다중채널을 위한 실용적인 공개키 Broadcast Encryption Scheme*

정지현, 김종희, 황용호, 이필중

포항공과대학교, 전자전기공학과

A Practical Public Key Broadcast Encryption Scheme for Multiple Channels*

Ji Hyun JEONG, Chong Hee KIM, Yong Ho HWANG, and Pil Joong LEE

Department of Electronic and Electrical Engineering, POSTECH

요 약

본 논문에서는 새로운 공개키 다중채널 broadcast encryption scheme(이하 PK-MCBE라 부른다)을 제안한다. 일반적인 broadcast encryption은 하나의 채널스트림을 전송하는 반면 PK-MCBE는 다수채널의 컨텐츠 스트림을 전송한다. 본 논문에서 제안하는 방식에서 수신자는 단지 하나의 비밀키만을 필요로하며 한번 받은 비밀키는 변경되지 않는다. 제안하는 방식에서는 각 채널당 송신자가 전송하는 메세지의 공통부분을 한번만 전송하여 전체 전송 메세지의 길이를 줄일 수 있다. 또한 배신자(traitors)를 추적하여 효과적으로 강제 탈퇴 시킬수 있다.

1. 서 론

Broadcast encryption scheme[4][1]은 하나의 전송자와 다수의 (합법적인) 수신자로 구성된다. 전송자는 암호화된 데이터를 여러 수신자들에게 보내고, 암호화된 자료가 여러 사용자에게 전송되면 미리 등록된 합법적인 사용자만이 이 자료를 해독할 수 있고, 등록되지 않은 사용자는 이 내용을 받아도 해독할 수 없다. Broadcast encryption scheme은 유료 TV, 저작권이 설정된 CD나 DVD의 배포, 동영상 스트리밍 서비스등의 용도에 사용될 수 있다.

또한 Broadcast encryption scheme은 배신자 추적 기법(traitor tracing method)을 적용하여 악의적인 사용자를 추적하여 강제로 탈퇴시킬 수 있다. 배신자를 추적하는 기법 [3]은 Chor 의해 처음 논의되었고, [4]에서 최대 t명의 악의적 사용자들이 공모하였을때 적어도 한명의 배신자의 키를 알아낼 수 있는 Threshold Traitor 모델로 개선되었다. Boneh와 Franklin는 [2]에서 number-theoretic deterministic 공개키 기반 배신자 추적 스킴을 제안하였다. 최근에 W. Tzeng과 Z. Tzeng은 dynamic share와 revocation 기술을 이용하여 효율적인 공개키 기반의 배신자 추적 및 강제 탈퇴[7] 방법을 제안했는데 Boneh와 Franklin[2]의 방법만큼 효율적이다.

다중 채널 broadcast encryption scheme 응용이 유료 TV이다. 최근 배신자 확인 능력을 가진 효율적인 유료 TV 스킴이 Narayanan[6]에 의해 제안되었다. 이 스킴의 안정성은 discrete logarithm problem에 기본을 두고 있다. 이 스킴에서 사용자는 일정수의 비밀키를 안전한 메모리

* 본 연구는 대학 IT연구센터 육성·지원사업과 교육부 두뇌한국 21사업, Com2MaC-KOSEF의 연구결과로 수행되었음

에 가지고 있으며 구독/취소에 의해 변하지 않는다. Narayanan의 방법은 배신자를 추적할 수 있지만, 많은 수의 비밀키를 저장해야하며, 채널당 메세지의 통신량이 많고, 배신자를 강제 탈퇴 시키기 위해서는 채널을 다시초기화 시켜 새로운 키를 분배해야하는 단점이 존재한다.

본 논문은 이러한 단점들을 해결한 효율적인 다중채널 broadcast encryption scheme을 제안한다.

본 논문의 2장에서는 기본이 되는 이론을 먼저 살펴보고, 3장에서는 Narayanan의 방법을 언급한다. 4장에서는 제안하는 스킴을 상세히 설명하고 Narayanan의 스킴과 비교한다. 5장에서는 결론을 맺는다.

