情報保護學會論文誌
第11券 第5號, 2001. 10

무제한 사용자 탈퇴를 제공하는 효율적인 공모자 추적 기법

김 현 정*, 임 종 인**, 이 동 훈**

Efficient Public-Key Traitor Tracing with Unlimited Revocation Capability

Hyun-Jeong Kim*, Jong-In Lim**, Dong-Hoon Lee**

요 약

브로드캐스트 암호화 기법에는 두 가지 중요한 요구사항이 있다. 하나는 공모자 추적 가능성이며 다른 하나는 추적된 공모자의 탈퇴 가능성이다. 이 논문에서는 공모자의 권한을 박탈하는데 있어서 그 인원수가 제한되지 않는 새로운 형태의 공모자 추적 기법을 제안하고자 한다. 더불어 좀 더 효율적이고 간단한 자기 방지 기법에 대해 소개한다. 또한 제안한 암호화 기법을 변형하여 적응-선택 암호문 공격에 안전한 방법을 구현한다.

ABSTRACT

Two important requirements in broadcast encryption schemes are traitor traceability and revocability. In this paper, we propose a new type of a traitor tracing scheme that can revoke an unlimited number of traitors' personal keys. Additionally, we propose an efficient and simple method to provide self-enforcement property. We also describe a variant of our scheme of which encryption algorithm is secure against adaptive chosen ciphertext attacks.

keyword : *traitor tracing, revocation, self-enforcement*

Ⅰ. 서 론

최근 브로드캐스트 암호화 기법은 공개된 네트워크 상에서 멀티미디어, 소프트웨어, 유료TV 등의 디지털 정보들을 전송하는데 적용되고 있다. 브로드캐스트 암호화 기법에서 중요한 것은 오직 사전에 허가받은 사용자만이 디지털 정보를 얻을 수 있어야 한다는 것이다. 일반적으로 브로드캐스트 메시지는 권한블록(enabling block)과 암호블록(cipher block)으로 이루어져 있다. 암호블록은 제공할 디지털 정보를 세션키를 이용하여 대칭키 방식으로 암호화한 메시지를 포함하고 있고 권한블록은 해당 세션키의 암호문이 포함되어 있다. 브로드캐스트 메시지가 전달되면 권한이 있는 사용자들은 자신이 사전에 부여받은 개인키를 이용하여 먼저 세션키를 복호화하고 이 세션키를 통하여 디지털 정보를 얻게 된다.

브로드캐스트 암호화 기법에 있어서 중요한 사항 한가지는 공모자 추적이다. 이 기법은 Chor와 그의 동료들[3]에 의해 제안된 것으로 권한을 부여받은 정당한 사용자 중 일부가 공모하여 브로드캐스트되는 디지털 정보를 복호화 할 수 있는 불법 디코더를 제작하는 경우 이 공모자들을 추적해내는 방법이다. 또 하나 중요한 사항은 권한 박탈이다. 공모자를 추적해낸 후 이 공모자들의 권한을 효과적으로 제거하

* 고려대학교 정보보호연구센터(CIST)(khj@cist.korea.ac.kr)
** 고려대학교 정보보호연구센터(CIST)({jilim,donghlee}@tiger.korea.ac.kr)

는 것이 필요하다.

이전의 공모자 추적 기법에서는 권한블록의 길이가 일정하지 않고 사용자 수 또는 추적 가능한 공모자의 수, 권한 박탈이 가능한 인원수 등에 의존하였다. [3,10,5]에서 제안된 기법에서는 권한블록 길이가 사용자 수와 추적 가능한 공모자 수에 기반하고 있다. 그러나 [2]에서 공개키 방식의 공모자 추적 기법을 제안함으로써 이전의 결과를 향상시켰다. 공개키 방식에서는 세션키를 암호화하기 위한 그룹의 암호화키는 하나이고 이를 복호화하는 키는 무수히 많이 존재함으로써 데이터 제공자(*DS*)는 하나의 키로 세션키를 암호화하고 각 사용자들은 서로 다른 개인키를 이용하여 이를 복호화 할 수 있도록 되어 있다. 따라서 [2,7,13]과 같은 공개키 방식의 기법에서는 권한블록의 크기는 사용자 수에는 독립적이고 추적 가능한 공모자 수에만 의존하게 되었다.

또한 [7,13]에서는 Shamir 다항식 기반의 비밀 정보 분할 기법[11]을 사용하여 세션키를 암호화하는 [1]의 기법을 이용한 효율적인 권한 박탈 기법을 제안하였다. 이 기법에서는 k명의 공모자까지 추적이 가능하며 사용하는 다항식 차수인 $z(\geq 2k-1)$명까지는 완전한 권한 박탈이 가능하다. [7,13]에서 제안된 기법에서 권한블록의 길이와 권한 박탈 가능한 사용자 수는 차수 z에 기반한다.

실생활에서 사용자 그룹은 유동적이다. 사용자들의 권한 기간이 만료될 수도 있고, 사용자 스스로 권한을 포기하고자 하는 경우도 있다. 또한 사용자 중 일부는 불법 사용자로 발견되어질 수도 있을 것이다. 이런 경우 데이터 제공자는 이들에게 부여했던 권한을 효율적으로 제거할 수 있어야 할 뿐 아니라 권한 박탈 인원수에 대한 제한도 없어야 할 것이다. 지금까지 이러한 조건을 모두 만족하는 기법이 제안되어 있지 않았다.

본 논문에서 새로운 형태의 공모자 추적 기법을 제안한다. 제안하는 기법에서 브로드캐스트 메시지는 권한블록, 갱신블록(renewal block), 암호블록 세 가지로 구성된다. 이전의 기법에서는 권한블록이 암호화된 세션키 전달 뿐 아니라 공모자 추적의 목적으로도 사용되었다. 이로 인하여 실제로 브로드캐스트 메시지가 암호화된 세션키 정보와 암호화된 실제 디지털 정보만을 전달하면 되는 경우에도 권한블록 길이와 계산량은 추적 가능한 공모자 수에 의존할 수밖에 없었다. 새롭게 제안하는 기법에서 권한블록은 오직 암호화된 세션키 전달에만 사용되고 공모자 추적 및 권한 박탈 과정을 위해서는 갱신블록이 사용된다. 갱신블록은 평소에는 사용되지 않으며 이 경우 체크 비트 $\xi=0$로 하고 그룹키 갱신, 공모자 추적, 권한 박탈 등의 과정이 진행되어야 하는 경우에만 갱신블록을 이용하며 이때 체크 비트는 $\xi=1$로 한다. 갱신블록의 길이가 기존에 제안된 기법에서 권한블록의 길이와 마찬가지로 추적 가능한 공모자의 수에 의존하게 되지만 갱신블록을 항상 사용하는 것이 아니며 제안하는 기법에서 권한블록의 길이는 공모자 수에 독립적으로 일정하게 유지된다. 따라서 제안하는 기법이 훨씬 효율적이며 갱신블록을 별도로 구분함으로써 제한없는 권한 박탈 기법을 제공한다는 점에서 좀 더 실용적이다.

