情報保護學會論文誌
第14券 第6號, 2004. 12

증명 가능한 트리기반 중앙 분배 방식의 그룹키 기법: 안전성 모델 및 변환모듈*

김 현 정†, 이 수 미‡, 이 동 훈
고려대학교 정보보호대학원

Provably Secure Tree-Based Centralized Group Key Distribution: Security Model and Modular Approach

Hyun-Jeong Kim†, Su-Mi Lee‡, Dong Hoon Lee
Graduate School of Information Security, Korea University

요 약

수년간 두 명의 사용자 혹은 세 명의 사용자 사이의 키교환 프로토콜을 위한 안전성 모델이 정의 되어왔다. 또한 최근에는 그룹키 관리 기법에 대한 안전성 모델에 관한 연구가 진행되고 있다. 그 결과 분산 방식의 그룹키 교환 기법을 위한 안전성 모델과 증명 가능한 프로토콜들이 다양하게 제시되고 있다. 그러나 중앙 분배 방식의 그룹키 분배 기법에 대해서는 구체적인 안전성 모델이나 증명 가능한 프로토콜에 대해 거의 언급되지 않았다. 본 논문에서는 중앙 분배 방식의 그룹키 분배 기법을 위한 안전성 요구 조건과 안전성 모델에 대해 설명한다: 이 모델은 강력한 사용자 공모 공격(strong user corruption attack) 능력을 지니고 있는 공격자에 의해 제어되는 채널에서 정의된다. 본 논문에서는 이 안전성 모델에 기반하여 기존의 중앙 분배 방식의 그룹키 기법을 안전성이 증명 가능한 기법으로 전환할 수 있는 변환 모듈을 제시하고자 한다.

ABSTRACT

During the last decade, security models have been defined for two- and three-parity key exchange protocols. Currently there is a growing research interest in security models for group key management schemes. While various security models and provably secure protocols have been proposed for distributed group key exchange schemes, no results are known for centralized group key distribution schemes in spite of their theoretical and practical importance. We describe security requirements and a formal security model for centralized group key distribution scheme: we define the model on the channel controlled by adversaries with the ability of strong user corruption. In the security model, we propose a conversion module which can transform centralized tree-based group key distribution schemes in the literature to provably secure centralized tree-based group key distribution schemes.

Keywords : *group key distribution, security model, conversion module.*

Ⅰ. 서 론

그룹키 관리 기법은 크게 중앙 분배 방식과 분산 방식으로 분류될 수 있다.[1] 중앙 분배 방식에서는 그룹 구성원과 그룹키를 관리하는 한 명의

접수일 : 2004년 8월 18일 ; 채택일 : 2004년 11월 26일
* 이 논문은 2003년도 고려대학교 박사수료후 연수과정 연구비 지원에 의하여 수행되었습니다.
† 주저자 : khj@cist.korea.ac.kr
‡ 교신저자 : smlee@cist.korea.ac.kr

그룹 관리자가 존재한다.[2-9] 분산 방식은 다시 분산 서브 그룹 방식[10-13]과 완전 분산 기법[14-17]으로 구분할 수 있으며, 분산 서브 그룹의 경우는 여러 명의 그룹 관리자가 존재하고 완전 분산 기법의 경우는 그룹 관리자가 존재하지 않는다. 중앙 분배 방식의 경우 중앙 서버가 존재하므로 안전성 측면에서 약점을 지니고 있으나, 이러한 취약점은 방화벽이나 IDS(침입 탐지 시스템)와 같은 시스템 솔루션을 이용하여 보완될 수 있다. 또한, 중앙 분배 그룹키 관리 기법이 컨텐츠 분배 서비스, 화상 회의, 유무선 인터넷 브로드캐스팅 시스템 등에 적합한 기법임은 의심의 여지가 없다.

그럼에도 불구하고 완전 분산 방식의 그룹키 교환 기법에 대한 안전성 모델과 증명 가능한 기법들은 Bresson et al.에 의해 제시되어 왔지만 중앙 분배 방식의 기법에 대해서는 Mayer와 Yung이 제시한 기법[18]을 제외하고는 언급된 내용이 없다. Mayer와 Yung은 처음으로 [18]에서 양자간 키 교환 기법을 중앙 분배 방식의 그룹키 분배 기법으로 전환하는 변환 모듈을 제시하였다. 그러나 이들은 안전성 요구조건이나 모델을 자세히 언급하지 않았으며 이 변환 모듈은 일대일 키교환 방식을 일대다 키교환 방식으로 확장하는 형태로 이루어지기 때문에 통신량이 사용자수에 기반하여 증가하는 문제점을 지니고 있다.

따라서 기존의 중앙 분배 방식의 그룹키 분배 기법은 대부분이 트리 구조에 기반하고 있다. 트리 기반에 구조한 중앙 분배 방식의 기법이 일대다 방식의 그룹키 분배 기법보다는 보다 효율적인 형태이기 때문이다. 사용자의 가입/탈퇴가 수시로 이루어지는 그룹키 분배 기법에서 중요시되는 안전성은 전방보호와 후방보호이다. 즉, 새로 가입한 구성원은 가입 이전의 그룹키 정보를 얻을 수 없어야 하고 탈퇴한 구성원은 탈퇴 이후의 그룹키 정보를 얻을 수 없어야 한다는 것이다. 기존의 트리 구조 기반 중앙 분배 방식의 그룹키 분배 기법들은 대부분 전방보호와 후방보호의 안전성은 만족시키고 있다. 그러나 이렇게 제시된 기법들에 대해 구체적인 안전성 모델에 기반한 안전성 증명은 제시되고 있지 않다. 더구나 기존 기법들 중에는 키관리 기법에서 가장 기본적으로 요구되는 안전성인 완전 전방보호는 만족되지 않는 경우도 있다. 본 논문에서는 중앙 분배 방식의 그룹키 분배 기법을 위한 안전성 모델을 제시하고, 이를 기반으로 변환 모듈을 제시하고자 한다. 이 변환 모듈을 이용하여 기존의 트리 기반의 중앙 분배 방식의 그룹키 분배 기법들을 안전성이 증명 가능한 기법들로 전환할 수 있다.

