량을 갖는다. 이후 [3], [4]에서 SD 기법은 (전송량을 증가시키면서) 저장량을 줄이는 방법으로 개선될 수 있었다. 최근에 Bhattacherjee와 Sarkar[5]는 이진트리 기반의 SD 기법을 k진트리 기법으로 확장하는 내용을 발표하였다.

브로드캐스트 암호시스템의 안전성은 공모공격 하에서 고려된다. 공모공격(collusion attack)은 공격자가 선택한 사용자들의 비밀키가 주어지고, 공격자는 이 비밀키들을 이용하여 암호문에 숨겨진 메시지의 정보를 획득하려고 하는 것을 말한다. 당연히 안전한 브로드캐스트 암호시스템이라 하면 공모자의 수에 (전체 사용자 수 n보다 적은 공모자 수 이내에서) 제한이 없어야 하여, 공모공격에 대한 내성을 가져야 한다. SD 기법과 그 변형된 것들[2-5]은 모두 이러한 공모공격에 대해 안전하다고 증명되었다.

브로드캐스트 암호시스템은 내부 공모자를 추적할 수 있는 traitor tracing[6] 기능을 제공하기도 한다. traitor tracing이란 내부의 정당한 권한을 가진 사용자가 자신의 비밀키를 이용하여 암호문을 복호화할 수 있는 기기를 만든 경우, 해당 기기에 비밀키를 제공한 사용자를 추적하는 것을 말한다. 또한 브로드캐스트 암호시스템은 전체 구성원이 공유하는 공개키를 이용하여 구성원 중 누구라도 송신자가 될 수 있는 환경에서 설계될 수 있다. 이를 공개키 기반 브로드캐스트 암호시스템이라 하는데, 공개키 환경에서도 브로드캐스트 암호시스템과 traitor tracing기능을 모두 제공하는 결과들 [7-10]이 제안되었다.

2006년 장지용, 양대헌, 송주석[11]은 기존 SD 기법을 이용한 브로드캐스트 암호시스템을 변형하여 전송량을 거의 절반으로 줄일 수 있는 획기적인 기법(이하에서는 '2-SD 기법'으로 명명함)을 제안하였다. 일반적으로 암호문 전송량과 비밀키 저장량은 반비례 관계를 보이는데, 2-SD 기법은 이러한 일반적인 관계를 따르지 않고 있다. 그 이유는 SD 기법과 비교할 때, 단지 암호문의 전송량 길이를 절반으로 줄이는데 비해, 수신자의 비밀키 저장량과 복호화 계산량은 변함이 없는 성능을 보여주었기 때문이다. 이를 위한 아이디어는 SD 기법이 하나의 Subtree 하에서 하나의 Subtree만 배제할 수 있었으나, [11]에서는 이를 변형하여 하나의 Subtree 하에서 두 개의 Subtree까지 배제할 수 있는 방법([11]에서는 이를 '2-Subset Difference'라 명명되었음)이었다. 그러나 본 논문에서는 그 획기적인 효율성에도 불구하고, 2-SD 기법이 공모공격에 전혀 안전하지 않음을 보인다. 특히 단 두 명의 탈퇴자들이 공모하면 암호문으로 숨겨진 메시지를 쉽게 복구할 수 있다는 것을 보일 것이다.

II. 브로드캐스트 암호시스템과 안전성 정의

1. 브로드캐스트 암호시스템

양의 정수 n은 전체 사용자 수라고 하고, N을 전체 사용자 집합이라고 하자. 브로드캐스트 암호시스템에 속한 사용자는 1부터 n 가운데 하나의 수로 특정된다고 하자. 이 경우 $N=\{1,...,n\}$이다. R은 N의 부분집합으로서 복호화를 할 수 없는 탈퇴자들의 집합이라 하자. 이 경우 $R \subseteq N$ 이다. 브로드캐스트 암호의 목표는 전송 메시지 M을 R에 속하는 탈퇴자는 복호화할 수 없도록 하고, N에서 R을 제외한 집합인 $N \setminus R$에 속하는 사용자는 복호화할 수 있도록 하는 것이다.