2. Preliminaries

Polynomial Interpolation: $f(x) = \sum_{i=0}^{z} a_i x^i$ 가 차수가 z인 다항식이라고 가정한다. 이때 z개의 점 $(x_0, f(x_0))$, ..., $(x_z, f(x_z))$을 지나는 유일한 다항식 $f(x)$의 계수 $a_0, a_1, ..., a_z$는 다음과 같은 선형방정식에서 구할 수 있다.

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^z \\ 1 & x_1 & x_1^2 & \cdots & x_1^z \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_z & x_z^2 & \cdots & x_z^z \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_z \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_z) \end{pmatrix}$$

각각의 가입자 j가 $(x_j, f(x_j))$를 가지고 있다고 가정하자. 이 경우, z+1명으로 구성된 가입자들의 그룹은 Lagrange 보간법을 이용하여 $f(x)$를 구할 수 있다.

Theorem 1. ("Lagrange 보간법 정리") $f(x)$ 의 차수가 $n-1$이하일 때 $y_n = f(x_n)$를 만족하는 경우 (x_1, y_1), , (x_n, y_n)의 n개의 점을 지나는 함수는 $f(x)$가 유일하며 다음 식으로 구할 수 있다.

$$f(x) = \sum_{t=1}^{n} y_t \frac{(x-x_1)\cdots(x-x_{t-1})(x-x_{t+1})\cdots(x-x_n)}{(x_t-x_1)\cdots(x_t-x_{t-1})(x_t-x_{t+1})\cdots(x_t-x_n)}$$

$$= \sum_{t=1}^{n} y_t \frac{g_t(x)}{g_t(x_t)},$$

$$g_t(x) = (x-x_1)\cdots(x-x_{t-1})(x-x_{t+1})\cdots(x-x_n)$$

$$= \frac{(x-x_1)(x-x_2)\cdots(x-x_n)}{(x-x_t)}.$$

XA = F으로 위식을 나타내면 det(X)≠0이면 $f(x)$의 모든 계수를 $A=X^{-1}F$를 이용하여 계산할 수 있다. 이 경우 상수항 a_0는 X^{-1}의 첫 열벡터에 F를 곱한 것이 되며 다음과 같다.

$$f(x) = \sum_{t=0}^{z} (f(x_t) \cdot \prod_{0 \le j \ne t \le z} \frac{x - x_j}{x_t - x_j}).$$

이때 a_0은 다음과 같이 계산할 수 있다.

$$a_0 = \sum_{t=0}^{z} (f(x_t) \cdot \lambda_t) ,$$

$\lambda_t = \prod_{0 \le j \ne t \le z} \frac{x_j}{x_j - x_t}, 0 \le t \le z$ 는 Lagrange 계수이다. 지수의 경우 $(x_0, g^{f(x_0)})$, $(x_1, g^{f(x_1)})$, ..., $(x_z, g^{f(x_z)})$을 알게 되면 임의의 r에 대하여 $g^{ra_0} = \prod_{t=0}^{z} (g^{f(x_t)})^{\lambda_t}$ 를 계산할 수 있다. 한편, det(X) = 0이면 a_0 이나 g^{a_0}에 대해 아무것도 알 수 없다. 이때 사용자 U_j 는 $(x_j, f(x_j))$ 를 알고 있기 때문에 $(x_0, f(x_0))$, $(x_{z-1}, f(x_{z-1}))$을 이용하여 a_0를 계산할 수 있다. m명의 악의적인 사용들이 자신의 비밀키 $j_1, j_2, , j_m$의 선형 조합으로 새로운 비밀키를 생성하여 불법 사용자, P에게 주더라도 m≤z인 경우 P와 악의적인 사용자들의 그룹은 함께 a_0를 계산할 수 없으며 g^{a_0}도 계산할 수 없다. 제안하는 방법은 이런 원리를 이용한 Tzeng의 배신자 추적 알고리즘 [6]을 이용한다.