더불어 좀 더 간단하고 효율적인 자기 방지(self-enforcement) 기법을 제안한다. 자기 방지의 개념은 사용자가 소유하는 개인키에 사용자의 중요한 정보를 포함시키는 것이다[4]. 따라서 만일 사용자가 자신의 개인키를 유출하면 동시에 신용카드 번호나 계좌 번호 등과 같은 자신의 중요한 정보가 함께 유출되도록 함으로써 사용자 스스로 자신의 개인키를 불법 디코더 제작자에게 유출할 수 없도록 하고자 하는 것이다. 여기서 제안하는 기법에서는 사용자의 중요한 정보를 직접 개인키에 포함시키는 것이 아니라 개인키를 사용자의 중요한 정보를 암호화/복호화 하는데 사용하고자 한다. 만일 직접 개인키에 사용자의 정보를 포함시키는 경우 그 정보가 변경될 때 개인키도 따라서 변경되어야 하는 문제가 발생할 수 있는데 제안하는 기법에서는 이런 경우 유연하게 대처할 수 있게 된다.

논문에서 새롭게 제안하는 기법에서 브로드캐스트 암호화 알고리즘은 권한블록과 암호블록만을 사용하고 갱신블록은 그룹키 갱신, 공모자 추적 및 권한박탈 알고리즘에서 사용된다. 따라서 제안하는 기법에 관하여 브로트캐스트 암호화 알고리즘과 그 외의 알고리즘을 구분하여 설명하고자 한다. 이 논문의 구성은 다음과 같다.

먼저 2장에서 제안 기법에 관해 간략히 설명하고 기존 기법과 비교를 통한 효율성을 보인다. 3장에서 브로드캐스트 암호화 알고리즘을 제안하고 그에 관한 안전성을 증명한 후 그룹키 갱신, 공모자 추적 및 권한 박탈 알고리즘을 구현한다. 4장에서는 새롭게 제안하는 자기 방지 기법에 관해 설명하고 5장에서 제안하는 암호화 기법을 변형하여 적응-선택 암호문 공격에 안전한 형태를 구현한다.

Ⅱ. 제안하는 기법에 관한 개요 및 기존 기법과의 비교

이 장에서는 새로운 기법을 구체적으로 제안하기에 앞서 제안하고자 하는 기법에 관해 간략히 설명하고 기존에 제시된 기법들과의 비교, 분석을 통하여 새로운 기법의 효율성을 먼저 보이고자 한다.

2.1 제안하는 기법에 관한 개요

제안하는 기법은 데이터 제공자와 n명의 사용자로 구성되어 있다. 각 사용자들은 권한블록에 포함되는 세션키를 복호화 하는데 필요한 개인키와 갱신블록에 포함되는 갱신정보를 복호화하기 위한 갱신키를 소유하게 된다. 평상시 *DS*는 갱신블록은 사용하지 않는다. 오직 그룹키 갱신, 공모자 추적, 권한 박탈 과정이 필요한 경우에만 갱신블록을 사용한다. 제안된 기법에서 갱신블록을 위한 알고리즘은 [13]에서 제안된 방법을 이용하였다. 제안하는 기법은 다음과 같은 과정으로 구성되어 있다.

[초기구성]

시스템 매개변수를 구성하는 알고리즘이다. 이 알고리즘에서 *DS*는 먼저 참여하게 될 사용자의 인원수를 예측하고 그에 따라 세션키를 암호화하기 위한 그룹키를 생성한다. 향후 사용자의 수가 예측한 인원을 초과하는 경우 그룹키 갱신 과정을 수행하면 된다.

[등 록]

*DS*와 디지털 정보에 대한 권한을 원하는 신규 가입자 사이에 수행되는 프로토콜이다. 이 과정을 통하여 신규 가입자는 개인키와 갱신키를 *DS*로부터 받게 된다.

[브로드캐스트 암호화]

브로드캐스트 암호화 메시지를 생성하는 알고리즘이다. 이 암호화 알고리즘을 통하여 생성된 암호문에 대해 어떤 공모 집단도 불법적으로 복호화 키를 생성해 내는 것이 쉽지 않다. 즉, 불법 디코더 내에 포함되는 복호화 키는 *DS*가 각 사용자에게 분배한 개인키 중 하나일 수 밖에 없다. 암호화 알고리즘의 안전성은 RSA 가정과 Strong RSA 가정에 기반한다. 또한 Diffie-Hellman Decision 문제의 어려움에 기반하여 수동적인 공격자에 대해 의미론적 안전성(semantic security)을 제공한다.

[복호화]

각 사용자는 자신의 개인키를 이용한 복호화 과정을 통하여 브로드캐스트 메시지로부터 디지털 데이터를 얻는다.

[그룹키 갱신]

*DS*는 갱신블록을 이용하여 그룹키 변경과 관련된 정보를 브로드캐스트하고 각 사용자는 갱신키를 이용한 복호화 과정을 통하여 갱신정보를 얻어 이를 이용하여 자신의 개인키를 변경한다. 각 사용자가 개인키를 변경하는데 필요한 연산량은 두 번의 모듈러 곱이며 이 알고리즘은 빈번이 수행될 필요는 없다.

[공모자 추적 및 권한 박탈]

이 알고리즘은 그룹키 갱신 알고리즘과 동일하다. *DS*가 불법 디코더를 발견하면 갱신블록을 이용한 브로드캐스트 메시지를 디코더에 입력하여 그룹키 갱신이 진행되도록 한다. 불법 디코더는 이것이 그룹키 갱신을 위한 것인지 공모자 추적을 위한 것인지 구별 할 수 없다. 이렇게 추적된 공모자의 권한을 박탈하는 과정은 공모자들을 제외한 합법적인 사용자들만이 자신의 개인키를 변경할 수 있도록 갱신블록을 형성하여 그룹키 갱신 과정을 수행하므로써 이루어질 수 있다.

2.2 기존 기법과 새롭게 제안하는 기법과의 비교

이 절에서 기존에 제안된 다른 기법과 여기서 제안하는 새로운 기법을 비교해 보고자 한다.

Crypto'94에서 Chor와 그의 동료들(CFNP)[3]이 "Traitor Tracing Schemes"을 제안하였다. 이 기법의 복호화 과정에서는 오직 XOR 연산만이 요구되지만 개인키 길이와 권한블록 길이가 모두 사용자 수에 의존하고 있다.

Crypto'99에서는 Boneh와 Franklin(BF)[2]이 "A Public-Key Traitor Tracing Scheme"를 제안하였다. 이 기법에서는 개인키의 길이가 고정되었고 권한블록의 길이는 추적 가능한 공모자 수에 의존하게 되었다.

