Journal of Korea Multimedia Society Vol. 7, No. 7, July 2004(pp. 922-933)

피어 그룹을 위한 ID 기반의 그룹키 관리 프로토콜

박영호[†], 이경현[††]

요 약

최근 분산 시스템이나 협업 시스템을 위한 피어 투 피어(Peer-to-Peer) 네트워크에 대한 연구가 진행되고 있으며 암호학 분야에서는 pairing을 이용한 ID 기반의 공개키 암호 기법에 대한 연구가 활발히 이루어지고 있다. 본 논문에서는 동적 피어 그룹(Dynamic Peer Group, DPG) 멤버간의 안전한 그룹통신을 위한 ID 기반의 그룹키 관리 기법을 제안한다. 각 멤버들은 Private Key Generator(PKG)로부터 자신의 ID에 대한 공개키/개인키쌍을 발급 받으나, 그룹키를 관리하기 위한 중앙 관리개체를 이용하지 않고 멤버들간의 협력(collaboration)을 통해 자발적으로 그룹키를 관리함으로써 중앙 관리개체의 오류에 대한 문제(single-point of failure)를 예방할 수 있다. 그리고 동적 피어 그룹의 성질을 고려하여 멤버의 참여와 탈퇴에 대한 그룹키의 비밀성을 제공한다.

ID-Based Group Key Management Protocols for Dynamic Peer Groups

Young-Ho Park[†], Kyung-Hyune Rhee[††]

ABSTRACT

In recent years, peer-to-peer network have a greate deal of attention for distributed computing or collaborative application, and work of ID-based public key systems have been focusing on the area of cryptography. In this paper, we propose ID-based group key management protocols for secure communication in autonomous peer group. Each member obtains his public/private key pair derived from his identification string from Private Key Generator. No central server participates in group key management protocol instead, all group members share the burden of group key management by the collaboration of themselves, so that our scheme avoids the single point of failure problem. In addition, our scheme considers the nature of dynamic peer group such as frequent joining and leaving of a member.

Key words: Peer Group(피어 그룹), Group Key(그룹키), ID-Based System(신원기반 시스템)

1. 서 론

컴퓨터 네트워크 기술의 발전과 인터넷의 활용 영역이 다양한 분야로 확대되면서 정보의 교환과 처리를 위해 대부분 서버의 컴퓨팅 자원에 의존하던 환경에서 벗어나 서로 동등한 관계에서 개개인의 컴퓨팅 자원을 공유하고 정보를 교환하기 위한 피어 투 피어(Peer-to-Peer) 네트워킹에 대한 관심이 고조되고 있으며, 피어 투 피어 네트워킹을 이용한 분산 시스템이나 파일 공유, 협업 시스템과 같은 다양한 응용 분야에 대한 연구가 활발히 진행되고 있다. 분산 시스템이나 인스턴트 메신저 또는 협업 응용서비스 같은 경우, 동일한 관심사항이나 동일한 목적의 서비스를 제공하거나 제공받기 위해 사용자들간에는 피어 그룹(Peer Group)이라 부르는 자신들만의 커뮤니티(Community)를 형성할 수 있으며 그룹멤버들간의

※ 교신저자(Corresponding Author) : 이경현, 주소 : 부산광역시 남구 대연3동 599-1(608-737), 전화 : 051)620-6395, FAX : 051)626-4887, E-mail : khrhee@pknu.ac.kr
접수일 : 2003년 9월 9일, 완료일 : 2003년 12월 4일
[†] 준회원, 부경대학교 전자컴퓨터정보통신공학부
(E-mail : pyhoya@mail1.pknu.ac.kr)
[††] 종신회원, 부경대학교 전자컴퓨터정보통신공학부

안전한 통신을 요구하게 된다.

안전한 그룹 통신을 위해서는 그룹 사용자들간에 교환되는 정보에 대한 기밀성과 무결성 그리고 그룹 사용자에 대한 인증과 같은 보안 서비스들이 제공되어야 하며 이를 위한 핵심 요구사항으로 효율적이고도 안전한 키 관리가 이루어져 한다. 피어그룹의 경우 사용자들의 빈번한 그룹 가입과 탈퇴에 대한 동적인 성질을 가지게되며, 본 논문에서는 피어그룹의 동적인 성질을 고려한 그룹키 관리에 초점을 맞춘다.