3. Narayanan의 방법[5]

최근 Narayanan은 RSA에 기반을 둔 악의적인 사용자의 추적능력을 가진 실용적인 유료 TV 스킴을 제안하였다. 이때 악의적인 사용자를 추적하는 방식은 다음의 원리를 이용하였다:

임의의 s(< t) 개의 벡터들의 선형 조합으로 주어진 n개의 (t+1)차 벡터 $x_1, x_2, ..., x_n$을 구성하면, 사용된 정확한 벡터들을 높은 확률로 알아낼 수 있다.

3.1 Narayanan 스킴의 프로토콜

먼저 m개의 채널을 broadcast하는 콘텐츠 제

공자와 n명의 사용자가 있다고 가정한다. 이 스킴은 *Setup, AddUser, Broadcast, Receive, Subscribe Unsubscribe* 7개의 알고리즘으로 구성된다, 사용자의 채널 수신여부는 $m \times n$ 행렬인 *Subsc* 로 나타내며 사용자 U_j 가 S_i에등록되어 있으면 *Subsc[i, j]* 는 1의 값을 가지고 등록되어 있지 않으면 0의 값을 가진다.

Algorithm Setup

컨텐츠 제공자는 다음과 같은 변수값을 만들어 낸다. $N = pq$, R, $d_r \leq R$ {1, 2. ... $\varphi(N)$} 이때 $1 \leq r \leq 4 + t$ 이고 p와 q는 큰 값의 소수이며 R은 랜덤한 값이다. p, q, d는 콘텐츠 제공자의 비밀키이고, 컨텐츠 제공자는 공개키 N을 공개한다.

Algorithm AddSream

시스템에 새로운 채널스트림 S_i를 추가하기 위해 컨텐츠 제공자는 큰 위수를 갖는 임의의 $g_i \in Z_N^*$를 선택한다. *subsc[i, j]* 는 모든 j에 대해 0의 값을 가지도록 설정하고 g_i값은 공개하지 않는다

Algorithm AddUser

새로운 사용자 U_j를 가입하려면 컨텐츠 제공자는 $\sum_{r=1}^{t+4} e_{rj} d_r = R\Phi(N)+1$를 만족하는 $(e_{1j}, e_{2j}, \ldots, e_{(t+4)j})$을 선택한다. 이때 U_j는 비밀키를 안전한 메모리에 저장한 복호기(Set-Top Terminal)를 받게된다. U_j의 비밀키는 $(e_{1j}, e_{2j}, \ldots, e_{(t+4)j})$이다.

Algorithm Subscribe

사용자 U_j 가 서비스 S_i를 구독하면 컨텐츠 제공자는 사용자 U_j에게 g_i^{e1j}를 전송하고 *Subsc[i, j]*값을 1로 한다.

Algorithm Unsubscribe

사용자 U_j가 서비스 S_i의 구독을 중지하면 컨텐츠 제공자는 *Subsc[i, j]* = 0으로 설정하고, *AddStream*알고리즘에서 했던 것과 같이 새로운 g_i 값을 선택하여 *Subsc[i, j]* = *1*인 모든 사용자들에게 g_i^{e1j}를 전송한다.

Algorithm Broadcast

메시지 M을 채널스트림 S_i에 전송하려면 컨텐츠 제공자는 $\Phi(N)$과 서로 소인 랜덤값 x를 선택하여 암호화된 데이터 $C = (x, C_1, C_2, \ldots, C_{t+4})$를 전송한다. 이때 $C_1 = M^{d1} g_i^x$, $C_2 = M^{d2}$, $C_{t+4} = M^{dt+4}$이다.

Algorithm Receive

채널스트림 S_i로 전송되는 암호화된 데이터 $C = (x, C_1, C_2, \ldots, C_{t+4})$를 복호화 하기 위하여 사용자 U_j는 비밀키 $(e_{1j}, e_{2j}, \ldots, e_{(t+4)j})$를 이용하여 $(\prod_{r=1}^{t+4} C_r^{e_{rj}}) / g_i^{xe_{1j}}$ 을 계산한다. 사용자 U_j는 다음과 같은 과정을 거쳐 콘텐츠 데이터 M를 복원할 수 있다.

$$(\prod_{r=1}^{t+4} C_r^{e_{rj}}) / g_i^{xe_{1j}} = M^{R\Phi(N)+1} = M$$

Cracker identification

악의적인 사용자를 추적하기 위하여 Narayanan의 스킴은 $\sum_{i=0}^{t} \mu_i x_i \equiv 0 (\mathrm{mod}\ P)$ 을 만족하는 $\Upsilon = (\mu_0, \mu_1, \ldots, \mu_t, P)$를 생성한다.