브로드캐스트 암호시스템은 다음의 세 가지 알고리즘으로 구성되어 있다. (1) 초기설정(Setup) 알고리즘은 시스템 파라미터와 전체 사용자 수 n을 입력받은 후, 시스템에 속

하는 사용자 각각에게 복호화 키를 할당한다. (2) 암호화(Encryption) 알고리즘은 집합 R과 메시지 M을 입력받은 후, R에 속한 탈퇴자들이 복호화하지 못하도록 암호문을 생성한다. (3) 복호화(Decryption) 알고리즘은 사용자의 비밀키와 암호문을 입력받은 후, 사용자가 R에 속하지 않는다면 암호문을 복호화하여 메시지 M을 출력한다.

2. Subset Cover 방식

Subset Cover 방식은 먼저 전체 사용자 집합 N에서 탈퇴자 집합 R을 제외한 집합, 즉 $N \setminus R$, 을 부분집합의 집합으로 표현한다. 이 경우 $N \setminus R$은 서로 겹치지 않은 Subset들의 집합인 $S_{i_1},...,S_{i_m}$으로 분리한다. 즉 $N \setminus R = \bigcup_{j=1}^{m} S_{i_j}$ 가 된다. 각각의 부분집합 S_j들은 그에 대응하는 그룹키 L_j를 할당하고, S_j안의 수신자는 L_j를 유도해 낼 수 있다. 그리고 메시지를 암호화하기 위한 세션키 K는 $S_{i_1},...,S_{i_m}$ 들에 대응하는 그룹키 $L_{i_1},...,L_{i_m}$ 들로 암호화를 하고, 실제 전송하고자 하는 메시지는 K를 이용하여 암호화한다.

구체적으로 설명하면 Subset Cover 방식 하에서는 두 가지 종류의 암호 기법을 사용한다.

(1) $F_k : \{0,1\}^* \mapsto \{0,1\}^*$ 로 전송 메시지 M을 세션키 K로 암호화하는 기법
(2) $E_L : \{0,1\}^l \mapsto \{0,1\}^l$ 로 세션키 K를 분할된 subset에 대응하는 그룹키로 암호화하는 기법

이러한 Subset Cover 방식에 기반한 브로드캐스트 암호시스템은 다음의 세 알고리즘으로 설명할 수 있다.

2.1 초기설정(Setup)

모든 수신자 u는 비밀키 정보 I_u를 할당 받는다. S_i에 속한 모든 수신자 u는 자신의 비밀키 I_u를 이용하여 S_i에 대응되는 그룹키 L_i를 유도할 수 있다. 여기서 그룹키 L_i는 각각의 그룹 별로 독립적인 랜덤 값으로 할당하거나, 유사난수함수를 이용하여 할당하는 등 비독립적으로 할당할 수도 있다.

2.2 암호화(Encryption)

암호화 알고리즘은 집합 R과 메시지 M을 입력받은 후, R에 속한 탈퇴자들이 복호화하지 못하도록 다음과 같이 암호문을 생성한다.

(1) 메시지 암호화용 세션키 K를 선택한다.
(2) 주어진 탈퇴자 집합 R에 대해 $N \setminus R$을 Subset Cover 방식을 이용하여 서로 겹치지 않은 Subset $S_{i_1},...,S_{i_m}$로 분할하고, 각각의 부분집합에 대응하는 그룹키 $L_{i_1},...,L_{i_m}$를 구한다.
(3) 메시지 암호화용 세션 키 K와 $L_{i_1},...,L_{i_m}$를 이용하여 메시지 M을 다음과 같이 브로드캐스트 암호문 CT을 구성한다.

$$CT = < [i_1, i_2, ..., i_m, E_{L_{i_1}}(K), E_{L_{i_2}}(K), ..., E_{L_{i_m}}(K)], F_K(M) >$$

대괄호 []로 쌓인 부분을 헤더(header)라 하고, $F_K(M)$을 body라고 부른다.