PKC'01에서 W. Tzeng와 Z. Tzeng(TT)[13]은 "A Public-Key Traitor Tracing Scheme

〔표 1〕 여러 기법의 비교 : m-사용자 수, k-추적 가능한 공모자 수, H-해쉬함수, p, q-소수 $|p|>1024$, $|q|>160$, $q|(p-1)$, z-Shamir 다항식의 차수, n-RSA 모듈러스 $|n|>1024$, p', q'- 소수 $|p'|>|n|$이고 $q'|(p'-1)$

기법 / 항목	CFNP open one-level	BF	TT	제안 기법
개인키	$O(k^2\log m)\times\|H\|$	$\|q\|$	$\|q\|$	$2\|n\|+\|\phi(n)\|+\|q'\|$
암호블록 길이	$O(k^4\log m)\times\|H\|$	$(2k+1)\times\|p\|$	$O(z)\times\|p\|$	$3\|n\|+\|\phi(n)\|\left(+\ O(z)\times\|p'\|\right)$
암호화 계산량	$O(k^4\log m)$ XORs	$\approx(2k+1)$ Exps. (mod p)	$O(z)$ Exps. (mod p)	1Exp +2*Mls*. (mod n) $\left(+\ O(z)\text{Exps.}\ (\text{mod } p')\right)$
복호화 계산량	$O(k^2\log m)$ XORs	$\approx(2k+1)$ Exps. $+(2k+1)$ *Mls*. (mod p)	$O(z)$Exps. $+O(z)$ *Mls*. (mod p)	2Exps. +2*Mls*. (mod n) $\left(+\ O(z)\text{Exps.} + O(z)\text{Mls.}\ (\text{mod } p')\right)$
권한박탈 인원수	-	-	$\le z$	∞

with Revocation"을 제안하여 최대 z명의 사용자까지 완전한 권한 박탈이 가능한 기법을 소개하였다.

여기서 위의 기법들과 새롭게 제안하는 기법간의 차이를 분석한다.

새롭게 제안하는 기법에 있어서 갱신블록이 사용되지 않는 경우 계산 복잡도는 표 윗 부분과 동일하며 갱신블록이 사용되는 경우에만 괄호 안의 복잡도가 추가로 합하여진다.

Ⅲ. 제안하는 기법

이 장에서는 공개키 기반의 새로운 공모자 추적 기법을 제안하고 그 안전성을 증명한다. 제안하는 기법은 브로드캐스트 암호화 알고리즘과 공모자 추적 및 권한 박탈에 관한 알고리즘으로 분리하여 설명하고자 한다.

3.1 브로드캐스트 암호화 알고리즘

먼저 브로드캐스트 암호화 알고리즘을 제안하고 알고리즘에 대한 안전성을 분석한다. 이 안전성에 기반하여 다음 절에서 공모자 추적 및 탈퇴에 관한 알고리즘을 제안할 것이다.

3.1.1 브로드캐스트 암호화 알고리즘 제안

*DS*는 사용자의 인원수를 미리 예측하고 시스템 초기 구성을 시작한다. 이후에 사용자 수가 초과되면 다음 장에서 설명하는 그룹키 갱신 알고리즘을 이용하면 된다. 미리 예측한 사용자의 수를 n이라 하자.

[초기구성]

G_1과 G_2를 의사 난수 생성기라 하고 H를 모듈러 연산에 기반한 해쉬함수[6]라 하자. *DS*는 다음과 같이 시스템 초기구성을 한다.

1. 소수 p', q'에 대해 $p=2p'+1, q=2q'+1$을 만족하는 소수 p, q를 선택하고 $n=pq$를 계산한다.
2. $(a\pm1, n)=1$인 군 $G=\langle a\rangle$를 선택한다. 이때 군 G는 위수가 $p'q'$인 순환 부분군이 되고 $\left(\frac{a^2}{n}\right)=1$이다.
3. 임의의 수 $h, t\in G$와 $x_M\in Z^*_{p'q'}$를 선택한다.
4. 초기값 s_v를 입력하여 의사 난수 생성기 G_1로부터 난수 열 $\{v_1,\cdots,v_n\}$을 생성하고 이 난수열을 입력값으로 하여 해쉬함수 H를 통해 모듈러 $p'q'$에 대한 출력값 $\{e_1,\cdots,e_n\}$을 얻는다. 이때 모든 e_i는 $e_i\in Z^*_{p'q'}$를 만족해야 하고 만약 이를 만족하지 않는 e_i의 경우 관련 인덱스를 별도로 관리한다.(*DS*는 오직 초기값 s_v만을 저장한다.)
5. $u_i^{e_i}\equiv h^{x_M}\ (\text{mod } n)$를 만족하는 $\{u_1,\cdots,u_n\}$을 계산한다. 즉, $u_i\equiv(h^{x_M})^{1/e_i}\ (\text{mod } n)$.
6. 초기값 s_w을 입력하여 의사 난수 생성기 G_2로부터 난수 열 $\{w_1,\cdots,w_n\}$을 생성하고 이 난수열을 입력값으로 하여 해쉬함수 H를 통해 모듈러 pq에 대한 출력값 $\{a_{11}, a_{21},\cdots, a_{n1}\}$을 얻는다. 이때 모든 a_{i1}는 $a_{i1}\in Z^*_{pq}$를 만족해야 하고 만약 이를 만족하지 않는 a_{i1}의 경우 관련 인덱스를 별도로 관리한다.(*DS*는 오직 초기값 s_w만을 저장한다.)

7. 각 u_i에 대해 $a_{i2} \equiv u_i - a_{i1} \pmod{n}$를 계산한다.
8. 임의의 값 $a_{r1}, a_{r2} \in_R G$에 대해
   $A_1 \equiv a_{11}a_{21}\cdots a_{n1}a_{r1} \pmod{n}$와
   $A_2 \equiv a_{12}a_{22}\cdots a_{n2}a_{r2} \pmod{n}$를 계산한다.

DS는 (x_M, A_1, A_2)와 두 개의 초기값 (s_v, s_w)을 비밀값으로 저장하고 (h, t, n)을 공개한다. 신규 가입자가 가입을 원하는 경우 DS는 (h, x_M, A_1, A_2)와 초기값 (s_v, s_w), 의사 난수 생성기 (G_1, G_2), 해쉬함수 H를 이용하여 가입자의 개인키를 생성한다.

[등 록]

이 프로토콜을 통하여 사용자 B_i는 세션키를 복호화하기 위해 필요한 개인키를 DS로부터 받는다. 먼저 DS는 B_i의 신원을 확인한 후 개인키를 다음 과정을 통해 발급한다.

1. h^{x_M}와 초기값 (s_v, s_w), 의사 난수 생성기 (G_1, G_2), 모듈러 해쉬함수 H를 이용하여 (a_{i1}, a_{i2}, e_i)를 생성한다.
2. $\sigma_{i1} = (A_1/a_{i1})$와 $\sigma_{i2} = (A_2/a_{i2})$를 계산한다. 즉,
   $\sigma_{i1} = a_{11}a_{21}\cdots a_{(i-1)1}a_{(i+1)1}\cdots a_{n1}a_{r1}$,
   $\sigma_{i2} = a_{12}a_{22}\cdots a_{(i-1)2}a_{(i+1)2}\cdots a_{n2}a_{r2}$.
3. 개인키 $(\sigma_{i1}^{-1}, \sigma_{i2}^{-1}, e_i)$를 B_i에게 전송한다.