악의적인 사용자(배신자)를 추적하는 알고리즘은 다음과 같다.

입력 : $\mathbf{x}_j$의 s < t개의 선형조합인 벡터 **y**

- 용의자 집합 S를 $\{x_1, x_2, \ldots, x_n\}$를 설정
- **y**에 의해 만족되는 각각의 Υ에 대하여 S에 속하는 각각의 x_j에대해 x_j가 를 만족하지 않으며 S로부터 제거한다.
- S를 출력한다.

3.2 Narayanan 스킴의 문제점

Narayanan의 스킴은 몇 가지 문제점을 가지고 있다. 첫째, *AddUser* 알고리즘에서 모든 사용자는 (t+4)개의 비밀키를 자신의 단말기 기억장치에 저장하고 있어야 배신자를 추적할 수 있다. 두 번째로 하나의 채널당 $(x, C_1, C_2, \ldots, C_{t+4})$ 의 통신량이 필요하다. 통신량은 채널의 개수와 연관되어 있기 때문에 채널이 늘어나면 통신량도 증가하는 문제가 있다. 세 번째로, 컨텐츠 제공자는 배신자 U_j를 찾는다 하더라도 U_j를 제외한 모든 가입자에게 다시 새로운 비밀키를 배포해야만 U_j를 실제로 탈퇴시킬 수 있다.

4. 제안된 방법

본 논문에서는 기존의 방법보다 효율적인 다중 채널 broadcast encryption 스킴을 제안한다. 이 방의 안전성은 discrete logarithm problem(DLP)에 기반을 두고 있다. 이 논문은 Narayanan의 방법에 비해 각각의 수신자의 키 저장공간 및 전송량 측

면에서 효율적이다. 본 논문에서 제안한 스킴에서 사용자는 하나의 비밀키만 필요하며 비밀키를 받은 후 사용자의 비밀키는 변하지 않는다. 각 채널마다 공통적인 *enabling block*이 사용되므로 *enabling block* 전송시 공통된 부분을 한번만 전송해주면 되므로 통신량을 크게 줄일 수 있다. 또한 본 방식에서는 배신자를 강제로 탈퇴시키기 위해서 새로운 비밀키를 다시 분배할 필요가 없다. 추가적으로 공개키를 가진 어떠한 컨텐트 제공자도 사용자들에게 컨텐츠를 전송할 수 있다.

4.1 제안하는 스킴의 프로토콜

먼저 m개의 채널을 broadcast하는 콘텐츠 제공자와 n명의 사용자가 있다고 가정한다. 콘텐츠 제공자는 시스템 파라메타를 설정하고 공개키를 공개한다. 본 스킴에서 콘텐츠 제공자는 신뢰할수 있다고 가정한다. 본 스킴은 *System Setup, AddStream, AddUser(User Registration), Subscribe, Unsubscribe, Encryption and Broadcast, Receive and Decryption, Traitor Tracing, Revocation*으로 구성된다. t는 최대로 공모할 수 있는 악의적 가입자의 수이고, z는 강제로 탈퇴시킬 수 있는 배신자의 최대수이다. $z \geq 2t-1$로 지정한다.

System Setup

q가 매우 큰 소수이면 G_q는 큰 소수의 위수를 갖는 그룹이다. 콘텐츠 제공자는 차수가 z이고 계수가 Z_q의 원소인 다항식 $f(x)=\sum_{i=0}^{z} a_i x^i = a_0 + a_1 x + \cdots + a_z x^z$를 만든다. $f(x)$가 콘텐츠 제공자의 비밀키가 된다. 콘텐츠 제공자는 공개키 $<g, g^{a0}, g^{f(1)}, \ldots, g^{f(z)}>$를 공개한다.