2.3 복호화(Decryption)

수신자 u는 브로드캐스트 암호문 $< [i_1, i_2, ..., i_m, C_1, C_2, ..., C_m], C >$을 수신하면, 자신의 비밀키 I_u를 이용하여 복호화 절차를 실행한다.

(1) $u \in S_{i_j}$가 되는 i_j를 찾는다. $u \in R$의 경우에는 적당한 i_j를 찾을 수 없다.
(2) I_u로부터 대응하는 그룹키 L_{i_j}을 유도한다.
(3) $D_{L_{i_j}}(C_j)$로 복호화하여 세션키 K를 얻는다.
(4) $D_k(C)$로 복호화하여 최종 메시지 M을 얻을 수 있다.

3. 브로드캐스트 암호시스템의 안정성

기본적으로 브로드캐스트 암호시스템은 집합 R에 속한 모든 탈퇴자들이 자신들의 비밀키를 이용하여 공모공격을

하는 경우에 대처할 수 있어야 한다. 또한 탈퇴자들이 평문에 대응하는 암호문을 수집할 수 있는 능력뿐만 아니라, 암호문에 대응하는 평문을 수집할 수 있는 능력까지 고려할 수 있다. 공격자의 능력이나 안정성(security) 목적에 따라 브로드캐스트 암호시스템의 안전성 개념은 선택평문공격(CPA: chosen plaintext attack) 또는 선택암호문-중간공격(CCA1: chosen ciphertext *launch-time* attack) 또는 선택암호문공격(CCA: chosen ciphertxt attack)으로 정의될 수 있다.

본 논문에서는 대부분의 응용환경에서 충분히 안전하다고 간주되는 CCA1 안전성에 대해 간단히 설명한다. 물론 2-SD 기법의 안전성은 위의 세 가지 안전성 모델 중 어느 것에서도 취약하다는 것을 보일 것이다. CCA1 안전성은 브로드캐스트 암호화기법을 공격하는 공격자 A와 챌린저 B사이의 (다음과 같은) 게임으로 정의된다.

- Setup. B가 Setup(1^{λ},n) 알고리즘을 수행하여 사용자 $u(u \in U)$ 각각에 대한 비밀정보를 생성한다.

- Adversarial Action. A는 다음의 세 가지 질의를 할 수 있다. (1) A가 선택한 사용자 u'의 비밀키 $I_{u'}$를 요청한다. (2) A가 선택한 부분집합 R과 메시지 M을 B에게 보내서 대응하는 암호문을 받는다. (3) A가 선택한 암호문과 임의의 u를 B에게 보내서 u의 비밀키로 암호문을 복호화하여 얻은 메시지를 받는다.

- Challenge. A는 메시지 M^*과 탈퇴자 집합 R^*을 B에게 보낸다. 여기서 탈퇴자의 집합 R^*은 A가 비밀키를 요청한 사용자들을 모두 포함해야 한다. B는 랜덤한 bit $b \in \{0,1\}$을 선택한다. b=1인 경우에는 Encrypt(M^*, R^*)의 결과를 암호문으로서 A에게 준다. b=0인 경우는 M^*와 같은 길이를 갖는 랜덤 메시지 R_M를 선택하여 Encrypt(R_M, R^*)의 결과를 암호문으로서 A에게 준다.

- Guess. A는 추측한 $b' \in \{0,1\}$를 내보낸다.

위의 게임에서 A가 bit b를 정확하게 추측한 상황을 $CGues$로 나타내자. 안전성 상수 λ에 대해 A의 advantage는 $Adv_A(\lambda) = |2\Pr[CGues] - 1|$ 로 정의된다.

브로드캐스트 암호시스템이 다항식 시간에서 공격할 수 있는 능력을 가진 공격자 A가 가지는 $Adv_A(\lambda)$이 무시할 만한(negligible) 수준이라면, 우리는 '브로드캐스트 암호시스템이 CCA1 공격 환경에서 안전하다'라고 말한다.

III. SD 또는 2-SD 방식에 기반한 브로드캐스트 암호시스템

이해의 편의를 위해 2-SD 기법을 설명하기에 앞서 SD 기법의 핵심 아이디어를 먼저 살펴본다.