[권한블록 암호화]

s를 세션키라 하자. 권한블록 E는 다음과 같이 구성된다. 이때 $r, d \in_R Z^*_{p'q'}$는 원타임 랜덤 변수라 하자.

$$E = \langle sh^{dx_M}, A_1 t^{rx_M}, A_2 t^{rx_M}, t^{-rx_M d}, t^r, d \rangle$$

DS는 세션키의 암호화 메시지를 포함하는 권한블록과 이 세션키로 암호화된 디지털 데이터를 포함한 암호블록을 브로드캐스트한다.

[권한블록 복호화]

사용자 B_i가 브로드캐스트 메시지를 받으면 개인키를 이용하여 다음과 같이 권한블록으로부터 세션키를 복호화 한다.

$$s = sh^{dx_M} / \{(A_1 t^{rx_M} \cdot \sigma_{i1}^{-1} + A_2 t^{rx_M} \cdot \sigma_{i2}^{-1})^d t^{-rx_M d}\}$$

사용자들은 복호화한 세션키 s를 이용하여 암호블록의 디지털 데이터를 얻을 수 있다.

3.1.2 안전성 분석

제안한 암호화 알고리즘은 수동적 공격자에 대해 의미론적인 안전성을 제공하고 있으며 사용자가 h^{x_M}을 계산해내는 것이 쉽지 않다. 또한 어떤 공모 집단도 새로운 복호화 키 $(\gamma_1, \gamma_2, \alpha)$를 불법적으로 생성하는 것이 쉽지 않다. 안전성 증명에 앞서 Diffie-Hellman Decision 가정, RSA 가정, Strong RSA 가정에 대해 먼저 살펴본다. 이를 위해 군 G에 대해 두 집합을 다음과 같이 정의하자.

$$D := \{(g_1, g_2, y_1, y_2) \in G^4 \mid ord(g_1) = ord(g_2) = n',\ \log_{g_1} y_1 = \log_{g_2} y_2\},$$
$$R := \{(g_1, g_2, y_1, y_2) \in G^4 \mid ord(g_1) = ord(g_2) = ord(y_1) = ord(y_2) = n'\}.$$

[가정 1 (Diffie-Hellman Decision 가정)]

모든 확률적인 다항식 시간 알고리즘 A와 충분히 큰 k에 대해 다음 두 확률분포

$\Pr[a = 1 \mid G := T(1^k), K \in_R D, a := A(K)]$와

$\Pr[a = 1 \mid G := T(1^k), K \in_R R, a := A(K)]$를 계산적으로 구별해내기 어려운 확률적인 알고리즘 T가 존재한다.

[가정 2 (RSA 가정)]

모든 확률적인 다항식 시간 알고리즘 A, 모든 다항식 $p(\cdot)$, 그리고 충분히 큰 k에 대해 다음을 만족하는 확률적인 알고리즘 T가 존재한다.

$$\Pr[z = u^e \mid (G, z, e) := T(1^k), u := A(G, z, e)] < \frac{1}{p(k)}.$$

[가정 3 (Strong RSA 가정)]

모든 확률적인 다항식 시간 알고리즘 A, 모든 다항식 $p(\cdot)$, 그리고 충분히 큰 k에 대해 다음을 만족하는 확률적인 알고리즘 T가 존재한다.

$$\Pr[z = u^e \wedge e > 1 \mid (G, z) := T(1^k), (u, e) := A(G, z)] < \frac{1}{p(k)}.$$

이제 위의 가정에 기반하여 안전성을 증명하도록 한다.

[정리 1]

만일 Diffie-Hellman Decision(DHD) 문제가 어렵다면 제안하는 암호화 알고리즘은 수동적 공격자에 대해 의미론적 안전성을 제공한다.

(증명)

모순을 보이기 위해 공격자 A가 제안된 암호화 알고리즘에 대해 의미론적 안전성 측면에서 공격에 성공할 수 있다고 가정하자. 즉, 공개키 (h, t)에 대해 A는 두 개의 세션키 s_0와 s_1중 하나를 암호화한 암호문 C에 대해 어떤 세션키에 대한 암호문인지 무시할 수 없는 값 ε정도의 차이로 구별 가능하다고 하자. 이때 확률적인 다항식 시간 알고리즘 B가 존재해서 A를 이용하여 ε만큼의 차이로 DHD 문제를 풀 수 있음을 보이면 된다. (g_1, g_2, y_1, y_2)가 B에게 입력값으로 주어지면 B는 A를 이용하여 다음과 같은 과정으로 입력값이 D에 속하는지 R에 속하는지 결정한다.

1. $h=g_1$와 $t=g_2$를 A에게 전달하고 s_0와 s_1를 A로부터 받는다.
2. $r, d \in_R Z$, $b \in_R \{0,1\}$, $A_1, A_2 \in_R G$를 선택하여 세션키 s_b에 대해 암호블록 E를 다음과 같이 생성한다.

   $E = \langle s_b y_1^d, y_2^r A_1, y_2^r A_2, y_2^{-rd}, g_2^r, d \rangle$

3. E를 A에게 전달하고 A로부터 b'를 받는다. $b=b'$이면을 1을 출력한다.

만일 $(g_1, g_2, y_1, y_2) \in D$이면 $h=g_1$, $t=g_2$, $y_1=g_1^x$, $y_2=g_2^x$, $t^r=g_2^r$이다. 따라서 $y_1^d=g_1^{xd}$이고 $y_2^{rd}=(g_2^r)^{xd}$이다.

그러므로 이 경우 $\Pr(b=b')=1/2+\varepsilon$, 그렇지 않으면 $\Pr(b=b')=1/2$ 이다. 결과적으로 B는 DHD 문제를 무시할 수 없는 값 ε만큼의 차이로 구별 가능하다. □

이제 공모자들이 세션키를 복호화하기 위해 필요한 어떤 정보도 찾는 것이 어렵다는 것을 보일 것이다. 만일 공모자들이 그룹키 A_1와 A_2를 찾을 수 있다면 그들은 자신의 개인키로부터 h^{x_M}값을 계산할 수 있다. 그러나 다음 정리를 통하여 이것이 쉽지 않은 문제임을 보일 것이다. 또한 공모자들은 자신의 개인키로써 $(\sigma_{i1}^{-1}, \sigma_{i2}^{-1}, e_i)$만을 알고 있기 때문에 이를 이용하여 권한블록으로부터 h^{x_M}에 대한 정보를 얻기는 어렵다. 결국 DS 이외에는 h^{x_M}를 찾는 것이 쉽지 않다.

[정리 2]

RSA 문제가 풀기 어렵다는 가정하에 $k(\leq n)$명의 공모자가 $A_1 = a_{11} \cdots a_{n1} a_{r1}$이나 $A_2 = a_{12} \cdots a_{n2} a_{r2}$를 찾는 것은 쉽지 않다. 다시 말하면, t^{rx_M}을 찾는 것은 쉽지 않다.