AddStream

컨텐츠(M)를 전송하는 채널 스트림을 S_i라고 하자. 컨텐츠 제공자는 $g_i \in_R Z_q^*$인 g_i를 임의로 선택한다. 이때 $i=1,2,\ldots,m$ 은 채널의 정보이다. g_i는 채널의 비밀키에 해당하므로 안전하게 유지한다.

AddUser(User Registration)

사용자 $U_j (j > z)$가 등록하면 콘텐츠 제공자는 사용자 U_j에게 비밀키 $(j, f(j))$를 가지는 복호기(decoder)를 제공한다.

Subscribe

사용자 U_j가 서비스 S_i를 구독하면 컨텐츠 제공자는 사용자 U_j에게 $g_i^{f(j)}$를 채널로 전송한다. 이때 전송채널은 보안이 이루어지지 않은 채널이어도 무방하다. 이제 사용자 U_j는 $(j, f(j))$와 $g_i^{f(j)}$를 동시에 가지고 있다. 이때, $f(j)$만은 tamper-resistant한 안전한 메모리에 저장하면 된다.

Unsubscribe

사용자 U_j가 서비스 S_i의 구독을 취소하면, 컨텐츠 제공자는 새로운 g_i를 만들고 $g_i^{f(j)}$를 현재의 과금기간(billing period)이끝나면 U_j를 제외한 S_i를 구독하는 사용자들에게 전송한다.

Encryption and Broadcast

먼저 컨텐츠 제공자는 가입자들에게 제공되지 않은 z개의 인덱스를 무작위로 선택한다.

$(j_1, f(j_1)), (j_2, f(j_2)), (j_3, f(j_3)), \ldots, (j_z, f(j_z))$

또한 랜덤값 $r \in_R Z_q$도 만든다. 다음으로 컨텐츠 제공자는 다음과 같이 *enabling block*을 계산한다.

$$T_i = <s_i g^{ra_0}, (g_i^{j_1} g^r), (j_1, g^{rf(j_1)}), (j_2, g^{rf(j_2)}), \ldots, (j_z, g^{rf(j_z)})>$$

이때 T_i는 S_i의 *enabling block*이며 s_i는 채널 S_i에서 여러 사용자에게 동시에 전송되는 데이터를 암호화하는 세션키이다. 컨텐츠 M을 채널내의 사용자에게 동시 전송하려면 컨텐츠 제공자는 $<T_i, E(s_i, M)>$를 전송한다. 이때 E는 대칭키 암호 시스템이다.

Receive and Decryption

가입자 U_j가 $<T_i, E(s_i, M)>$를 받으면, 먼저 s_i를 계산하고, s_i를 이용하여 $E(s_i, M)$를 복호화하여 원래 메시지 M을 얻을 수 있다. 가입자 U_j는 Lagrange 보간법을 이용하여 다음과 같이 복호화한다.

$$s_i = \frac{s_i g^{ra_0}}{(g_i^{j_1} g^r)^{f(x_z)\lambda_z} \cdot \prod_{t=0}^{z-1} (g^{rf(x_t)})^{\lambda_t}} \cdot (g_i^{f(j)})^{j_1 \lambda_z}$$

이때, λ_z는 Lagrange 계수이며 $x_0 = j_1$, $x_1 = j_2$, ..., $x_{z-1} = j_z$, $x_z = j$이다.

Traitor tracing

[13]의 블랙박스 배신자 추적 알고리즘을 응용하여 다음과 같이 배신자를 추적할 수 있다.