본 논문의 구성은 다음과 같다. 2절에서는 안전한 중앙 분배 방식의 그룹키 분배 기법을 위한 안전성 용어 및 요구 조건과 안전성 모델에 대해 설명한다. 3절에서 변환 모듈을 제시하고 4절에서 제시된 변환 모듈의 안전성을 증명한다. 5절에서는 제시된 변환 모듈을 이용하여 기존의 기법을 전환하는 예를 보이고 마지막으로 5절 이 논문에 대한 결론을 논한다.

II. 안전성 모델

이 절에서는 중앙 분배 방식의 그룹키 분배 기법을 위한 안전성 용어와 안전성 요구조건을 정리하고 안전성 모델을 설명한다.

1. 정의 및 용어

- **참가자 (Participants).** 그룹 관리자 (GC)는 트리기반 중앙 분배 방식의 그룹키 분배 기법 P를 초기화하고 그룹 구성원들의 가입/탈퇴 및 모든 비밀키를 관리 한다. U는 사용자 집합으로써 여기에 속한 사용자 u_i은 누구나 프로토콜 P에 참여할 수 있다. 이때 공격자는 프로토콜의 참여자는 아니다.
- **고정키와 임시키 (Long-Lived Keys와 Short-Lived Keys).** 기존의 트리 기반의 중앙 분배 방식의 그룹키 분배 기법들을 살펴보면 구성원들의 고정키는 주로 사용자가 그룹에 가입하는 시점에서 사용자 인증을 위해 사용된다. 즉, 사용자 인증을 위한 서명키로 사용된다. 이 경우 고정키는 사용자가 그룹에 가입한 이후에 새로운 그룹키를 전송받기 위해서는 사용되지 않으며 대신 임시키가 사용된다. 이 경우 고정키가 공격자에게 노출되었을 때 전방 공격에 안전할 지라도 임시키가 공격자가 노출되었을 때 전방 공격에 안전하지 않다면

전방 공격에 대한 안전성이 보장된다고 할 수 없다. 따라서 중앙 분배 방식의 그룹키 분배 기법에서는 임시키의 안전성이 고정키의 안전성과 동일하게 중요시 되어야 한다.

- **Grouping과 GID.** 그룹의 구성원이 변경될 때마다 프로토콜 P의 실행 결과로써 새로운 그룹 G_j가 형성된다: G_1은 최초의 그룹 구성원들의 집합을 의미하며 인텍스 j는 새로운 그룹이 생성될 때마다 증가한다. 각 그룹 G_j는 유일한 그룹키 K_j와 대응되며 G_j의 그룹 구성원들은 그룹키 K_j를 이용하여 그룹 데이터를 얻을 수 있다. 이 개념은 기존의 partnering 개념과 유사하다. 새로운 그룹 G_j를 생성하기 위해 프로토콜 P가 정상적으로 수행된 후에 G_j의 각 구성원들은 그룹키 K_j와 그룹 아이디(GID)를 얻을 수 있다. GID=$\|ID$로 정의되며 이때 ID는 새로운 그룹 G_j에 속한 그룹 구성원들의 아이디의 모임을 의미한다.

[**정의** 1] P를 트리 기반 중앙 분배 방식의 그룹키 분배 기법이라 하고 G_j는 구성원 $u_1, \ldots, u_n$의 집합이라 하자. 이때 다음을 만족하면 G_j의 구성원들은 grouped되었다고 한다.

(1) G_j에 대응되는 유일한 그룹키 K_j가 존재한다.
(2) 각 구성원 u_i는 K_j와 GID를 얻는다.

- **공격자 능력.** 중앙 분배 방식의 그룹키 관리 기법에서 GC가 공격자 A와 공모한다면 모든 그룹 구성원들의 고정키/임시키와 그룹키가 A에게 노출될 수 있으므로 중앙 분배 방식의 그룹키 관리 기법에서는 서버 공모 공격에 대한 문제는 시스템 솔루션 측면에서 해결될 수 있어야 한다. 따라서 중앙 분배 방식의 그룹키 관리 기법의 안전성 모델에서 공격자는 서버 공모 공격 능력을 수행할 수 없다. 반면 사용자 공모 공격에 대해서 공격자는 강력한 사용자 공모 공격을 수행할 수 있도록 한다. 그 이유는 중앙 분배 방식의 그룹키 관리 기법에서는 고정키의 안전성만큼 임시키의 안전성도 중요시되어야 하기 때문이다. 강력한 사용자 공모 공격을 수행할 수 있는 경우 공격자는 사용자의 고정키 뿐만 아니라 사용자의 임시키도 획득할 수 있다. 따라서 중앙 분배 방식의 그룹키 분배 방식의 안전성 모델에서 공격자는 서버 공모 공격 능력이 제한된 능동적 공격자로 정의될 수 있다.