그리고 최근 공개키 암호시스템 분야에서는 기존의 PKI(Public Key Infrastructure) 구조에서 발생하는 인증서의 발행과 취소, 저장과 같은 인증서 관리에 대한 부가적인 작업을 완화하기 위한 대체 시스템으로 pairing을 이용한 ID 기반의 공개키 암호시스템(ID-based public key cryptography, ID-PKC)에 대한 연구가 진행되고 있다. ID-PKC의 경우 사용자의 공개키는 인증서 형태로 제공되는 대신에 자신의 이메일 주소나 IP 주소와 같이 사용자와 연관시킬 수 있는 유일한 식별정보로(ID)부터 유도될 수 있으며 PKG 혹은 KGC(Key Generation Center)라 불리는 신뢰되는 개체에 의해 자신의 ID에 대한 개인키를 안전하게 발급 받게 된다. 사용자의 ID로부터 직접 그 사용자의 공개키를 계산할 수 있으므로 기존의 PKI 구조의 인증서 관리와 관련된 작업들의 부담을 줄일 수 있다.

본 논문에서는 pairing을 이용한 보다 효율적인 ID 기반의 그룹키 관리 기법을 제안한다. 2장에서는 관련연구로서 그룹키 관리와 ID 기반의 공개키 암호시스템의 핵심 연산인 pairing에 대해 간략히 소개하고, 3장에서는 제안하는 그룹키 관리 프로토콜들에 대해 기술한다. 4장에서는 제안된 시스템을 분석하고 5장에서 결론을 맺는다.

2. 관련연구

2.1 그룹키 관리

최근 수 년간에 걸쳐 다양한 그룹키 관리 기법들이 연구되어 왔으며, 이러한 그룹키 관리 기법들은 표 1에 나타내듯이 크게 그룹키 분배(distribution)와 그룹키 합의(agreement) 프로토콜로 구분될 수 있다. 그룹키 분배는 그룹키 관리 센터가 모든 그룹 멤버들간에 공유되는 그룹키를 생성하여 멤버들에게 안전하게 분배하는 중앙집중식 방법이며, 그룹키 합의는 중앙의 키 센터에 의존하지 않고 모든 멤버들이 자신의 공개정보를 제공하여 멤버들의 협력을 통해 그룹키를 설정하는 프로토콜이다. 그룹키 분배시스템 경우 키 분배센터의 능력에 따라 상당히 많은 수의 멤버들을 관리할 수 있지만 멤버들의 기여(contribution)와 협력으로 구성되는 그룹키 합의 프로토콜은 멤버들간에 여러 번의 통신 횟수와 많은 계산을 요구하므로 중소규모의 그룹에 적합하다.

그룹키를 관리함에 있어서, 통신량과 계산량의 비용을 줄이기 위한 방안으로 키-트리(key-tree)를 이용한 다양한 기법들이[1-4] 제안되었으며 키-트리를 그룹키 관리에 적용할 경우 통신량과 계산량을 $O(n)$에서 $O(\log n)$으로 줄일 수 있다. 현재까지 제안된 대부분의 그룹키 관리 프로토콜은 PKI 기반의 인증서를 교환함으로써 멤버의 인증을 수행하고 있다. 그러나 PKI 구조에서는 인증서의 발행과 취소, 인증서의 검증 그리고 디렉토리 관리 등으로 인해 시스템의 복잡성을 야기하며, 이러한 복잡성을 해결하기 위한 하나의 방안으로 최근 ID 기반의 공개키 암호시스템에 대한 연구가 이루어지고 있다.

안전한 그룹통신의 핵심은 그룹키의 관리이며, 그룹키를 관리함에 있어서 요구되는 보안 요구사항은 다음과 같다.

- 그룹키에 대한 인증(Group Key Authentication)

그룹 멤버들 이외에 어떤 악의적인 공격자가 그룹키를 도출하는 것이 계산상 불가능하여야 한다.