1. SD 기법

SD 기법[8]은 이진트리 구조를 이용하여 leaf 노드를 각 사용자로 하는 트리를 구성하고 권한 있는 사용자들만 복호화할 수 있도록 한다. SD 기법은 구체적으로 다음의 세 가지 알고리즘으로 구성된다.

1.1 초기설정

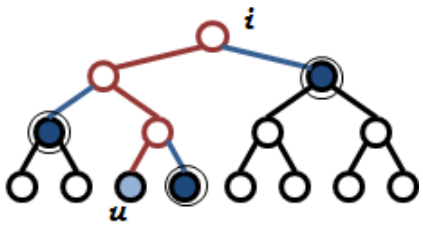


그림 1. SD기법의 비밀키 할당
Fig. 1. Key assignment in SD

비밀키 분배는 유사난수생성기(Pseudo-random generator) 함수 $G: \{0,1\}^k \rightarrow \{0,1\}^{3k}$의 일방향성을 이용하여 이루어진다. Subtree의 root에서는 k 비트의 랜덤값이 할당되고, 이 값을 G의 입력으로 넣어 $3k$ 비트의 출력이 발생한다. $3k$ 비트를 3등분하여 왼쪽부분을 왼쪽 자식 노드에 할당되는 레이블 값으로, 가운데 부분을 해당 노드의 키값으로,

오른쪽부분은 오른쪽 자식 노드에 할당되는 레이블 값으로 하여 각각의 leaf 노드까지 키 분배가 이루어지도록 한다. 각 사용자가 보유하는 키는 Fig.1을 예로 들어 설명하면 다음과 같다. 사용자 u의 경우 Subtree의 root인 i노드부터 각 사용자 u노드에 이르는 경로는 빨간색으로 나타난다. 이 빨간색 경로와 만나지 않는 노드 중 빨간색 경로 상에서 파생되는 첫 번째 자식 노드들 - Fig.1에서 이중 원으로 그려진 노드들 - 이 결정되면, Subtree의 root인 i노드에서 G를 이용하여 할당한 값들 중 해당 노드들에 대응되는 레이블을 키값으로 준다. 이러한 작업은 사용자 u노드에 이르는 경로 상의 모든 노드들에 대해 시행된다. 사용자 u가 저장해야 되는 비밀키의 총 저장량은 각 Subtree별로 높이 d만큼 저장해야하고 각 leaf 노드가 포함될 수 있는 Subtree의 높이는 1부터 $\log(n)$ 까지 된다. 또한 탈퇴자가 없는 경우에 대응되는 키 1개를 포함하여 전체 비밀키 저장량은 다음과 같다.

$$1+\sum_{d=1}^{\log(n)} d=\frac{1}{2}log^2(n)+\frac{1}{2}log(n)+1$$

1.2 암호화

먼저 전체 트리에서 권한이 있는 사용자들만 포함하고 서로 겹치지 않는 Subset들의 집합들 $S_{i_1,j_1},S_{i_2,j_2},....,S_{i_m,j_m}$으로 분할한다. 여기서 $S_{i,j}$는 i를 root로 하는 Subtree에서 j노드를 root로 하는 Subtree에 속한 사용자들이 탈퇴되는 집합을 의미한다. 각각의 $S_{i,j}$에는 i노드에 대응되는 랜덤값으로 시작하여 G를 이용하여 j노드에 대응되는 키값이 존재한다. 이를 $K_{i,j}$라 하자.

메시지 암호화용 랜덤 키 K를 선택하고, 암호화하고자 하는 메시지 M은 $E_K(M)$으로 암호화한다. 그리고 K는 Subset들에 대응되는 키 $K_{i,j}$로 암호화한다. 이렇게 하여 얻어지는 전체 암호문의 형태는 다음과 같다.

$$<[(i_1,j_1),...,(i_m,j_m),E_{K_{i_1,j_1}}(K),...,E_{K_{i_m,j_m}}(K)],E_K(M)>$$

여기서 암호문 헤더의 길이는 K를 각각의 Subset에 해당하는 그룹키로 암호화하는 것에 의해 결정된다. 따라서 Subset의 개수가 메시지의 길이를 결정하게 된다. 기존 SD 기법에서는 Subset의 개수가 탈퇴자 수 r에 대해 최악의 경우 $2r-1$로 될 수 있음을 보이고 있다.