2. 안전성 요구조건

키관리 기법에서 요구되는 기본적인 안전성은 키 구별 불가능성 (Key Indistinguishability)과 완전 전방 보호 (Perfect Forward Secrecy)이다. 여기에 그룹키 관리 기법에서 추가로 요구되어지는 안전성 요구 조건에는 새로 가입한 구성원은 가입 이전의 그룹키를 알 수 없어야 하고, 그룹을 탈퇴한 구성원은 탈퇴 이후의 그룹키를 알 수 없어야 한다는 것이다. 이는 각각 후방보호 (Backward Secrecy)와 전방보호 (Forward Secrecy)로 표현되어진다. [19]에서는 전방보호와 후방보호 그리고 전후방 보호라는 개념에 대해서 정의하고 있다. [19]의 전후방 보호의 정의를 조금 수정하면 전후방 보호는 전방보호와 후방보호를 동시에 만족하는 개념으로 새로 가입한 구성원과 탈퇴한 구성원이 공모하여 자신들이 속하지 않은 그룹의 그룹키를 얻을 수 없어야 함을 의미한다. 따라서 안전한 중앙 분배 방식의 그룹키 분배 기법을 위한 안전성 요구 조건은 다음과 같다.

- 키 구별 불가능성: 공격자는 그룹키를 암호화한 메시지를 얻더라도 그룹키에 대한 정보를 얻는 것이 쉽지 않아야 한다.
- 완전 전방 보호: 공격자가 사용자의 고정키나 임시키 또는 그룹키를 얻더라도 그 이전의 그룹키에 대한 정보를 얻는 것이 쉽지 않아야 한다.
- 전후방 보호: 새로 가입한 구성원과 그룹 탈퇴자가 공모하더라도 그들이 속하지 않은 그룹의 그룹키 정보를 얻는 것이 쉽지 않아야 한다.

3. 안전성 모델

중앙 분배 방식의 그룹키 관리 기법 P에 대한

공격자 A의 공격은 공격자가 수행하는 다양한 질의로 정의될 수 있다.

- Send(GC, G_j, m): 이 질의를 이용하여 A는 GC를 대신하여 G_j에 속한 모든 사용자들에게 메시지 m을 전송할 수 있다. 이때 사용자들은 응답 메시지를 GC에 전송하지 않으므로 메시지 전송 후 어떤 응답도 받을 수는 없다.
- Send(u_i, G_j, m): 이 질의를 이용하여 A는 사용자 u_i를 대신하여 GC에게 메시지 m을 전송할 수 있다. 이때 m="join" 또는 "leave"의 경우 GC는 Join 알고리즘이나 Leave 알고리즘을 수행하므로써 새로운 그룹 G_{j+1}을 생성한다. A는 이 질의 수행 후 그 응답 메시지를 받을 수 있다.
- Setup(GC),Join(J),Leave(R): 이 질의를 이용하여 A는 GC로 하여금 프로토콜 P를 초기화하도록 하거나, 집합 J에 속한 사용자들을 새로운 구성원으로 가입하거나 R에 속한 그룹 구성원들을 탈퇴하도록 한다.
- Reveal(G_j): 이 질의를 이용하여 A는 그룹 G_j의 그룹키 K_j를 얻을 수 있다.
- Corrupt(u_i, G): A는 u_i의 고정키와 그룹키 K_j를 얻기 위해 사용한 u_i의 임시키를 얻을 수 있다.
- Test(G_j): A는 이 질의를 fresh한 그룹 G_j에 대해 수행할 수 있으며 그 수행 시점에는 제한이 없으나 수행 횟수는 단 한번으로 고정된다. (fresh에 대해서는 아래서 정의한다.) 이 질의를 수행하면 랜덤 비트 b의 결과에 따라 $b=1$이면 A는 G_j의 그룹키 K_j를 받고 $b=0$이면 랜덤값을 받게 된다. A가 자신이 수행할 수 있는 모든 질의들을 수행한 후에 최종적으로 b'을 출력하게 되고 이때 $b'=b$가 되면 A가 그룹키 K_j에 대한 특정 정보를 얻었음을 의미하고 프로토콜 P는 안전하지 않음을 의미한다.

[정의 2] 그룹 G_j가 Reveal(G_j)질의를 받지 않고 G_j의 어떤 구성원 u_i도 Corrupt(u_i, G_k), ($k \le j$)질의를 받지 않는다면 그룹 G_j는 fresh하다고 정의한다.

Adv_P^{ind}를 공격자가 Test질의의 랜덤 비트 b를 추측하기위해 얻을 수 있는 가치 있는 정보의 량(advantage)이라 하자. 이때 충분히 큰 비트 k에 대해 $Adv_P^{ind} \le e(k)$를 만족하는 negligible function e이 존재하면 프로토콜 P는 안전한 중앙 분배 방식의 그룹키 분배 기법이라 할 수 있다.

Ⅲ. 변환모듈

1. 변환모듈

이 절에서는 기존에 제안된 트리기반 중앙 집중 방식의 그룹키 분배 기법 P를 증명 가능한 안전성을 지닌 기법 P+로 전환하는 모듈을 제안한다. 기존에 제안된 기법들은 대부분 전방보호와 후방보호를 만족하고 있는데 변환 모듈을 적용하면 본 논문에서 제안하는 안전성 모델에 안전한 기법으로 변환이 가능하다. 변환 모듈을 적용하기 위해 기존 프로토콜 P가 만족해야하는 조건은 전방보호만을 만족하는 기법이면 된다.