- 전방 비밀성(Forward Secrecy)

악의적인 공격자가 이전의 그룹키들에 대한 정보를 알고있더라도 이후의 그룹키를 계산하지 못 함으

표 1. 그룹키 관리 기법 분류

	그룹키 분배		그룹키 합의
키관리 유형	중앙집중식, 키관리 센터		분산형, 멤버들의 협력(Collaboration)
계산량	키센터	멤버	high (멤버들이 모두 동등)
	high	low	
특징	single communication round, 단일 지점 오류 가능		multiple communication round

로써 현재 데이터에 접근할 수 없어야 한다.

• 후방 비밀성(Backward Secrecy)

악의적인 공격자가 이후에 알려진 그룹키에 대한 정보를 가지고서 이전의 그룹키를 계산하지 못 함으로써 이전의 데이터에 접근할 수 없어야 한다.

• 키 독립성(Key Independency)

그룹키 집합 K의 적당한 부분집합 K'을 알고있는 공격자가 다른 어떠한 그룹키 $\overline{K} \in (K - K')$를 계산할 수 없어야 한다.

2.2 ID 기반의 암호시스템

ID 기반의 공개키 암호시스템(ID-PKC)은 Shamir[5]에 의해 처음 제안되었으며, 최근 Boneh와 Franklin에 의해 제안된 Weil pairing을 이용한 ID 기반의 공개키 암호 기법(Identity-based Encryption, IBE)[6] 이후로, ID 기반의 암호화뿐만 아니라 전자서명(Identity-based Signature)[7,8] 그리고 키 합의(Identity-based Key Agreement) 프로토콜[9-11]과 같은 다양한 암호시스템들이 활발히 연구되고 있다. 이 절에서는 최근 ID 기반의 암호시스템의 핵심 연산인 pairing에 대한 성질을 간략히 설명하도록 한다.

G_1과 G_2를 위수(order)가 큰 소수(prime number) q인 두 순환군(cyclic group)이라고 하면, 이 두 군에 대해 정의되는 bilinear 함수 $e: G_1 \times G_1 \rightarrow G_2$는 다음 성질들을 만족해야 한다[6].

1. Bilinearity : $P_i, Q_i \in G_1$와 $a, b \in Z_q^*$에 대해,
   $e(P_1 + P_2, Q_1) = e(P_1, Q_1) \cdot e(P_2, Q_1)$
   $e(P_1, Q_1 + Q_2) = e(P_1, Q_1) \cdot e(P_1, Q_2)$
   $e(aP_1, bQ_1) = e(P_1, Q_1)^{ab}$
2. Non-degeneracy : $e(P, Q) \neq 1$을 만족하는 $P, Q \in G_1$가 존재한다.
3. Computability : 모든 $P, Q \in G_1$에 대해 $e(P, Q)$를 계산할 수 있는 알고리즘이 존재한다.

Pairing은 위의 성질들을 만족하는 bilinear 함수이며, 암호학적인 용도로 사용하기 위해서 Tate pairing[12,13]과 Boneh와 Franklin에 의해 수정된 Weil pairing[6]이 연구가 되고 있다. 본 논문에서는 bilinear 함수 $e: G_1 \times G_1 \rightarrow G_2$는 Tate pairing이나 수정된 Weil Pairng을 가정한다.

pairing에서 G_1은 유한체상에서 정의된 타원곡선 E/F_p의 점들로 구성되는 덧셈군이며, G_2는 유한체 F_{p^k}에 대한 곱셈군으로 정의된다. Bilinear 함수 $e: G_1 \times G_1 \rightarrow G_2$는 G_1에서의 이산대수 문제가 G_2에서의 이산대수 문제가 될 수 있음을 의미하며[14], 본 논문에서는 보안 파라미터들이 G_1에서의 이산대수 문제의 어려움이 G_2에서의 이산대수 문제의 어려움을 만족하도록 선택되었다고 가정한다.

ID 기반의 키 합의 프로토콜은 세션키를 설정하기 위한 두 사용자와 각 사용자의 ID에 대한 공개키/개인키쌍을 발급하는 PKG로 구성된다. Boneh와 Franklin의 IBE 이후로 pairing을 이용한 ID 기반의 몇몇 키 합의 프로토콜이 연구되어왔다. Smart에 의해 pairing을 이용한 ID 기반의 키 합의 프로토콜[11]이 제안되었으며, Smart의 기법을 향상시킨 여러 프로토콜들이 제안되어 왔다[9,11].