1.3 복호화

사용자 u는 메시지를 수신한 후 Subset을 나타내는 인덱스 정보 $(i_1,j_1),...,(i_m,j_m)$ 를 이용하여 자신이 Subset $S_{i_1,j_1},S_{i_2,j_2},....,S_{i_m,j_m}$들 중 어느 부분집합에 속하는 지를 결정한다. 예를 들어, $S_{i,j}$안에 u가 속한다고 하면, i노드는 사용자 u에 이르는 경로 상에 있는 하나의 노드이면서 동시에 j노드는 사용자 u에 이르는 경로 상에 존재하지 않아야 한다. 일단 $S_{i,j}$가 결정되면, u는 $S_{i,j}$에 대응하는 그룹키 $K_{i,j}$를 생성하여 해당되는 암호문 $E_{K_{i,j}}(K)$를 복호화할 수 있고, K를 이용하여 $E_K(M)$를 복호화할 수 있다.

$S_{i,j}$에 속한 (탈퇴자가 아닌) 사용자는 i노드를 root로 두고 할당된 키값 중 하나를 이용하여 j노드에 해당하는 $K_{i,j}$를 구할 수 있다. 이 경우 함수 G를 최대 $\log(n)$번 호출하여 $K_{i,j}$를 알아낼 수 있다. 반면 j노드 아래에 있는 탈퇴자들은 저장하고 있는 키값을 이용하여 $K_{i,j}$를 알아낼 수 없다.

2. 2-SD 기법

2-SD 기법은 기본적으로 초기설정, 즉 비밀키 분배과정이 SD 기법과 동일하므로, 여기서는 암호화와 복호화 과정만 설명하기로 한다.

2.1 암호화

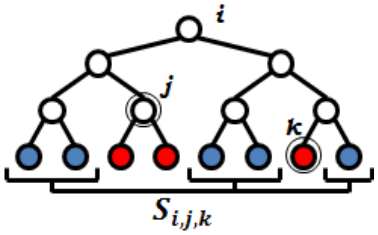


그림 2 서브셋 $S_{i,j,k}$
Fig. 2 Subset $S_{i,j,k}$

2-SD 기법은 SD 기법과 마찬가지로 전체 이진트리에서

사용자들만을 포함하면서도 서로 겹치지 않는 Subset들로 분할한다. 그리고 각 Subset에 대응되는 그룹키로 메시지 암호화용 키를 암호화한다. 차이점은 Subset의 형태가 SD 기법과 다르다는 것이다. 기존 SD 기법과 같은 형태의 Subset $S_{i,j}$이외에, Subset $S_{i,j,k}$의 형태로 분할할 수 있다. 여기서 $S_{i,j,k}$는 i노드의 후손들 중 j노드와 k노드의 후손을 배재하는 형태의 Subset을 의미한다. Fig.2에서 보면, j노드와 k노드의 최소공통조상은 i노드가 된다. 이 경우 Subset $S_{i,j,k}$에 대응되는 그룹 키는 i를 root로 하는 Subtree에서 j노드에 해당하는 그룹키 $K_{i,j}$와 k노드에 해당하는 그룹키 $K_{i,k}$를 XOR 연산한 값 $K_{i,j}\oplus K_{i,k}=K_{i,j,k}$이 된다. 2-SD 기법에 의하면 하나의 Subset으로 탈퇴자 집합 두 부분을 수용할 수 있기 때문에 기존 SD 기법보다 Subset의 개수가 줄어들 수 있게 된다. 최악의 경우에도 탈퇴자수 r 명에 대해 r개의 Subset을 설정할 수 있다. 그 결과 (2-SD 기법이 안전하다면) 암호문 헤더의 전송량 측면에서 기존 SD 기법의 $2r-1$ 보다 훨씬 더 좋은 개선을 보일 수 있게 된다.