(증명)

공모자 개인키 집합을 $I = \{(\sigma_{11}^{-1}, \sigma_{12}^{-1}, e_1), \cdots, (\sigma_{k1}^{-1}, \sigma_{k2}^{-1}, e_k)\}$이라 하자. 공모자들이 A_1을 안다는 것과 $\{a_{11}, \cdots, a_{k1}\}$중 적어도 하나의 값을 안다는 것은 동치이다. 따라서 공모자들이 A_1을 찾기 위해서는 다음 방정식을 풀어야 한다.

$$\begin{pmatrix} \sigma_{11} & 0 & \cdots & 0 \\ 0 & \sigma_{21} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_{k1} \end{pmatrix} \begin{pmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{21} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{k1} \end{pmatrix} = \begin{pmatrix} A_1 & 0 & \cdots & 0 \\ 0 & A_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_1 \end{pmatrix} \quad (1)$$

이 방정식 (1)의 해는 A_1에 따라서 항상 유일하게 존재함을 알 수 있다. 그러므로 정확한 해인지 판단하기 위해서는 암호블록에서 세션키를 정확히 복호화 해 낼 수 있는지 검증해 봐야만 한다. 결국 공모자들은 암호블록의 값 $A_1 t^{rx_M}$으로부터 정확한 A_1을 찾을 수 있어야 옳은 해를 구할 수 있다. 그러나 $A_1 t^{rx_M}$로부터 A_1를 찾는 것은 (t^r, t^{-rx_M}, d)로부터 t^{rx_M}을 찾는 것과 동일하며 이는 RSA 문제를 푸는 것임을 알 수 있다. □

다음의 정리에서 $k(\leq n)$명의 공모자가 공모한 어떤 공모 집단이라 할지라도 자신들의 개인키를 이용하여 새로운 개인키를 생성하는 것이 쉽지 않음을 보이고자 한다.

[정리 3]

$I = \{(\sigma_{11}^{-1}, \sigma_{12}^{-1}, e_1), \cdots, (\sigma_{k1}^{-1}, \sigma_{k2}^{-1}, e_k)\}$를 공모자의

개인키 집합이라 하자. Strong RSA 문제가 풀기 어렵다는 가정하에 $k(\leq n)$명의 공모집단에서 $\omega \notin I$인 새로운 복호화 키 ω를 생성하는 것은 쉽지 않다.

(증명)

만일 $\omega=(\gamma_1, \gamma_2, \alpha)$가 새로운 복호화 키라고 하면 이 키는 암호블록 $\langle sh^{dx_M}, A_1 t^{rx_M}, A_2 t^{rx_M}, t^{-rx_{M0}}, t^r, d\rangle$에 대해 다음 수식을 만족해야 한다.

$$\{(A_1 t^{rx_M}\gamma_1 + A_2 t^{rx_M}\gamma_2)^d t^{-rx_{M0}}\}^{\alpha} = h^{dx_M} \quad (2)$$

식 (2)로부터 $(A_1\gamma_1 + A_2\gamma_2)^{\alpha} = h^{x_M}$를 얻을 수 있다. 이로부터 새로운 복호화 키 ω를 얻는 것은 랜덤변수 r, d에 상관없이 $\beta^{\lambda} = h^{x_M}$을 만족하는 쌍 (β, λ)을 얻는 것과 동일한 것임을 알 수 있다. 이때 만일 h^{x_M}가 알려져 있다 해도 쌍 (β, λ)을 찾는 것은 Strong RSA 문제를 푸는 것과 같다. 더구나 h^{x_M}를 아는 것조차 쉽지 않다. □

정리 3으로부터 불법 디코더를 제작하기 위해서 공모자들은 자신들의 개인키를 변경하지 않고 그대로 사용해야만 함을 알 수 있다. 따라서 불법 디코더의 형태는 반드시 블랙박스 형태이어야 하고 그렇지 않으면 *DS*가 쉽게 개인키를 이용하여 공모자들을 밝혀 낼 수 있다.

3.2 그룹키 갱신, 공모자 추적 및 권한 박탈 알고리즘

이 절에서는 제안된 기법의 그룹키 갱신, 공모자 추적, 권한 박탈 알고리즘 과정을 설명한다. 이 알고리즘은 [13]에서 제안한 방법을 기반으로 하고 있다. 따라서 공모자 추적은 최대 k명까지 모든 공모자 추적이 가능하다.

먼저 그룹키 갱신 알고리즘을 설명한다. 공모자 추적과 권한 박탈 알고리즘 또한 이와 유사하다.

3.2.1 그룹키 갱신

각 사용자는 권한블록을 복호화하기 위한 개인키를 받을 뿐 아니라 갱신블록을 복호화하기 위한 갱신키도 부여받게 된다. 그룹키 갱신 과정은 그룹키 A_1와 A_2에 포함되는 랜덤변수 a_{r1}, a_{r2}을 변경하기 위한 것으로 각 사용자들은 이에 따라 자신 개인키 σ_{i1}^{-1}와 σ_{i2}^{-1}에 포함되어 있는 랜덤변수 a_{r1}, a_{r2}를 변경해야만 한다.

[초기구성]

*DS*는 브로드캐스트 암호화 기법을 위한 초기구성 과정에서 다음 단계도 함께 수행한다.

1. 큰 소수 o ($|o| \geq |n|$)를 위수로 하는 군 G_o을 선택한다.
2. 랜덤변수 $g \in G_o$를 선택한다.
3. 차수가 $z(\geq 2k-1)$인 다항식 $f(x) = \sum_{l=0}^{z} a_l x^l \pmod{o}$ 을 선택한다. 이때, $a_l \in Z_o$.
4. $\langle g, g^{a_0}, g^{f(1)}, \cdots, g^{f(z)}\rangle$를 공개하고 다항식 $f(x)$는 비밀로 유지한다.

[가 입]

*DS*는 가입자 B_i에게 갱신키 쌍 $(i, f(i))$을 보낸다. 가입자는 자신의 갱신키를 공개된 값들과 Lagrange coefficients[12]를 이용하여 검증할 수 있다.

[갱신블록 암호화]

*DS*가 그룹키의 랜덤 값 a_{r1}, a_{r2}을 각각 $a_{r'1}, a_{r'2}$으로 변경하고자 한다고 가정하자. *DS*는 체크비트를 $\xi=1$로 하고 [13]의 알고리즘을 이용하여 $\theta_1 \equiv a_{r1} a_{r'1}^{-1}$와 $\theta_2 \equiv a_{r2} a_{r'2}^{-1}$를 암호화는 갱신블록을 구성한다.

[갱신블록 복호화]

갱신블록 내에 정보가 포함된 경우(즉 $\xi=1$) 각 사용자들은 자신의 갱신키를 이용하여 갱신 메세지를 복호화 한 후, 이를 이용하여 자신의 개인키를 변경한다. 이 변경된 개인키로 권한블록의 세션키를 복호화 할 수 있다. 먼저 사용자는 [13]의 복호화 알고리즘을 이용하여 θ_1와 θ_2를 얻고 이를 이용하여 다음과 같이 개인키를 변경한다.

$$\sigma_{ib}^{-1} \cdot \theta_b = (a_{1b} \cdots a_{(i-1)b} a_{(i+1)b} \cdots a_{nb} a_{r'b})^{-1}$$

이때 $b \in \{1, 2\}$.