2.3 ID 기반의 그룹키 합의 프로토콜

대부분의 그룹키 관리 프로토콜에서 멤버의 인증은 PKI 기반의 인증서를 교환함으로써 이루어지며, [14]의 AGKA-G와 [15]의 ID-AGKA에서 ID 기반의 공개키 시스템을 이용한 그룹키 관리 프로토콜을 제안하였다. AGKA-G[14]는 그룹 멤버들간의 상호인증을 위해 Günther의 ID 기반의 키 교환 프로토콜[16]을 적용하였고, ID-AGKA[15]는 Smart의 두 개체간의 키 합의 프로토콜[11]을 키-트리에 적용하여 그룹키 합의 프로토콜로 제안하였다

본 절에서는 ID-AGKA에 대해 간략히 설명하고 4장에서 제안기법과 비교하도록 한다. 그림 1과 같이 4명의 멤버들로 키-트리가 구성된 경우 ID-AGKA의 그룹키 합의 프로토콜의 수행과정은 표 2와 표 3과 같다. 표에서 s와 P_{PKG}는 PKG의 마스터 개인키와 공개키이며 P는 PKG의 공개 파라미터이고, Q_i

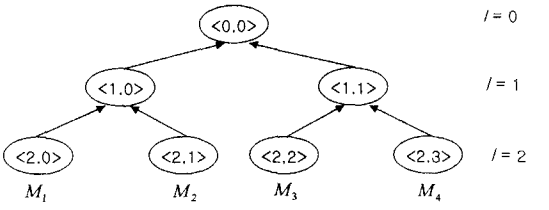


그림 1. $d=2$ 인 키-트리 예제

표 2. 그림 1에 대한 ID-AGKA : 1 라운드

M_1	M_2	M_3	M_4
공개키 Q_1에 대한 개인키를 PKG에게 획득 $S_1=[s]Q_1$	공개키 Q_2에 대한 개인키를 PKG에게 획득 $S_2=[s]Q_2$	공개키 Q_3에 대한 개인키를 PKG에게 획득 $S_3=[s]Q_3$	공개키 Q_4에 대한 개인키를 PKG에게 획득 $S_4=[s]Q_4$
랜덤값 $r_1 \in Z_q^*$을 선택	랜덤값 $r_2 \in Z_q^*$을 선택	랜덤값 $r_3 \in Z_q^*$을 선택	랜덤값 $r_4 \in Z_q^*$을 선택
$T_1=[r_1]P$ 계산	$T_2=[r_2]P$ 계산	$T_3=[r_3]P$ 계산	$T_4=[r_4]P$ 계산
T_1와 T_2를 교환		T_3와 T_4를 교환	
키 $K_{\langle 1,0\rangle}$를 계산	키 $K_{\langle 1,0\rangle}$를 계산	키 $K_{\langle 1,1\rangle}$를 계산	키 $K_{\langle 1,1\rangle}$를 계산

표 3. 그림 1에 대한 ID-AGKA : 2 라운드

M_1, M_2	M_3, M_4
공개키 $Q_{12}=H(Q_1+Q_2)$에 대한 개인키를 PKG에게 획득 $S_{12}=[s]Q_{12}$	공개키 $Q_{34}=H(Q_3+Q_4)$에 대한 개인키를 PKG에게 획득 $S_{34}=[s]Q_{34}$
$K'_{\langle 1,0\rangle}=f(K_{\langle 1,0\rangle})$를 계산	$K'_{\langle 1,1\rangle}=f(K_{\langle 1,1\rangle})$를 계산
$T_{12}=[K'_{\langle 1,0\rangle}]P$ 를 계산	$T_{34}=[K'_{\langle 1,1\rangle}]P$를 계산
T_{12} 와 T_{34} 를 교환	
$K_{\langle 0,0\rangle}$를 계산	$K_{\langle 0,0\rangle}$를 계산

와 S_i는 각각 멤버 M_i의 ID에 대한 공개키와 개인키이다.