2.2 복호화

2-SD 기법에서도 사용자는 자신이 속하는 Subset을 찾아서 해당 Subset에 대응하는 그룹키로 메시지 암호화용 키 K를 먼저 복구하고, K를 이용하여 메시지를 복구한다. 기존 SD 기법과 같은 형태의 Subset $S_{i,j}$일 경우 SD 기법과 같은 방법으로 복호화를 하고, 2-SD기법에서 추가된 Subset 형태인 $S_{i,j,k}$ 경우에는 해당하는 그룹 키를 $K_{i,j}\oplus K_{i,k}=K_{i,j,k}$로 구한다. $K_{i,j,k}$를 구하기 위해서는 $K_{i,j}$와 $K_{i,k}$를 구해야 하는데, (공모공격을 고려하지 않는다면) j노드의 후손은 $K_{i,j}$를 알 수 없어서 $K_{i,j,k}$를 유도할 수 없고 k노드 후손은 $K_{i,k}$를 알 수 없어서 마찬가지로 $K_{i,j,k}$를 알아낼 수 없다. 결과적으로 탈퇴자는 $K_{i,j,k}$를 알 수 없고 탈퇴자가 아닌 사용자들만이 복호화가 가능하게 된다. 계산량 측면에서는 사용자가 보유한 키로 $K_{i,j}$와 $K_{i,k}$, 두 개의 키 값을 유도해야 하므로 최악의 경우 SD 기법의 2배인 $2\log(n)$의 함수 G 호출이 필요하다.

Ⅳ. 2-SD 기법에 대한 공모 공격

효율성 측면에서는 2-SD 기법이 SD 기법에 비해 복호화를 위한 계산량은 약간 증가하였으나 전송량을 거의 절반으로 줄일 수 있는 획기적인 장점을 보였다. 안전성 측면에서는 SD 기법이 CCA1 안전성 모델 하에서 (공모공격에 대해) 안전하다는 것이 증명된 반면, 2-SD 기법은 그에 대한 안전성 증명이 전혀 제시되지 않았다. 이 절에서는 2-SD 기법이 공모공격을 기본전제로 하고 있는 CPA, CCA1, 또는 CCA 안전성 모델에서 전혀 안전하지 않음을 보인다. 여기서 공모공격이라 함은 공격자가 선택한 (즉, 탈퇴된) 사용자들의 비밀키들을 모두 이용할 수 있다는 것을 기본적으로 가정하고 있으며, 이를 이용하여 암호문에 숨겨진 메시지의 정보를 파악하고자 하는 것이다.

2-SD 기법에서는 $S_{i,j,k}$ 형태의 Subset을 설정할 수 있는데, 이 경우 $S_{i,j,k}$에 대응하는 그룹 키는 $K_{i,j}\oplus K_{i,k}=K_{i,j,k}$임으로 $K_{i,j}$와 $K_{i,k}$, 두 개의 그룹 키를 구할 수 있어야 한다. 앞서 설명한 바와 같이 [11]에서는 공모공격이 이루어지지 않으면 j노드 후손들은 자신들만의 비밀키를 이용해서는 $K_{i,k}$를 알 수 있으나 $K_{i,j}$를 알 수 없다. 또한, k노드의 후손들은 자신들만의 비밀키를 이용해서는 $K_{i,j}$를 알 수 있으나 $K_{i,k}$을 알 수 없다. 따라서 공모공격이 이루어지지 않는다면 j노드와 k노드 아래의 후손들은 그룹 키 $K_{i,j,k}$를 구성하는 $K_{i,j}$와 $K_{i,k}$의 부분만 알게 되므로 $K_{i,j,k}$를 유도 할 수 없다. 그러나 브로드캐스트 암호시스템의 안전성 모델에서 가장 기본적인 공격인 공모공격이 이루어지면, 탈퇴자들은 자신들의 비밀키를 서로 공유할 수 있으므로 j노드 후손들과 k노드의 후손들은 자신들의 가진 모든 비밀키를 공유하게 된다. 당연히 공유된 비밀키들로부터 얻어지는 추가적인 정보까지 공유하게 된다. 먼저 j노드 아래의 후손들은 자신들의 비밀키를 이용하여 $K_{i,k}$를 구할 수 있고, k노드의 아래의 후손들은 자신들의 비밀키를 이용하여 $K_{i,j}$를 구할 수 있다. 그리고 j노드 아래의 탈퇴자와 k노드 아래의 탈퇴자가 공모하는 상황을 가정하면, 두 노드의 후손들(즉, 탈퇴자들)은 $K_{i,k}$과 $K_{i,j}$를 서로 공유하게 된다. 결국 그룹 키 $K_{i,j,k}$를 구성하는데 필요한 두 개의 부분적인 키 $K_{i,j}$와 $K_{i,k}$