$k \leq t$일때, k 명의 가입자 $\{c_1, c_2, \ldots, c_k\}$가 자신들의 비밀키를 조합하여 새로운 형태의 비밀키를

만들었다고 가정하자. *Enabling block*의 비밀키 인덱스 $\{j_1, j_2, ..., j_z\}$가 $\{c_1, c_2, ..., c_k\} \subseteq \{j_1, j_2, ..., j_z\}$를 만족하면 G_q에서 정의된 DLP 문제는 쉽게 풀 수 없다는 가정에 따라 복호기는 암호화된 데이타를 복호하기 위해 불법적으로 제작한 키와 *enabling block*를 사용할 수 없다. 만약 가입자 $\{c_1, c_2, ..., c_k\}$가 배신자일 것으로 의심되면 다음과 같이 인덱스를 *enabling block*에 포함하여 전송한다

$$<s_i g^{r_0}, (g_i^{c_1} g^r), (c_1, g^{rf(c_1)}), ... (c_m, g^{rf(c_m)}), (j_1, g^{rf(j_1)}), ... (j_{z-m}, g^{rf(j_{z-m})})>$$

이때 $j_1, j_2, , j_{z-m}$는 사용되지 않은 인덱스이며 $\{c_1, c_2, ..., c_m\}$와는 다른 값이다. 구체적인 블랙박스 배신자 추적 알고리즘은 다음과 같다.

가입자중에서 k명($k \leq t$)으로 조합가능한 $\{c_1, c_2, ..., c_k\}$에 대하여 다음을 실행한다.

1. 임의의 $z - m$ 개의 사용하지 않은 비밀키 $\{j_1, , j_{z-k}\}$ 를 선택한다.
2. 다음과 같으 테스트 메시지 $<T_i, E(s_i, M)>$를 만든다.

$$T_i = <s_i g^{r_0}, (g_i^{c_1} g^r), (c_1, g^{rf(c_1)}), ... (c_m, g^{rf(c_m)}), (j_1, g^{rf(j_1)}), ... (j_{z-m}, g^{rf(j_{z-m})})>$$

3. $<T_i, E(s_i, M)>$를 복호기에 보낸다.
4. 복호기가 잘못된 M을 출력하면 $\{c_1, c_2, \cdots, c_k\}$이 배신자일 것으로 추측한다.

Step 4에서 발견한 모든 가능한 배신자들 중에서 최소한의 사람들을 배신자로 출력한다.

Revocation

컨텐츠 제공자가 배신자 추적 알고리즘을 통해 배신자 $\{c_1, c_2, \cdots, c_k\}$, $k \leq z$,를 발견했다고 가정하자. 컨텐츠 제공자는 채널을 초기화 시키지 않으면서 효과적으로 배신자를 탈퇴시킬수 있다. 컨텐츠 제공자는 먼저 배신자들의 인덱스로 *enabling block* $(c_1, g^{rf(c1)})$, $(c_2, g^{rf(c2)})$, ..., $(c_k, g^{rf(ck)})$를 만든다. 그리고나서 $z - k$개의 사용되지 않은 인덱스로 나머지 *enabling block* $(j_1, g^{rf(j1)})$, $(j_2, g^{rf(j2)})$, ..., $(j_{z-k}, g^{rf(jz-k)})$를 만든다. 컨텐츠 제공자는 이렇게 만든 다음의 *enabling block*를 전송한다.

$$T_i = < s_i g^{r_0}, (g_i^{c_1} g^r), (c_1, g^{rf(c_1)}), ..., (c_k, g^{rf(c_k)}), (j_1, g^{rf(j_1)}), ..., (j_{z-k}, g^{rf(j_{z-k})}) >$$

배신자들은 그들의 인덱스가 *enabling block*에 사용되었기 때문에 대칭키 s_i를 계산할 수 없다. 참고로 Narayanan의 방법은 배신자를 탈퇴시키기 위해 채널을 다시 초기화 시켜야한다.

4.2 제안하는 스킴의 안전도

본 절에서 제안하는 스킴의 안전도에 대해 논의 한다. 제안하는 스킴의 안전도는 이산대수 문제(DLP)가 풀기 어렵다는 것에 기반 한다. 제안하는 스킴의 안전도를 보이기 위하여 다음의 claim에 대해서 논의한다.