M_1과 M_2은 다음과 같이 키 $K_{\langle 1,0\rangle}$를 계산하게 되며, 프로토콜이 성공적으로 수행되었다면 동일한 키를 계산하게 될 것이다.

$$M_1 : K_{\langle 1,0\rangle} = e([r_1]Q_2, P_{PKG}) \cdot e(S_1, T_2)$$
$$M_2 : K_{\langle 1,0\rangle} = e([r_2]Q_1, P_{PKG}) \cdot e(S_2, T_1)$$

표 3에서 $f()$ 는 $f: F_{q^k}^* \rightarrow Z_q^*$로 정의되는 함수이며, $K_{\langle 0,0\rangle}$의 계산은 다음과 같다.

$$M_1, M_2 : K_{\langle 0,0\rangle} = e([K'_{12}]Q_{34}, P_{PKG}) \cdot e(S_{12}, T_{34})$$
$$M_3, M_4 : K_{\langle 0,0\rangle} = e([K'_{34}]Q_{12}, P_{PKG}) \cdot e(S_{34}, T_{12})$$

위 프로토콜에서 보여지듯이, ID-AGKA 프로토콜에서는 키-트리의 중간노드에 개인키를 할당하기 위해 해당 노드의 하위에 있는 모든 멤버의 ID에 대한 공개키를 결합하고, 이 결합된 중간노드의 공개키에 대한 개인키를 PKG로부터 발급받아야 하므로 그룹키를 설정하기 위해 PKG가 그룹키 합의 프로토콜에 반드시 관여해야만 한다. 그러나 각각의 멤버들에 의해 자율적으로 구성되는 동적 피어 그룹 환경을 고려할 때, PKG와 같은 관리개체(central controller)가 그룹키 관리 프로토콜에 관여하는 것은 바람직하지 않을 수 있다. 만일 PKG에 시스템상의 오류가 발생하거나 PKG의 네트워크 단절이 발생하는 경우, 멤버들이 더 이상 그룹키 프로토콜을 수행할 수 없는 단일지점 오류(single point of failure)의 문제가 발생하기 때문이다.

3. 그룹키 관리 프로토콜 제안

이 장에서는 본 논문에서 제안하는 그룹키 관리 기법에 대해 설명하도록 한다. 제안하는 프로토콜은 AGKA-G의 프로토콜에서 Günther의 기법대신 pairing을 이용한 ID 기반의 키 합의 프로토콜을 적용하여 효율성을 증대시키고 ID-AGKA의 단일지점 오류문제를 해결하기 위한 기법으로, 중소규모의 자치 그룹(small to medium-sized autonomous group)에 적당한 그룹키 관리 프로토콜이다.

제안 방안에서 PKG는 단지 각각의 멤버에 대한

공개키/개인키 발급만 담당하고 그룹키 관리 프로토콜에는 참여시키지 않고 키-트리의 중간 노드와 연관되는 비밀키를 해당 노드의 왼쪽과 오른쪽 서브트리에 속한 멤버들에 의해 합의하여 분배하도록 함으로써 PKG의 관여 없이 오직 멤버들간의 협력을 통해 프로토콜을 수행할 수 있으며, 또한 PKG가 중간 노드에 대한 개인키를 계산할 필요가 없으므로 PKG의 연산에 대한 부담을 줄여주고 그룹 멤버들도 ID-AGKA에서 발생하는 PKG의 중간 노드에 대한 개인키 분배의 통신 오버헤드를 줄일 수 있다. 한편 계산상의 효율성을 위해 Smart 프로토콜의 보안성과 연산량을 개선시킨 Chen과 Kudla의 키 합의 프로토콜[9]을 적용한다.

3.1 표기 및 정의

본 절에서는 본 논문에서 사용되는 표기와 키-트리에 대해 간략히 설명한다. 제안 프로토콜에 사용되는 표기들은 표 4에 기술하였다.