만일 사용자 인원수가 *DS*가 예상했던 인원을 초과하는 경우 *DS*는 의사난수 생성기 G_1, G_2와 해쉬함수 H를 이용하여 A_1와 A_2를 재생성 한다. 예를 들어 A_1와 A_2를 $(a_{11} \cdots a_{n1} a_{r1})$과 $(a_{12} \cdots a_{n2} a_{r2})$로부터 $(a_{11} \cdots a_{n1} a_{(n+1)1} a_{r'1})$과 $(a_{12} \cdots a_{n2} a_{(n+1)2} a_{r'2})$

로 변경한다고 하면 DS는 변경되는 값에 대한 $\theta_1 \equiv a_{r1}(a_{(n+1)1}a_{r1})^{-1}$와 $\theta_2 \equiv a_{r2}(a_{(n+1)2}a_{r2})^{-1}$를 암호화한 갱신블록을 생성하면 된다.

3.2.2 공모자 추적

공모자 추적은 그룹키 갱신과정에 W. G. Tzeng와 Z. J. Tzeng이 제안하는 방법[13]을 적용한다. [13]에서는 두 가지의 공모자 추적 방식을 제안하고 있다. 이 방식을 여기서 제안한 기법에 맞추어 간략히 설명한다.

[방법 1]

먼저 DS는 공모자로 추정되는 사용자 집합 $\{c_1, c_2, \cdots, c_m\}$, $(m \le k)$을 구성한다. 공모자 추적을 위하여 체크 비트를 $\xi=1$로 두고 θ_1와 θ_2, 공모자로 추정되는 사용자들의 m개 갱신키 $\{(c_1, f(c_1)), \cdots, (c_m, f(c_m))\}$와 $(z-m)$개의 새로운 쌍을 이용하여 갱신블록을 생성한다.

만일 불법 디코더가 소유하고 있는 갱신키가 $\{(c_1, f(c_1)), \cdots, (c_m, f(c_m))\}$중에 포함되어 있다면 이 디코더는 갱신블록을 정확히 복호화 할 수 없을 것이다. 따라서 불법 디코더가 주어진 브로드캐스트 메시지에 대해 정확한 디지털 정보를 출력하지 못하면 DS는 $\{c_1, c_2, \cdots, c_m\}$집합을 공모자 집단으로 확정한다.

[방법 2]

이 방법에서는 오직 불법 디코더만이 정확하게 디지털 정보를 출력할 수 있다. 먼저 $\{c_1, c_2, \cdots, c_m\}$, $(m \le k)$를 공모자로 추정되는 사용자들의 집합이라 하자. DS는 $\{(c_1, f(c_1)), \cdots, (c_m, f(c_m))\}$를 해의 일부로 하고 나머지 근들은 $f(x)$와 일치하지 않는 새로운 다항식 $h(x)$를 선택한다. 즉,

$$\{f(x)\text{의 해}\} \cap \{h(x)\text{의 해}\} = \{(c_1, f(c_1)), \cdots, (c_m, f(c_m))\}.$$

DS는 이 새로운 다항식 $h(x)$와 공모자로 추정된 사용자들의 갱신키, 그리고 새로운 $(z-m)$개의 쌍을 이용하여 갱신블록을 생성하고 체크 비트는 $\xi=1$로 한다. 이러한 갱신블록을 포함한 브로드캐스트 메시지에 대해 불법 디코더가 정확한 디지털 정보를 출력해내면 DS는 $\{c_1, c_2, \cdots, c_m\}$를 공모자 집단으로 확신한다. 이때 [13]의 보조정리1은 불법 디코더가 다항식 $f(x)$를 이용한 갱신블록과 $h(x)$를 이용한 갱신블록에 대해 계산적으로 구별 불가능함을 보장한다.

3.2.3 권한 박탈

$\{c_1, c_2, \cdots, c_m\}$, $(m \le z)$를 권한이 소멸되어야 할 사용자 집합이라 하자. 이들은 불법 사용자들일 수도 있고, 권한기간이 만료되거나 스스로 권한을 포기하고자 하는 사용자들일 수도 있다. DS는 먼저 이 대상자들의 갱신키 $I=\{(c_1, f(c_1)), \cdots, (c_m, f(c_m))\}$를 포함하여 다음과 같이 갱신블록을 생성한다.

1. m개의 갱신키 I와 $(z-m)$개의 사용하지 않는 쌍을 이용하여 다음과 같이 갱신블록을 구성한다.

$$\langle \theta_1 g^{\mu a_0}, \theta_2 g^{\mu a_0}, (c_1, g^{\mu f(c_1)}), \cdots, (c_m, g^{\mu f(c_m)}), (j_1, g^{\mu f(j_1)}), \cdots, (j_{z-m}, g^{\mu f(j_{z-m})}) \rangle$$

   이때 $\{(j_1, g^{\mu f(j_1)}), \cdots, (j_{z-m}, g^{\mu f(j_{z-m})})\}$는 사용하지 않은 쌍이고 $\mu \in {}_R Z_o$이다.

2. 체크비트는 $\xi=1$로 한다.

DS는 이 갱신블록을 포함하여 브로드캐스트 메시지를 전송한다. 이때 사용자 $\{c_1, c_2, \cdots, c_m\}$는 갱신블록을 복호화 할 수 없고 따라서 자신들의 개인키를 수정할 수 없다. 결국 이들의 개인키는 더 이상 사용가치가 없게 된다.

여기서 제안하는 권한 박탈 기법은 [13]의 방식과 동일하다. 그러나 [13]에서는 최대 z명까지만 완전한 권한 박탈이 가능하고 만일 z명을 초과하게 되면 불법 디코더는 디지털 정보 중 최대 비율 c정도까지는 복호화가 가능하게 된다. 만일 비율 $(1-c)$가 무시할 수 있을 정도의 양이라면, z명을 초과하면서부터 [13]의 권한 박탈 기법은 더 이상 실용적이라고 할 수 없다. 반면 여기서 제안하고 있는 기법은 권한이 박탈된 사용자들의 정보를 계속해서 유지해야 하는 기존의 기법과 달리 권한 박탈이 이루어진 사용자에 대해서는 더 이상 사용자 정보를 관리해야 할 필요가 없다. 왜냐하면 한 번이라도 갱신블록의 내용을 복호화하지 못해서 자신의 개인키를 갱신하지 못한 사용자는 그 이후 갱신블록의 내용을 복호화 할 수 있다 할지라도 개인키가 정당하지 못하므로 권한블록의 세션키를 복호화 할 수 없기 때문이다. 따라서 제안하는 기법에서는 한번 권한 박탈 알고리즘을 수

행할 때 권한 박탈이 가능한 인원수가 최대 z명까지이고 이 인원수를 초과하는 경우 권한 박탈 알고리즘을 인원수에 맞게 여러 번 반복해서 수행하면 모든 인원의 권한 박탈이 가능하게 된다. 결과적으로 해당 사용자들의 개인키는 더 이상 쓸모 없게 되고 불법 디코더는 디지털 정보를 전혀 복호화 할 수 없게 된다.