를 모두 알 수 있게 된다. $K_{i,j} \oplus K_{i,k} = K_{i,j,k}$이므로 $S_{i,j,k}$에 대응하는 그룹 키 $K_{i,j,k}$를 알게 되고, 이를 이용하여 j노드 아래의 탈퇴자와 k노드 아래의 탈퇴자들은 메시지 암호화용 세션키를 복구하게 된다. 이러한 공모공격은 j노드 아래의 탈퇴자 한 명과 k노드 아래의 탈퇴자 한 명, 즉 단 두 명의 공모자가 생기면 가능하게 된다. 브로드캐스트 암호시스템의 CPA, CCA1, 또는 CCA 안전성 모델에서 공격자 A가 탈퇴자 집합 R에 속한 사용자의 비밀키 정보를 모두 얻을 수 있다고 가정하기 때문에, 2-SD 기법은 위의 세 모델 하에서 공히 안전성이 보장되지 않는다. 실제적으로도 이러한 공모공격이 가능한 형태는 $S_{i,j,k}$ 형태의 모든 Subset에서 일어날 수 있으므로, 임의의 공격자는 공모공격이 가능한 탈퇴자 두 명을 찾아내기만 하면 된다.

Ⅴ. 결 론

2-SD 기법이 안전하다면, 기존 SD 기법에 비해 저장량과 복호화 연산량은 비슷하게 유지하면서 전송량을 절반으로 줄일 수 있는 획기적인 기법이 될 수 있었다. 그러나 본 논문에서는 2-SD 기법이 그 우수한 장점에도 불구하고 브로드캐스트 암호시스템 안전성의 기본 공격인 공모공격에 전혀 안전하지 않다는 것을 발견하였다.

SD 기법을 3진트리에서 일반적인 형태로 설계한 브로드캐스트 암호시스템 기법들[5], [12]들이 제시되었다. 3진트리의 SD 기법에서는 3진트리의 (동일한 레벨의) 세 자손들 중 두 개 이하의 자손 노드들이 모두 탈퇴했을 경우, SD 기법처럼 부분집합으로 분할이 가능하고 부분집합별 그룹 키를 부여할 수 있는 방법이다. 일견 2-SD 기법과 유사하게 보이나 양자는 매우 큰 차이가 있다. 3진트리의 SD 기법에서는 오직 같은 레벨에서만 두 개 이하의 자손 노드들을 배제하는 것이 가능한데 비해, 2-SD 기법에서는 서로 다른 레벨에 위치한 두 개 이하의 자손 노드들을 자유롭게 배제하는 것이 가능하다. 따라서 3진트리의 SD 기법이 여전히 2-SD 기법에 비해 제한적인 형태로 부분집합을 분할할 수 있게 된다.

2-SD 기법처럼 서로 다른 레벨의 노드들에 대해 부분집합을 분할하려면 기존 기법과는 다른 비밀키 분배방식이 필요할 것으로 예상된다. SD 기법과 그 변형된 기법들[5], [12]은 모두 유사난수생성기의 일방향성을 이용하여 사용자의 비밀키를 생성한 관계로, 현재까지는 그 확장성에 어려움을 겪고 있다. 한 가지 새로운 비밀키 분배 방식으로는 $(2, n)$-비밀분산(secret sharing) 기법을 이용하여 사용자의 비밀키 값에 간단하지만 대수적인 구조를 적용하는 것이다. 이후의 연구는 비밀분산 기법으로 SD 기법을 설계하는 것과, 이를 일반화하여 2-SD 기법을 고안하는 것이 될 것이다.