그룹키 관리에 있어서 키-트리는 계산량과 통신량의 측면에서 효율적인 그룹키의 관리를 위해 많은 연구가 되어왔으며, 본 논문에서는 TGDH[3]의 키-트리 모델에 ID 기반의 키 합의 프로토콜을 적용한다. 그림 1은 키-트리의 예를 보여준다. 키-트리의 깊이를 d라고 할 때, 트리에서 최하위의 단말노드들은 레벨 d에 위치하고 최상위의 루트 노드는 레벨 0에 위치하며 각 중간 노드(intermediary node)의 비밀키는 하위 노드들의 비밀키값에 의해 상향식으로 계산되고 모든 멤버들이 공유하는 그룹키가 루트노드에 할당된다. 각 노드는 $\langle l, v\rangle$로 표기되며 노드의 비밀키 $k_{\langle l,v\rangle}$와 연관되어진다.

표 4. 프로토콜 표기

M_i	그룹 멤버; $i \in \{1, \ldots, n\}$
d	키-트리의 깊이(depth)
$\langle l, v\rangle$	트리의 l-레벨의 v번째 노드
$[x]Y$	점 Y에 대한 스칼라 x의 곱
e	Bilinear function
P	G_1의 생성자
P_{PKG}	PKG의 공개키
$s \in Z_q^*$	PKG의 개인키
Q_i	M_i의 ID에 대한 공개키
S_i	M_i의 개인키; $S_i = [s]Q_i$
r_i	M_i의 비밀 랜덤값
$k_{\langle l,v\rangle}$	노드 $\langle l,v\rangle$의 노드 키
$B_{\langle l,v\rangle}$	노드 $\langle l,v\rangle$의 블라인드 키
H_1, H_2	암호학적 해시함수 $H_1: \{0,1\} \rightarrow G_1^*$ $H_2: G_2 \rightarrow Z_q^*$

각 멤버 M_i는 단말노드와 연관되고, 노드 $\langle l, v\rangle$에 할당된 멤버 M_i는 $\langle l, v\rangle$부터 $\langle 0,0\rangle$까지의 키 경로[1]상에 있는 노드의 비밀키를 안다. 예를 들어, 그림 1에서 멤버 M_2는 $\{k_{\langle 2,1\rangle}, k_{\langle 1,0\rangle}, k_{\langle 0,0\rangle}\}$을 알고 있다.

3.2 그룹키 합의 프로토콜

이 절에서는 그룹 구성단계에서 그룹키를 생성하기 위한 그룹키 합의 프로토콜에 대해 설명한다. 본 논문에서는 그룹키 합의를 위해 이진 균형트리(balanced binary tree)를 사용하고 모든 멤버들은 동일한 키-트리 이미지를 가지며 각각의 멤버들이 어느 노드에 할당되어 있는지 알고 있다고 가정한다. 이때, 각 멤버들의 위치는 키-트리를 구성하기 전에 모든 멤버들이 자신의 ID를 서로 교환하여 ID의 사전적 순서(lexicographical order)에 의해 노드의 위치를 할당하는 방법을 사용할 수 있다.

PKG는 공개 파라미터들을 다음과 같이 생성한다; G_1상의 생성자 P를 선택하고 랜덤 비밀값 $s \in Z_q^*$를 임의로 선택하여 PKG의 공개키 $P_{PKG} = [s]P$를 설정한다. 이때 s는 PKG의 마스터 개인키가 되며 공개 파라미터 $< G_1, G_2, e, P, P_{PKG}, H_1, H_2>$를 공개한다. 각각의 멤버 $M_i(1 \le i \le n)$는 그룹에 참여하기 전에 자신의 공개키 $Q_i = H_1(ID_{M_i})$에 해당되는 개인키 $S_i = [s]Q_i$를 PKG의 인증을 통해 안전하게 발급 받는다. 이때 PKG의 파라미터들은 2.2절에서 언급한 것처럼 프로토콜의 안전성을 만족하도록 선택되었다고 가정한다.

그룹키를 설정하기 위해 각 멤버 M_i는 임의의 비밀값 $r_i \in Z_q^*$를 선택하고 $[r_i]P$를 계산하여 자신과 연관되는 최하위 단말노드의 블라인드 키로 할당한다. 프로시져-1은 키 $k_{\langle l,v\rangle}$의 설정과정을 보여준다.

1) 해당 멤버의 단말노드부터 루트노드까지의 경로를 키 경로로 정의한다[12].