Ⅳ. 제안 기법에 적용 가능한 자기 방지 기능

이 장에서는 앞에서 제안한 공모자 추적기법에 적용될 수 있는 간단하고 효율적인 자기 방지 기법을 구현한다. 여기서 구현하는 방법은 기존에 M. Naor가 제안한 기법과 달리, 개인키에 사용자의 중요한 정보를 직접 포함시키는 것이 아니라 개인키 중 e_i를 이용하여 자기 방지 기법을 구현한다. 신규 가입자 등록 단계에서 DS는 사용자 B_i의 신원을 확인하고 그 사용자의 신용카드 번호 혹은 계좌번호 등의 중요 정보를 받는다. 그 다음에 DS는 e_i를 포함한 개인키를 B_i에게 전달한다. 이때 e_i에 대한 e_i^{-1}값은 군 G의 위수를 알고 있는 DS만이 계산 가능하다. 따라서 B_i에게 개인키를 전송한 후 DS는 다음 과정을 수행한다.

1. $e_i \cdot e_i^{-1} \equiv 1 \pmod{p'q'}$를 만족하는 e_i^{-1}를 계산한다.
2. 사용자의 중요한 자료 m을 e_i^{-1}를 이용하여 암호화하고 이 암호화된 데이터 $E_{e_i^{-1}}(m)$을 공개 디렉토리에 공개한다.

이후 DS는 결제내역 등 각 사용자와 관련하여 새롭게 변경되는 자료들을 암호화하여 공개 디렉토리에 공개한다. 각 사용자들은 사용료나 결제자료 등에 관한 정보를 자신의 개인키 e_i를 이용하여 $D_{e_i}(E_{e_i^{-1}}(m)) = m$으로 복호화하여 확인해 볼 수 있다. 또한 사용자의 카드 번호나 계좌 번호 등이 변경되는 경우 사용자는 자신의 키 e_i를 이용하여 변경 내용을 암호화하여 DS에게 전달할 수 있다. 따라서 만일 누군가 특정 사용자의 개인키를 얻게 되면 그의 중요 정보를 얻을 수 있을 뿐만 아니라 동시에 이 정보를 변경하여 DS에게 전달하는 것도 가능해 진다. 그러므로 사용자 스스로가 자신의 개인키를 비밀로 유지하는데 좀더 신경을 쓰게 되고 이로부터 자기 방지 기법을 이룰 수 있다. 이 기법은 간단할 뿐만 아니라 사용자의 중요 정보가 변경되는 경우에도 유연하게 대처할수 있어서 보다 효율적이다.

Ⅴ. 적응-선택 암호문 공격에 안전한 제안 기법의 변형

Ⅲ장에서 제안된 기법은 수동적인 공격자에 대해 의미론적인 안전성을 제공하고 있다. 여기서는 Ⅲ장의 기법을 변형하여 적응-선택 암호문 공격에 대한 안전성을 제공할 수 있도록 할 것이다. 이를 위하여 T. Okamoto와 D. Pointcheval의 REACT 하이브리드 전환 방식을 사용한다. 이 방식은 [9]의 정리 11에 의해 랜덤 오라클 모델하에 적응-선택 암호문 공격에 대해 안전성을 제공한다. Okamoto와 Pointcheval은 주어진 메시지 m과 암호문 c에대해 공격자가 $E(m) = c$이 정당한지 아닌지를 판별해 내는 새로운 공격법을 정의하고 이를 평문-체크 공격이라 명칭하였다.

[정의1 (평문-체크 공격)]

공격자는 평문-체크 오라클(PCO)에 어떠한 제한없이 항상 접근 가능하다. 이때 주어진 평문 m과 암호문 c에 대해 PCO가 1을 출력한다는 것은 $E(m) = c$가 성립한다는 것과 동치이다.

하이브리드 전환 방식[9]을 설명하기에 앞서 몇가지 용어정의를 하도록 한다. OW-PCA에 대해 안전한 암호화 기법이라 함은 평문-체크 공격에 있어서 일방향 안전성(one-wayness security)을 제공하는 암호기법을 의미한다. 이때 $(K^{asym}, E^{asym}, D^{asym})$를 OW-PCA에 안전한 비대칭 암호화 기법이라 하고 (E^{sym}, D^{sym})를 수동 공격자에 대해 의미론적 안전성을 제공하는 대칭 암호화 기법이라 하자. 또한 G와 H는 해쉬함수이다. 이때 하이브리드 기법 $(K^{hyb}, E^{hyb}, D^{hyb})$은 다음과 같이 구현된다.

[하이브리드 전환 기법]

- $K^{hyb}(1^k)$: $K^{asym}(1^k)$를 수행하며 키 쌍 (sk, pk)을 출력한다.
- $E^{hyb}(m, R, r)$: 랜덤변수 R와 r을 이용하여 메시지 m에 대한 암호화 메시지 (c_1, c_2, c_3)를 다음과 같이 출력한다.

$c_1 = E_{pk}^{asym}(R)$, $c_2 = E_s^{sym}(m)$ 이때 세션키 $s = G(R)$, $c_3 = H(R, m, c_1, c_2)$.

- $D^{hyb}(c_1, c_2, c_3)$: $R = D_{sk}^{asym}(c_1)$로부터 세션키 $s = G(R)$를 계산한다. $c_3 = H(R, m, c_1, c_2)$가 성립하면 메시지 m을 출력한다.

만일 비대칭 암호기법 $(K^{asym}, E^{asym}, D^{asym})$이 OW-PCA에 대해 안전성을 제공하고 대칭 암호화 기법 (E^{sym}, D^{sym})이 수동 공격자에 대해 의미론적 안전성을 제공하면 이 하이브리드 전환 기법 $(K^{hyb}, E^{hyb}, D^{hyb})$은 적응-선택 암호문 공격에 대해 의미론적 안전성을 제공한다[9]. 따라서 이 기법을 이용하여 변형기법을 제안하고자 한다. 변형을 위해서 필요한 것은 단지 이 논문에서 제안한 기법 중 2.1절에서 설명한 브로드캐스트 암호화 기법이 OW-PCA에 대해 안전하다는 것만을 보이면 된다. 이를 위한 증명은 Gap-Diffie-Hellman 문제(GDH)에 기반하고 있다[8,9].

[문제 1. (GDH 문제)]

Decision Diffie-Hellman 오라클을 이용하여 Computational Diffie-Hellman 문제를 해결하는 문제.

[정리 4]

이 논문에서 제안한 암호화 기법은 GDH 문제에 기반해서 OW-PCA에 대한 안전성을 제공한다.