Claim 1. *Z 명 이하의 사용자들이 공모하여 다른 정당한 사용자의 비밀키를 알아 낼 확률은 무시할 수 있다.*

Proof. 제안하는 스킴에서 *System Setup*과 *User Registration* 단계는 W. Tzeng과 Z. Tzeng의 배신자 추적 알고리즘[6]과 같다. 그러므로, claim 1의 증명은 [6]의 theorem 2의 것과 동일하다.

Claim 2. *Z명 이하의 사용자들이 공모하여 자신들이 등록하지 않은 채널의 비밀키를 알아 낼 확률은 무시할 수 있다.*

Proof. 채널 S_j에 등록되지 않은 사용자들이 S_j의 비밀키 g_j를 얻기 위해 공모한다고 가정하자. 공모하는 사용자들이 등록되어 있는 채널의 비밀키들을 $(g_{k_1}, ..., g_{k_v})$ (단, $k_i \neq j$, $1 \leq i \leq v$)라 하고 S_j에 등록된 사용자들을 $(U_{j_1}, ... U_{j_m})$이라 하자. 그러면 공모하는 사용자들은 자신들의 비밀키와 $(g_{k_1}, ..., g_{k_v})$, $(g_j^{f(j_1)}, ..., g_j^{f(j_m)})$ 들로부터 g_j를 얻으려고 시도할 것이다. 그런데, g_j는 $(g_{k_1}, ..., g_{k_v})$와는 독립적으로 G_q에서 랜덤하게 선택된 값이기 때문에 공모하는 사용자들은 $(g_{k_1}, ..., g_{k_v})$로부터 g_j에 관한 정보를 얻을 수 없다.

그리고, Claim 1에 의해 적어도 z명 이상의 사용자들이 자신의 비밀키를 알려주지 않으면 그들은 $f(j_i)$를 알아낼 수 없다. 따라서, 그들에게 $g_j^{f(j_1)}, ..., g_j^{f(j_m)}$는 단순히 G_q에서 랜덤하게 생성되어진 값으로 보여질 것이다. 결국, 공모한 사용자들이 $g_j^{f(j_1)}, ..., g_j^{f(j_m)}$로부터 g_j를 계산할 확률은 $G_q \setminus g_j^{f(j_1)}, ..., g_j^{f(j_m)}$에서 랜덤하게 선택한 g가 g_j와 같을 확률과 같다. 그러므로 g_j에 등록하지 않은 z명 이하의 사용자들이 공모하여 g_j를 얻을 확률은 $1/q \approx 1/(q-m)$이고 이는 무시할 수 있을 정도로 작은 확률이다.

4.3 제안하는 스킴과 Narayanan 스킴과의 비교

본절에서는 제안하는 스킴과 Narayanan 스킴을 비교하여 제안하는 스킴의 효율성을 보인다.

User memory requirement

Narayanan의 방식에서 각 사용자는 $(e_{1j}, e_{2j}, ..., e_{(t+4)j})$에 해당하는 t+4개의 비밀키를 가져야 한다. 그러나 본 논문에서 제안하는 방식에서는 모든 사용자는 단 하나의 비밀키 $f(j)$만 가지면 된다. 그러므로 본 논문은 각 사용자의 비밀키의 수를 상당히 줄였다. 여기서 t는 공모가능한 가입자의 최대수 이다.

Communication overhead

Narayanan의 방식에서 컨텐츠 제공자가 $(x, C_1, C_2, ..., C_{t+4})$를 각 채널마다 전송해야 하므로 채널수가 증가할 수록 통신량도 증가했다. 그러나 본 논문에서 제안한 방식에서는 각 채널의 *enabling block* $T_i=\langle s_i g^{ra0}, (g_1^{j1} g^{r}), (j_1, g^{r(j1)}), ..., (j_z, g^{r(jz)})\rangle$에서 $\langle (j_1, g^{r(j1)}), ..., (j_z, g^{r(jz)})\rangle$ 부분을 공통으로 사용할 수 있다. 따라서 공통적인 부분을 단 한번만 전송하면 통신량을 크게 감소시킬 수 있다.