기호. $G_n = \{u_1, \cdots, u_m\}$을 현재 그룹이라 하자. G_n의 각 구성원은 그룹키 K_n을 소유하고 높이가 $\log m$ 이진트리의 리프(leaf)에 대응된다. 트리의 각 노드 인덱스는 이진수 v로 표기하고 각 노드의 키는 k_v라 한다. 특히 루트노드의 경우 인덱스 0로 표현되며 루트키는 k_0로써 그룹키 K_n을 의미한다. 노드 v에 대하여 $v\|0$와 $v\|1$은 각각 왼쪽 자식 노드와 오른쪽 자식 노드를 의미하고 $p(v)$는 v의 부모노드이고 $s(v)$는 v의 형제노드가 된다. 예를 들어, 구성원 u_i가 리프 $v = b_0 b_1 \cdots b_d$에 대응된다고 하자. 이때 $d = \log m$이고 모든 $j \in \{0,1,\cdots,d\}$에 대해 $b \in \{0,1\}$이다. 이때 구성원 u_i의 개인키는 리프 v에서 루트 0까지 연결되는 경로에 존재하는 모든 노드들의 키가 된다. 즉, u_i가 소유하는 비밀키는 $\{k_v, k_{p(v)}, \cdots, k_{p^{t-1}(v)}\}$와 그룹키 $k_{p^t(v)} = K_n$이 된다. (여기서 모든 $1 \le t \le 0$에 대해 $p^t(v) = b_0 b_1 \cdots b_{d-t}$이다.)

Setup. GC는 프로토콜 P의 Setup 알고리

즘을 먼저 수행한다. 그 후 CCA공격 (Chosen Chiphertext Attacks)에 안전한 의사난수치환 (pseudo-random permutation) $F_S:\{0,1\}^{\ell}\rightarrow\{0,1\}^{s}$을 생성하고 CCA 공격에 안전한 대칭키 암호 알고리즘 E_K를 생성한다. 이때 S와 K는 키 집합 $\{0,1\}^{*}$로부터 선택되며 ℓ과 s는 안전성 매개변수이다. 또 일방향 함수 $H:\{0,1\}^{*}\rightarrow\{0,1\}^{\ell}$을 생성한다. F_S는 그룹 구성원들에게 공개되지 않고 대칭키 암호 알고리즘 E와 일방향 함수 H는 그룹 구성원들에게 공개된다. 이때 대칭키 암호 알고리즘의 비밀키 K는 그룹 구성원들에게 안전하게 전송되어야 하며 이 값은 그룹 구성원들의 고정키가 된다.

Join. J집합에 속한 새로운 사용자들이 현재 그룹 G_n에 가입시키기 위해서 GC는 새로운 그룹 $G_{n+1}=G_n\cup J$를 생성하고 다음 과정을 수행한다.

1. GC는 랜덤한 새로운 그룹키 $K_{n+1}\in\{0,1\}^{\ell}$과 랜덤수 $R\in\{0,1\}^{\ell}$을 선택한다.
2. GC는 두개의 노드키 k_{00}와 k_{01} 그리고 F_S를 이용하여 $k_{00}\oplus F_S(R)\oplus R$과 $k_{01}\oplus F_S(R)\oplus R$을 계산한다.
3. GC는 다음과 같이 그룹 구성원들에게 키-재생성 암호 메시지 (re-keying encryption message) $B^{J}(K_{n+1})$을 생성하고 이를 서명키를 이용하여 서명한 후 서명값과 함께 그룹 구성원들에게 전송한다.

$$B^{J}(K_{n+1})=[E_{K\oplus F_S(R)\oplus R}(K_{n+1}),\ k_{00}\oplus F_s(R)\oplus R,\ k_{01}\oplus F_s(R)\oplus R].$$

4. GC는 두 노드키 k_{00}와 k_{01}를 일방향 함수 H를 이용하여 다음과 같이 재생성한다.

$$k'_{00}=H(k_{00}\|K_{n+1}),\ k'_{01}=H(k_{01}\|K_{n+1}).$$

5. 새로운 가입자들을 위해 새로운 노드가 필요한 경우 GC는 기존의 이진 트리에 새로운 노드를 추가하고 가입자들을 새로운 리프에 대응시킨다. 가입자들의 개인키는 해당 리프부터 루트에 해당하는 노드키들이 된다. 이때 기존에 존재하던 노드 키들 중 새롭게 변경되는 키는 k_{00}와 k_{01}를 제외하고는 없다.
6. GC는 새로운 가입자들에게 각 개인키와 그룹키 K_{n+1}, 대칭키 암호 알고리즘의 비밀키 K를 안전하게 전송한다. 가입자들이 k'_{00} 또는 k'_{01}를 얻을지라도 그 이전의 키 k_{00} 또는 k_{01}를 얻는 것은 쉽지 않다.

G_n에 속한 기존의 사용자들은 메시지 $B^{J}(K_{n+1})$를 전송받은 후 자신들의 노드키 k_{00} 또는 k_{01}를 이용하여 새로운 그룹키 K_{n+1}를 얻고, k'_{00} 또는 k'_{01}를 생성한다.

Leave. 현재의 그룹 G_n으로부터 R에 속한 구성원들을 탈퇴시키기위해서 GC는 새로운 그룹 $G_{n+1}=G_n-R$을 생성한 후 다음을 수행한다.