(증명)

주어진 공개키 (h, t)와 암호블록 $\langle sh^{dx_M}, A_1 t^{rx_M}, A_2 t^{rx_M}, t^{-rx_M d}, t^r, d\rangle$ 그리고 평문 메시지 s'에 대해 PCO는 $s = s'$인지 아닌지 구별한다. 이는 $(t^r, h, (t^r)^{x_M d}, sh^{dx_M}/s')$가 D에 포함되는지 아닌지 구별하는 것과 같다. 따라서 PCO가 1을 출력한다면 $(t^r, h, (t^r)^{x_M d}, sh^{dx_M}/s') \in D$가 성립한다는 것을 의미하는 것임을 알 수 있다. 따라서 이때 PCO는 정확하게 Decision Diffie-Hellman Oracle이 된다. □

이제 두 개의 해쉬함수 G와 H를 선택하고 수동 공격자에 대해 의미론적 안전성을 제공하는 대칭 암호화 기법 (E^{sym}, D^{sym})를 선택한다. 그러면 디지털 정보 m에 대해 다음과 같이 변형된 브로드캐스트 메시지 (c_1, c_2, c_3)를 구현하였을 때 이는 랜덤 오라클 모델하에 적응-선택 암호문 공격에 대한 안전성을 제공하게 된다.

$$c_1 = \langle sh^{dx_M}, A_1 t^{rx_M}, A_2 t^{rx_M}, t^{-rx_M d}, t^r, d\rangle$$
$$c_2 = E_K^{sym}(m) \quad where\ K = G(s)$$
$$c_3 = H(s \| m \| c_1 \| c_2)$$

Ⅵ. 결 론

본 논문에서는 새로운 형태의 공모자 추적 기법을 제안하였다. 공모자 추적 및 권한 박탈 과정을 브로드캐스트 암호화 과정과 분리시킴으로써 좀더 효율적인 기법을 구성하였다. 특히, 권한 박탈 과정에서 인원수에 대한 제한을 제거하였고 자기 방지 기법에 있어서도 유동적인 사용자 정보에 효율적으로 대응할 수 있도록 하였다. 또한 (h, t)를 굳이 공개할 필요가 없으며 암호블록 내에서도 t^r값을 생략할 수 있다. 이는 안전성에는 영향을 미치지 않으면서 전송 데이터의 양을 줄이는 효과를 가져온다. 더 나아가서 제안한 기법을 REACT을 이용하여 변형함으로써 적응-선택 암호문 공격에 대해 의미론적 안전성을 제공할 수 있도록 하였다.

참 고 문 헌

[1] J. Anzai, N. Matsuzaki and T. Matsumoto, "A Quick Group Key Distribution Scheme with Entity Revocation", *Proc. Advances in Cryptology - Asiacrypt '99*, Vol. 1716 of Lecture Notes in Computer Science, pp. 333~347, Springer Verlag, 1999.

[2] D. Boneh and M. Franklin, "An Efficient Public Key Traitor Tracing Scheme", *Proc. Advances in Cryptology - Crypto'99*, Vol. 1666 of Lecture Notes in Computer Science, pp. 338~353. Springer Verlag, 1999.

[3] B. Chor, A. Fiat and M. Naor, "Tracing Traitors", *Proc. Advances in Cryptology - Crypto '94*, Vol. 839 of Lecture Notes in Computer Science, pp. 257~270, Springer Verlag, 1994.

[4] C. Dwork, J. Lotspiech and M. Naor,

"Digital Signets: Self-Enforcing Protection of Digital Information", *28th Symposium on the Theory of Computation '96*, pp. 489~498, 1996.

[5] K. Kurosawa and Y. Desmedt, "Optimum Traitor Tracing and Asymmetric Schemes", *Proc. Advances in Cryptology - Eurocrypt '98*, Vol. 1403 of Lecture Notes in Computer Science, pp. 145~157, Springer Verlag, 1998.

[6] A. J. Menezes, P. C. van Oorschot and S. A. Vanstone, "Handbook of Applied Cryptography", *CRC Press*, pp. 351~352, 1996.

[7] M. Naor and B. Pinkas, "Efficient Trace and Revoke Schemes", *Proc. Finalcial Cryptography '00*, Anguilla, February 2000.

[8] T. Okamoto and D. Pointcheval, "The Gap-Problems: a New Class of Problems for the Security of Cryptographic Schemes", *International Workshop on Practice and Theory in Public-Key Cryptography - PKC '01*, Vol. 1992 of Lecture Notes in Computer Science, pp. 104-118, Springer Verlag, 2001.

[9] T. Okamoto and D. Pointcheval, "REACT: Rapid Enhanced-security Asymmetric Cryptosystem Transform", *The Cryptographers' Track of the RSA Conference '2001*, Vol. 2020 of Lecture Notes in Computer Science, pp. 159~175, Springer Verlag, 2001.

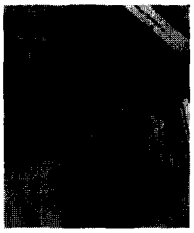
[10] B. Pfitzmann, "Trials of Traced Traitors", *Proc. Workshop in Information Hiding*, Vol. 1174 of Lecture Notes in Computer Science, pp. 49~64 Springer Verlag, 1996.

[11] A. Shamir, "How to Share a Secret", *Comm. ACM*, Vol. 22, No. 11, pp. 612~613, 1979.

[12] D. R. Stinson, "Cryptography Theory and Practice", *CRC Press*, pp. 330~331, 1995.

[13] W. Tzeng and Z. J. Tzeng, "A Public-Key Traitor Tracing Scheme with Revocation using Dynamic Shares", *International Workshop on Practice and Theory in Public-Key Cryptography - PKC '01*, Vol. 1992 of Lecture Notes in Computer Science, pp.207-224, Springer Verlag, 2001.

〈著者紹介〉



김 현 정 (Hyun-Jeong Kim) 학생회원
1994년 2월 : 경희대학교 수학과 졸업
1994년 1월~1999년 12월 : 삼성SDS 근무
1999년 9월~2001년 8월 : 고려대학교 수학과 석사
2001년 9월~현재 : 고려대학교 정보보호 대학원 박사과정
〈관심분야〉 암호이론, 암호 프로토콜, 정보은닉

임 종 인 (Jong-In Lim) 정회원
1980년 : 고려대학교 수학과 졸업
1982년 : 고려대학교 수학과 석사
1986년 : 고려대학교 수학과 박사
1986년~현재 : 고려대학교 수학과 교수
2000년~현재 : 고려대학교 정보보호 대학원 원장
〈관심분야〉 암호이론, 암호 프로토콜, 정보이론

이 동 훈 (Dong Hoon Lee) 정회원
1984년 : 고려대학교 경제학과 졸업
1987년 : Oklahoma Univ. 전산학과 석사
1992년 : Oklahoma Univ. 전산학과 박사
1993년~현재 : 고려대학교 전산학과 교수
2000년~현재 : 고려대학교 정보보호 대학원 교수
〈관심분야〉 암호이론, 암호 프로토콜, 정보이론