Traitor revocation

Narayanan의 방식에서는 컨텐츠 제공자가 배신자 찾아내더라도 배신자를 제외한 모든 다른 사용자에게 새로운 비밀키를 배포해야 배신자를 탈퇴시킬 수 있다. 이런 과정을 통해서 배신자를 강제로 탈퇴시키는 것은 번거로운 작업이다. 그러나 본 논문에서 제안한 방식에서는 컨텐츠 제공자는 단지 *enabling block*을 계산할 때 배신자의 비밀키를 이용하기만 하면 된다. 예를 들어서, 사용자 U_k가 $(k, f(k))$를 비밀키로 가지는 배신자라면, 컨텐츠 제공자는 U_k의 비밀키를 *enabling block*에 다음과 같이 포함한다. $T_i=\langle s_i g^{ra0}, (g_1^{j1} g^{r}), (j_1, g^{r(j1)}), ..., (k, g^{r(k)})\rangle$. 이제 U_k는 전송된 암호화된 데이터 $\langle T_i, E(s_i, M)\rangle$를 받더라도, U_k는 s_i를 얻지 못하므로 전달된 데이터를 복호화 할 수 없다.

Multiple contents suppliers

Narayanan의 방식에서는 컨텐츠 제공자가 시스템을 설정하고 관리해야 한다. 만일 많은 컨텐츠 제공자가 존재한다면, 각각의 컨텐츠 제공자는 독립적으로 시스템을 설정하고 관리해야 한다. 그러므로 사용자가 메모리에 저장해야 하는 비밀키의 수는 컨텐츠 제공자의 수가 증가함에 따라 선형적으로 증가하게 된다. 본 논문에서 제공하는 방식에서는 TTP(Trusted Third Party)가 시스템을 설정하고 관리할 수 있다. TTP는 공개키를 공개하고 비밀키 g_i를 안전한 채널을 통해 각각의 컨텐츠 제공자들에게 준다. 각각의 컨텐츠 제공자들은 TTP가 공개한 공개키와 자신에게 전송된 비밀키를 통해 컨텐츠 데이터를 암호화하여 사용자들에게 전송한다. 채널스트림 S_i를 구독하는 사용자 U_j는 TTP로 부터 $g_i^{f(j)}$를 받아 등록시 받은 비밀키 $(j, f(j))$와 $g_i^{f(j)}$로 암호화된 데이타를 복호화한다. 그러므로 각각의 사용자는 컨텐츠 제공자의 수와는 상관없이 TTP로 부터 등록시 받은 하나의 비밀키만을 저장하면 된다.

5. 결 론

본 논문은 discrete logarithm problem (DLP)에 안전도의 기반을 둔 효율적인 다중채널 broadcast encryption scheme을 제안하였다. 제안된 스킴에서 각 사용자는 단지 하나의 비밀키를 위한 저장공간만 필요하다. 메세지의 길이 측면에서도 *enabling block*의 공통부분을 한번만 전송해 주면 되기 때문에 통신량이 대폭 줄어들게 된다.

참고문헌

[1] C. Blundo and A. Cresti, Space requirements for broadcast encryption, EUROCRYPT'94, LNCS 950, pp.287-298, 1994

[2] D. Boneh and M. Franklin, An efficient public key traitor tracing scheme, CRYPTO'99, LNCS 1666, pp. 338-353, 1999

[3] B. Chor, A. Fiat and M. Naor, Tracing traitors, CRYPTO'94, LNCS 839, pp.257-270, 1994

[4] A. Fiat and M. Naor, Broadcast encryption, CRYPTO '93, LNCS V.773 pp.480-491, 1993

[5] M. Naor and B. Pinkas, Efficient Trace and Revoke Schemes, Financial Cryptography'2000, LNCS 1962, pp. 1-20, 2000

[6] A. Narayanan, Practical Pay TV schemes, to appear in the Proceedings of ACISP'03, July, 2003

[7] W.G. Tzeng and Z.J. Tzeng, A Public-Key Traitor Tracing Scheme with Revocation Using Dynamic Shares, PKC'2001, LNCS 1992, pp.207-224, 2001