1. GC는 기존 프로토콜 P의 Leave 알고리즘을 수행하여 키-재생성 암호 메시지 Γ를 생성한다. 기존 프로토콜 P의 Leave 알고리즘에 의해 생성된 Γ에 포함된 새로운 그룹키를 T라 표시하자.
2. GC는 랜덤한 새로운 그룹키 $K_{n+1}\in\{0,1\}^{\ell}$과 랜덤수 $R\in\{0,1\}^{\ell}$을 선택한다.
3. GC는 $T\oplus F_S(R)\oplus R$와 $E_{K\oplus F_S(R)\oplus R}(K_{n+1})$을 계산한 후 새로운 키-재생성 암호 메시지 $B^{R}(K_{n+1})$를 다음과 같이 생성하고 이를 자신의 서명키로 서명하여 G_{n+1}에 속한 구성원들에게 멀티캐스팅 한다.

$$B^{R}(K_{n+1})=[\Gamma,\ T\oplus F_S(R)\oplus R,\ E_{K\oplus F_S(R)\oplus R}(K_{n+1})].$$

$B^{R}(K_{n+1})$을 전송받은 그룹 구성원들은 Γ로부터 T를 얻은 후 이를 이용하여 그룹키 K_{n+1}을

계산한다. 그 후, Γ에서 암호화키로 사용된 노드키 k_i들을 일방향 함수 H를 이용하여 $k'_i = H(k_i \| \| K_{n+1})$로 재생성 한다.

2. 안전성

P를 기존에 제시된 트리기반 중앙 분배 방식의 그룹키 분배 기법이라 하자. 이때 탈퇴한 구성원들의 개인키를 이용하여 P의 Leave 알고리즘에 의해 생성되는 키-재생성 암호 메시지 Γ로부터 새로운 그룹키 T를 얻을 수 있는 확률의 상한값을 ξ이라 하자. P가 전방보호를 보장한다는 의미는 충분히 큰 비트 k에 대해 $\xi \leq \nu(k)$를 만족하는 negligible function ν가 존재함을 뜻한다.

이제 전방보호를 만족하는 프로토콜 P에 대해 제시하는 변환 모듈을 적용하여 생성된 새로운 프로토콜 P+에 대한 안전성을 위해 정리 1을 증명함으로써 제시된 변환 모듈의 안전성을 보이고자 한다. 증명에 앞서 먼저 CCA 공격에 안전한 의사 난수 치환에 대해 살펴보면 다음과 같다.

$\{0,1\}^s$를 의사 난수 치환 패밀리 (pseudo-random permutation family) F의 키집합이라 하자. 즉, $F: \{0,1\}^s \times \{0,1\}^\ell \rightarrow \{0,1\}^\ell$라 하자. $Perm$은 $g: \{0,1\}^\ell \rightarrow \{0,1\}^\ell$과 같은 난수 치환의 패밀리 (random permutation family)라 하자. 그러면 CCA 공격 능력을 갖춘 공격자 A에 대해 안전한 의사 난수 치환 패밀리 F는 다음과 같이 정의할 수 있다.

$$Adv_{F,A}^{pr\text{-}pcca} = P[b=1 : F \rightarrow g; A^{g, g^{-1}} \rightarrow b] - P[b=1 : Perm \rightarrow g; A^{g, g^{-1}} \rightarrow b],$$

$$Adv_F^{pr\text{-}pcca} = max_A \{ Adv_{F,A}^{pr\text{-}pcca} \}.$$

이때, $Adv_F^{pr\text{-}pcca} \leq \nu(k)$를 만족하는 negligible function ν가 존재하면 F는 CCA 공격에 안전하다고 한다.

[정리 1] P를 전방보호를 만족하는 트리기반 중앙 분배 방식의 그룹키 분배 기법이라 하고 F_S, E_K, H를 이용하여 변환 모듈에 의하여 전환된 기법을 P+라 하자. m은 그룹 구성원 수이고 $d = log m$이라 하자. 또한 q_j, q_l, q_r, q_c는 각각 Join, Leave, Reavel, Corrupt 질의의 개수라 하고 t는 공격자의 공격 수행 시간이라 하자. 만일, ξ를 P의 Leave 알고리즘에서 생성되는 키-재생성 암호 메시지로부터 T를 얻을 수 있는 확률의 상한값이라 하면 다음과 같은 부등식이 성립함을 알 수 있다.

$$Adv_{P+}^{ind}(t, q_j, q_l, q_r, q_c) \leq 2(1 + \frac{1}{1-\xi}) Adv_F^{pr\text{-}pcca}(t', q').$$

이때, $q' = q_j + q_l$이고 $t' = t + O(q')$이다.

(증명) P+가 안전하지 않다고 가정하면 P+를 깰수 있는 공격자 A가 존재한다는 것을 의미한다. 가정의 모순을 위하여 이런 공격자 A가 존재하면 이를 이용하여 의사 난수 치환 (pseudo-random permutation)과 난수 치환 (random permutaion)을 구별할 수 있음을 보이고자 한다.

의사 난수 치환과 난수 치환을 구별하고자 하는 알고리즘을 B라고 하자. 즉, 함수 g가 B에게 주어지고 B는 이 함수에 대해 오라클 접근을 통하여 자신이 선택한 입력값들에 대해 함수의 출력값을 얻은 후 이를 이용하여 주어진 함수 g가 의사 난수 치환인지 난수 치환인지를 판별해야 한다. 이때 B는 다음과 같이 공격자 A의 질의를 시뮬레이션 함으로써 A를 이용하여 g에 대해 판별할 수 있다.