만약 노드 $\langle l, v\rangle$가 단말노드이면($l=d$)이면 $k_{\langle l,v\rangle}$는 각 멤버들이 선택한 비밀 랜덤값이 된다. 즉, 노드 $\langle l, v\rangle$가 단말 노드이면 $k_{\langle l,v\rangle}$는 r_i가 되며, 블라인드 키 $B_{\langle l,v\rangle}=[r_i]P$가 된다. 만일 $\langle l, v\rangle$가 중간노드이면 $B_{\langle l,v\rangle}=[k_{\langle l,v\rangle}]P$가 된다. 프로시져-1에서 상위 노드의 키를 설정하기 위해 프로토콜을 수행하는 스폰서[2]는 각 서브트리의 가장 왼쪽에 위치한 단말노드의 멤버가 되며, 상대방의 공개키 Q_L이나 Q_R은 각자의 식별자 ID_L과 ID_R로부터 계산될 수 있다. 프로시져-1의 단계 5에서 키 교환이 성공적으로 이루어졌다면 두 멤버는 동일한 키 $k_{\langle l,v\rangle}$를 [9]에서 정의된 키 계산 알고리즘에 따라 계산하게 되고, 계산된 키를 각각 속한 서브트리의 루트노드의 키를 키 암호화 키(Key Encryption Key)로 사용하여 해당 서브트리에 속해 있는 멤버들에게 암호화하여 전달한다.

그림 1을 예로 들어 설명하면, 첫 라운드에서 멤버 M_1과 M_2에 의해 $k_{\langle 1,0\rangle}$가 합의되고 M_3와 M_4에 의해 $K_{\langle 1,1\rangle}$이 합의되며, 두 번째 라운드에서 $\langle 1,0\rangle$와 $\langle 1,1\rangle$을 루트노드로 하는 서브트리의 상위노드 키 $k_{\langle 0,0\rangle}$가 스폰서 M_1과 M_3에 의해 합의되고 해당 서브트리에 속한 멤버들에게 암호화되어 전달되며, 이 키는 안전한 그룹통신을 위해 모든 멤버들이 공유하는 그룹키로 사용된다. 프로시져-1에서 보여주듯이, 중간노드 $\langle l, v\rangle$의 키 $k_{\langle l,v\rangle}$의 계산은 두 자식노드 중에서 하나의 비밀 노드키와 형제노드(sibling node)의 블라인드 키를 필요로 한다.

ID-AGKA는 키-트리에서 중간노드의 공개키로 하부 멤버들의 ID에 대한 공개키들을 결합하여 생성하고, 결합된 공개키에 대한 개인키를 PKG로부터 직접 안전하게 발급 받아 중간노드의 키로 구성하였지만, 제안 프로토콜에서의 PKG는 그룹 구성단계에서 멤버들의 공개키/개인키 발급에만 관여할 뿐, 실질적인 키 합의 프로토콜에는 관여하지 않으며 키-트리의 중간노드에 대한 키의 계산은 각 서브트리의 멤버들에 의해 계산되어지고 전달된다. 그러므로 PKG의 중간노드에 대한 개인키 발급의 부담을 덜어줄 수 있으며 멤버들은 그룹키를 생성하기 위해 PKG에 의존할 필요가 없다.

2) 키 합의나 키 갱신 프로토콜을 수행하는 멤버를 지칭

프로시져-1 : Key agreement - $k_{\langle l,v\rangle}$

for each j ***step*** ($1 \le j \le d$);

1. $l = d-j$ (d : 키 트리의 깊이)
2. 각 서브트리의 가장 왼쪽 단말노드의 멤버가 스폰서로 지정됨

/** 왼쪽 서브트리의 스폰서 M_L과 오른쪽 서브트리의 스폰서 M_R.). 노드 $\langle l, v\rangle$의 왼쪽 서브트리의 루트 노드의 인덱스는 $v_L=2v$가 되고 왼쪽 서브트리의 루트 노드의 인덱스는 $v_R=2v+1$이 된다. */

if v_L(또는 v_R) == 단말노드

$B_{v_L}=[r_L]Q_L$ (또는 $B_{v_R}=[r_R]Q_R$)

else (i.e. v_L(또는 v_R) == 중간노드)