참 고 문 헌 (References)

[1] A. Fiat and M. Naor, "Broadcast encryption," Proceedings of the CRYPTO'93, volume 773 of LNCS, pp. 480-491, Aug. 1993.

[2] D. Naor, M. Naor and J. Lotspiech, "Revocation and tracing schemes for stateless receivers," Proceedings of the CRYPTO 2001, vol. 2139 of LNCS, pp. 41-62, Feb. 2001.

[3] D. Halevy and A. Shamir, "The LSD broadcast encryption scheme," Proceedings of the CRYPTO 2002, vol. 2442 of LNCS, pp. 47-60, Aug. 2002.

[4] M.T. Goodrich, J.Z. Sun and R. Tamassia, "Efficient tree-based revocation in groups of low-state devices," Proceedings of the CRYPTO 2004, vol. 3152 of LNCS, pp. 511-527, Aug. 2004.

[5] S. Bhattacherjee and P. Sarkar, “Tree based symmetric key broadcast encryption”, IACR Cryptology ePrint Archive, Report 2013/786, 2013.

[6] B. Chor, A. Fiat, and M. Naor, "Tracing traitors," Proceedings of the CRYPTO'94, vol. 839 of LNCS, pp. 257-270, Aug. 1994.

[7] Y. Dodis and N. Fazio, "Public key broadcast encryption for stateless receivers," Proceedings of the Digital Rights Management Workshop, vol. 2696 of Lecture Notes in Computer Science, pp. 61-80, 2002.

[8] ChongHee Kim, YongHo Hwang and PilJoong Lee, "An efficient public key trace and revoke scheme secure against adaptive chosen ciphertext attack," Proceedings of the ASIACRYPT 2003, vol. 2894 of LNCS, pp. 359-373, Nov/Dec. 2003.

[9] D. Boneh, C. Gentry and B. Waters, "Collusion resistant broadcast encryption with short ciphertexts and private keys," Proceedings of the CRYPTO 2005, vol. 3621 of LNCS, pp. 258-275, Aug.2005.

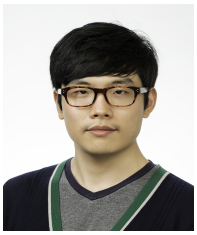
[10] D. Boneh and B. Waters, "A fully collusion resistant broadcast, trace, and revoke system," Proceedings of the ACM CCS 06, pp. 211-220, Oct/Nov. 2006.

[11] JiYong Jang, DaeHun Nyang, and JooSeok Song, “2-Subset Difference Scheme for Broadcast Encryption,” Journal of the Korea

Institute of Information Security and Cryptology, 16(4), pp. 1-5, Aug. 2006.

[12] K. Fukushima, S. Kiyomoto, Y. Miyake and K. Sakurai, "Revocation and tracing based on ternary tree: towards optimal broadcast encryption scheme," Proceedings of the IECTE 2011, vol. 314 of CCIS, pp. 233-248, 2012.

저 자 소 개



이 재 환

- 2009년 3월 ~ 현재 : 상명대학교 소프트웨어대학 컴퓨터과학과 학사과정
- 주관심분야 : 브로드캐스트 암호, 전자서명 등

박 종 환

- 1999년 2월 : 고려대학교 이과대학 수학과 (학사)
- 2004년 2월 : 고려대학교 정보보호대학원 정보보호학과 (석사)
- 2008년 8월 : 고려대학교 정보경영공학전문대학원 정보보호학과 (박사)
- 2009년 6월 ~ 2011년 5월 : 경희대학교(국제) 응용과학대학 학술연구교수
- 2011년 6월 ~ 2013년 8월 : 고려대학교 BK21정보보호사업단 연구교수
- 2013년 9월 ~ 현재 : 상명대학교 컴퓨터과학과 조교수
- 주관심분야 : 인증암호, ID-based 암호, 브로드캐스트 암호, 전자서명 등