- Send(GC, G_j, m), Send(u_i, G_j, m) : B는 A의 Send 질의를 받아서 A가 보내는 메시지의 서명값이 정당하지 않은 경우 GC 또는 사용자들은 그 이후의 작업을 수행하지 않는다.
- Setup(GC) : B는 P+의 Setup 알고리즘을 정상적으로 수행한다. 다만 의사 난수 치환 F_S를 선택하는 대신 자신에게 주어진 함수 g를 이용한다.
- Join(J): B는 g를 이용하여 키-재생성 암호 메시지 $B'(K_{n+1})$을 다음과 같이 생성한다.

$$B'(K_{n+1}) = [E_{K \oplus g(R) \oplus R}(K_{n+1}), k_{00} \oplus g(R) \oplus R, k_{01} \oplus g(R) \oplus R].$$

- Leave(R): B는 Join 질의와 마찬가지로 g를 이용하여 키-재생성 암호 메시지 $B^R(K_{n+1})$을 생성한다.
- Reveal(G_j), Corrupt(u_i, G_j) : B는 Reveal 질의에 대해서 G_j의 그룹키 K_j를 A에게 전송한다. Corrupt 질의에 대해서는 u_i의 고정키 (여기에는 서명키와 대칭 암호 알고리즘의 비밀키 K가 포함된다.)와 임시키 (각 노드키들)를 A에 전송한다. 이때 임시키는 그룹 G_j의 그룹키 K_j를 얻는데 사용된 임시키를 의미한다.
- Test(G_j): G_j가 fresh하지 않은 경우 B는 시뮬레이션을 중단한다. 그렇지 않으면 랜덤 비트 b의 값에 따라 그룹키 K_j 혹은 랜덤값 r을 A에게 전송한다. A의 출력값 b'이 만약 $b'=b$이면 B는 g를 의사 난수 치환으로 판별하고 그렇지 않으면 난수 치환으로 판별한다.

$Adv^{ind}_{P+,A}$는 $Adv^{ind}_{P+,A} = Adv^{ind}_{P+\,A(J)} + Adv^{ind}_{P+\,A(R)}$로 생각할 수 있다. 이때 $Adv^{ind}_{P+,A(J)}$는 A가 새로운 그룹 구성원이 가입하면서 생성되는 키-재생성 암호 메시지 $B^J(K_j)$에 대해서 얻을 수 있는 가치있는 정보량을 의미하며 $Adv^{ind}_{P+,A(R)}$는 A가 구성원 탈퇴에 의해 생성되는 키-재생성 암호 메시지 $B^R(K_j)$에 대해서 얻을 수 있는 정보량을 의미한다. 그러면 위에 설명한 시뮬레이션으로부터 다음과 같은 등식을 얻을 수 있다.

먼저 구성원 가입을 위한 키-재생성 암호 메시지에 대해서,

$$\begin{aligned} Adv^{prpcca}_{F,B} &= P[b=1 : F\rightarrow g; B^{g,g^{-1}}\rightarrow b] \\ &\quad - P[b=1 : Perm\rightarrow g; B^{g,g^{-1}}\rightarrow b] \\ &\geq \frac{1}{2}\{P[A=1|B^J(K_j)\rightarrow A \\ &\quad - P[A=1|B^J(r)\rightarrow A]\} \\ &\geq \frac{1}{2}Adv^{ind}_{P+,A(J)}. \end{aligned}$$

또한, 구성원 탈퇴를 위한 키-재생성 암호 메시지에 대해서는,

$$\begin{aligned} Adv^{prpcca}_{F,B} &= P[b=1 : F\rightarrow g; B^{g,g^{-1}}\rightarrow b] \\ &\quad - P[b=1 : Perm\rightarrow g; B^{g,g^{-1}}\rightarrow b] \\ &\geq \frac{1-\xi}{2}\{P[A=1|B^R(K_j)\rightarrow A \\ &\quad - P[A=1|B^R(r)\rightarrow A]\} \\ &\geq \frac{1-\xi}{2}Adv^{ind}_{P+,A(R)}. \end{aligned}$$

위의 두 부등식으로부터 다음의 결과를 얻을 수 있다.

$$\begin{aligned} Adv^{ind}_{P+,A} &= Adv^{ind}_{P+\,A(J)} + Adv^{ind}_{P+\,A(R)} \\ &\leq 2(1+\frac{1}{1-\xi})Adv^{prpcca}_{F}. \end{aligned}$$

□

Ⅳ. PRGT[4]에 기반하여 변환된 PRGT+

이 장에서는 제안된 변환 모듈을 기존 기법에 적용하는 예를 보이고자 한다. 기존에 Canetti et al.에 의해 제안된 트리 기반 중앙 분배 방식의 그룹키 분배 기법인 PRGT[4]에 본 논문에서 제안한 변환 모듈을 적용하여 안전성이 증명 가능한 PRGT+ 기법으로 변형하고자 한다.

Setup. GC는 안전성 매개변수를 정의하고 CCA2 공격에 안전한 대칭키 암호 알고리즘 E를 생성한다. 또한 의사 난수 생성기 G를 생성한다. G의 출력값의 길이는 입력값 길이에 두 배가 되어야 한다. 즉, $G(x)=y$이고 $|x|=\ell$일 때, $|y|=2\ell$이 된다. 출력값 y에 대해서 $y=\alpha(x)\|\beta(x)$로 표현할 수 있고 이때 $|\alpha(x)|=|\beta(x)|=\ell$이다. 또한, GC는 의사 난수 치환 F_S와 일방향 함수 H를 생성한다.

Join. J를 그룹 G_n에 가입하고자 하는 사용자들의 집합이라 하자. 이때 GC는 본 논문에서 제시된 Join 알고리즘에 따라 새로운 그룹 $G_{n+1}=G_n\cup J$를 생성하고 새로운 그룹키 K_{n+1}을 그룹 사용자들에게 안전하게 멀티캐스팅 한다.

Leave. 현재 그룹을 $G_n=\{u_1,\cdots,u_m\}$이라 하

자. 이때 G_n을 탈퇴하고자 하는 사용자 u_j를 $u_j = u$라고 하면, GC는 새로운 그룹 $G_{n+1} = G_n - \{u\}$를 생성한다. 설명의 편의를 위하여 u가 리프 u에 대응된다고 하자. ($v = b_0 b_1 \cdots b_d$이고 $d = \log m$, $b_j \in \{0,1\}$이다.) 그러면 GC는 리프 v를 삭제하고 리프 v에서 루트노드까지의 경로에 존재하는 모든 노드키들을 변경한다. 이 노드키들의 변경을 위하여 GC는 랜덤값 $r \in \{0,1\}^t$과 의사 난수 생성기 G를 이용한다. 그 결과에 따라 모든 $1 \le t \le d-1$에 대해 각 노드 $p^t(v)$는 새로운 노드키 $k'_{p^t(v)} = \alpha(\beta^t(r))$을 갖게 된다. GC는 $T = \alpha(\beta^d(r))$을 생성하고 새로운 랜덤 그룹키 $K_{n+1} \in \{0,1\}^t$과 랜덤값 $R \in \{0,1\}^t$을 선택한다. GC는 다음과 같은 키-재생성 암호 메시지를 생성하고 이에 대한 서명값을 생성하며 구성원들에게 멀티캐스팅한다.

$$B^R(K_{n+1}) = [\Gamma, T \oplus F_S(R) \oplus R, E_{K \oplus F_S(R) \oplus R}(K_{n+1})]$$

$$\Gamma = \{E_{k_{\cdots}}(\beta(r)), E_{k_{\cdots}}(\beta^2(r)), \cdots, E_{k_{\cdots}}(\beta^t(r))\}.$$

이때, $1 \le t \le d-1$에 대해 $b'_j = b_j \oplus 1$이고 $s(p^t(v)) = b_0 \cdots b'_{d-t}$이고 이다.

각 구성원들은 새로운 그룹키 K_{n+1}을 생성한 후, Γ에서 사용된 각 암호화키 $k_{s(p^t(v))}$를 일방향 함수 H를 이용하여 $k'_{s(p^t(v))} = H(k_{s(p^t(v))} \| K_{n+1})$로 재생성한다.

V. 효율성

본 논문에서 제시한 변환 모듈은 매우 효율적이다. 사용자가 새로 가입하는 경우의 통신량은 기존의 방식들이 사용자 수에 따라 증가하는 것과 달리 고정된 통신량을 지니고 있다. 사용자가 탈퇴하는 경우에는 기존의 사용자 계산량에서 추가적으로 요구되는 계산 비용이 매우 적음을 알 수 있다.

본 절에서는 기존에 제안된 기법들과 Ⅳ장에서 제안된 변환모듈을 적용한 PRGT+의 효율성에 대해 비교하고자 한다. 효율성을 분석할 기존의 기법들은 McGrew와 Sherman이 제안한 OFT[6]과 Rafaeli et al.이 제안한 EHBT[9], 그리고 Canetti et al.이 제안한 PRGT[4]이다. 표 1에서는 사용자 한 명이 새로 가입하는 경우에 대해 살펴보고 표 2에서는 사용자 한 명이 탈퇴하는 경우를 살펴보도록 한다. 다음은 표 1과 표 2에서 사용되는 기호들이다.

d - 트리의 높이
K - 키의 비트 길이
I - 키 인덱스의 비트 길이 ($I < K$)
R- 랜덤 넘버 생성 비용
F - 의사난수 생성 또는 의사 난수 치환 비용
H - 일방향 해쉬 함수 생성 비용
X - XOR 계산 비용
E - 대칭키 암호화/복호화 계산 비용

표 1. 한 명의 사용자 가입

	계산량			통신량	
	GC	신규 가입자	사용자	유니 캐스트	멀티 캐스트
OFT	$R+d(H+X)$ $+(2d=1)E$	$(d+1)E$ $+d(H+X)$	$E+d\times$ $(H+X)$	$(d+1)K$	dK
EHBT	$R+(d+1)\times$ $(X+H+E)$	$(d+1)E$	$(d+1)\times$ $(X+H)$	$(d+1)K$	dI
PRGT	$(d+1)R$ $+(2d+1)E$	$(d+1)E$	dE	$(d+1)K$	dK
PRGT+	$R+(d+1)E$ $+2F+4\times$ $(H+X)$	$(d+1)E$	$F+2\times$ $(H+X)$	$(d+1)K$	$2K$

표 2. 한 명의 사용자 탈퇴

	계산량		통신량
	GC	사용자	멀티캐스트
OFT	$R+d(H+X+E)$	$E+d(H+X)$	dK
EHBT	$d(X+H)$ $+(d-1)E$	$d(X+H)$	dK
PRGT	$R+(d-1)E$ $+(d-1)F$	$E+dF$	dK
PRGT+	$R+(d-1)E$ $+d(F+H)+X$	$E+(d+1)F$ $+H+X$	$2K$

VI. 결 론

본 논문에서 제시한 변환 모듈은 매우 효율적이다. Join 알고리즘의 경우 기존 기법들의 Join 알고리즘과 독립적으로 구성될 수 있으며 그 계산량은 두 번의 XOR 연산량과 한번의 대칭키 암호 알고리즘의 복호화 연산, 한번의 해쉬 함수 연산량만을 필요로 한다. Leave 알고리즘의 경우 기존 알고리즘의 Leave 알고리즘의 연산량에 한번의 XOR 연산량과 한번의 대칭키 암호 알고리즘의 복호화 연산량이 추가되어 진다. 따라서 Join 알고리즘 측면에서는 기존 기법들의 연산량보다 적은 량의 연산량이 요구되어지고 Leave 알고리즘의 경우 추가로 요구되어지는 연산량이 매우 적음을 알 수 있다.

그러나 본 논문에서 제시하고 있는 변환 모듈의 경우 스테이트리스 리시버 성질을 제공하지 못하며 이런 성질을 지니고 있는 기존 기법들에 적용할 경우 그 성질을 보존하지 못하는 문제점을 지니고 있다. 향후 이에 대한 연구가 진행되어야 할 것이다.

참 고 문 헌

[1] S. Rafaeli, "A Decentralised Architecture for Group Key Management," *PhD appraisal*,url=citeseer.nj.nec.com/rafaeli00decentralised.html, Lancaster University, Lancaster, UK, September 2000.

[2] 권정옥, 황정연, 김현정, 이동훈, 임종인, "일방향 함수와 XOR을 이용한 효율적인 그룹키 관리 프로토콜 ELKH," 정보보호학회논문지, 12(6), 2002.

[3] 조태남, 이상호, "(2,4)-트리를 이용한 그룹키 관리," 정보보호학회논문지, 11(4), 2001.

[4] R. Canetti, J. Garay, G. Itkis, K. Micciancio, M. Naor and B. Pinkas, "Multicast Security: A Taxonomy and Some Efficient Constructions," *INFOCOM'99*, pp.419-428, 1999.

[5] G. Caronni, M. Waldvogel, D. Sunand and B. Plattner. "Efficient Security for Large and Dynamic Multicast Groups," *WETICE'98*, IEEE Comp Society Press, 1998.

[6] D. A. McGrew and A. T. Sherman, "Key Establishment in Large Dynamic Groups Using One-Way Function Trees," *Technical Report No. 0755, TIS Labs at Network Associates*, Inc., Glenwood, MD, May 1998.

[7] D. Naor, M. Naor and J. Lotspech, "Revocation and Tracing Schemesfor Stateless Receivers," *Crypto'01*, LNCS 2139, Springer-Verlag, pp.41-62, 2002.

[8] A. Perrig, D. Song and J. D. Tygar, "ELK, A New Protocol for Efficient Large-Group Key Distribution," *2001 IEEE Symposium on Security and Privacy*, May 2001.

[9] S. Rafaeli, L. Mathy and D. Hutchison, "EHBT: An Efficient Protocol for Group Key Management," *3rd Intl. COST264 Workshop on Networked Group Communication-NGC 2001*, LNCS 2233, Springer-Verlag, pp.159-171, 2001.

[10] L. R. Dondeti, S. Mukherjee and A. Samal, "A Dual Encryption Protocol for Scalable Secure Multicasting," *The Fourth International Symposium on Computer and Communications*, July 1999.

[11] S. Mittra, "Iolus: A Framework for Scalable Secure Multicastin," *ACM SIGCOMM'97*, pp.277-288, 1997.

[12] R. Molva and A. Pannetrat, "Scalable Multicast Security in Dynamic Groups," *The 6th ACM CCS*, pp. 101-112, 1999.

[13] C. K. Wong, M. G. Gouda and S. S. Lam, "Secure Group Communications Using Key Graphs," *The ACM SIG-*

COMM'98, pp.68-79, 1998.

[14] E. Bresson, O. Chevassut and D. Pointcheval, "Provably Authenticated Group Diffie-Hellman Key Exchange," *The 8th ACM Conference on Computer and Communications Security*, pp.255-264, 2001.

[15] E. Bresson, O. Chevassut and D. Pointcheval, "Provably Authenticated Group Diffie-Hellman Key Exchange-The Dynamic Case," *Asiacrypt '01*, LNCS 2248, Springer-Berlag, pp.290-309, 2001.

[16] E. Bresson, O. Chevassut and D. Pointcheva,. "Dynamic Group Diffie-Hellman Key Exchange under Standard Assumptions," *Eurocrpt'02*, LNCS 2332, Springer-Verlag, pp.321-336, 2002.

[17] M. Just and S. Vaudenay, "Authenticated Multi-Party Key Agreement," *Asiacrypt'96*, LNCS 1163, Springer-Verlag, pp.36-49, 1996.

[18] A. Mayer and M. Yung, "Secure Protocol Transformation via "Expansion" from Two-Party to Multi-Party," *ACM CCS'99*, pp.83-92, 1999.

[19] H. Kurnio, R. Safavi-Naini and H. Wang, "A Secure Re-keying Scheme with Key Recovery Property," ACISP '02, LNCS 2384, pp.40-55, Springer-Verlag, 2002.

〈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 : 고려대학교수학과 석사
2001년 9월~현재 : 고려대학교 정보보호 대학원 박사과정
〈관심분야〉 암호이론, 암호 프로토콜

이 수 미 (Su-Mi Lee) 학생회원
1995년 2월 : 순천향대학교 화학과 졸업
2001년 3월~2003년 2월 : 고려대학교 정보보호대학원 석사
2003년 3월~현재 : 고려대학교 정보보호대학원 박사과정
〈관심분야〉 암호이론, 암호 프로토콜, 유비쿼터스

이 동 훈 (Dong Hoon Lee) 정회원
1984년 : 고려대학교 경제학과 졸업
1987년 : Oklahoma Univ. 전산학과 석사
1992년 : Oklahoma Univ. 전산학과 박사
1993년~2001년 : 고려대학교 전산학과 교수
2000년~현재 : 고려대학교 정보보호 대학원 교수
〈관심분야〉 암호이론, 암호 프로토콜, 정보이론