$B_{v_L}=[k_{v_L}]Q_L$ (또는 $B_{v_R}=[k_{v_R}]Q_R$)

/** 이때, k_{v_L}과 k_{v_R}은 각각 노드 $\langle l, v\rangle$의 왼쪽과 오른쪽 서브트리의 루트 키이고, Q_L과 Q_R은 M_L과 M_R의 공개키이다. */

3. $M_L \rightarrow M_R$: ID_L, , B_{v_L}
4. $M_L \leftarrow M_R$: ID_R, , B_{v_R}
5. 키 $K_{\langle l,v\rangle}$를 계산

   M_L : $k'_{\langle l,v\rangle}=e(S_L, B_{v_R}+[k_{v_L}]Q_R)$

   M_R : $k'_{\langle l,v\rangle}=e(B_{v_L}+[k_{v_R}]Q_L, S_R)$

   $k_{\langle l,v\rangle}=H_2(k'_{\langle l,v\rangle})$
6. M_L과 M_R은 새로운 키 $k_{\langle l,v\rangle}$를 서브트리에 있는 멤버들에게 브로드캐스트

   $M_L \rightarrow *_L$: $E_{k_{v_L}}(k_{\langle l,v\rangle})$

   $M_R \rightarrow *_R$: $E_{k_{v_R}}(k_{\langle l,v\rangle})$

3.3 멤버 참여(join)와 탈퇴(leave)

피어 그룹의 주요 특징은 그룹 멤버들의 빈번한 참여와 탈퇴로 인한 그룹의 동적인 성질이며, 동적 피어 그룹의 그룹키 관리 기법은 멤버들의 참여와 탈퇴에 대해, 새로 가입한 멤버에 대해서는 이전의 그룹 통신에 대한 비밀성을 보장할 수 있어야 하고 탈퇴하는 멤버에 대해서는 이후의 그룹 통신에 대한 비밀성을 보장할 수 있는 그룹키 관리를 지원해야 한다.

새로운 멤버가 그룹에 참여하는 경우 새 멤버가 이전의 그룹키를 획득하지 못하도록 함으로써 이전 메시지에 대한 비밀성을 보장하기 위해 추가된 새 멤버의 키 경로상에 있는 모든 노드의 키들을 갱신해야 한다. 어떤 멤버 M_{n+1}이 그룹에 참여하기를 원하

는 경우 M_{n+1}은 자신의 비밀 랜덤값 $r_{n+1} \in Z_q^*$을 선택하고 자신의 식별값 ID_{n+1}과 $B_{n+1}=[r_{n+1}]P$을 참여 요구 메시지와 함께 그룹으로 브로드캐스트 한다. 새로운 멤버를 키-트리에 추가하기 위해 현재 키-트리에서 가장 오른쪽의 가장 얕은 위치에 새 멤버에 대한 노드를 추가함으로써 키-트리 갱신에 대한 갱신 연산을 최소화하도록 한다. 3.2절에서 초기의 그룹키 합의 단계에서는 완전 이진트리 형태를 사용하여 키-트리를 구성하였으나, 세션중에 발생하는 멤버의 가입과 탈퇴에 대한 노드의 위치는 임의적인 순서로 발생하므로 멤버십 변경으로 인한 키-트리의 변경은 완전 이진트리 형태가 되지 않을 수 있다.

그림 2에서, 새 멤버 M_5는 노드 $\langle 1,1 \rangle$에 추가되어 노드 $\langle 2,3 \rangle$으로 할당되며 기존의 멤버 M_4는 노드 $\langle 2,2 \rangle$로 옮겨지고 새로운 비밀 랜덤값 r_4'를 선택하여 $[r'_4]P$를 노드 $\langle 2,2 \rangle$의 블라인드 키로 갱신한다. M_5의 키 경로상의 모든 키들이 새로 갱신되어져야 하므로, $k_{\langle 1,1 \rangle}$과 $k_{\langle 0,0 \rangle}$가 프로시져-2에 따라 갱신된다. 이때, M_5는 자신의 비밀 랜덤값 r_5를 선택하고 $[r_5]P$를 자신의 단말노드의 블라인드 키로 사용한다.

멤버가 그룹에서 탈퇴하는 경우, 전방 비밀성을 제공하기 위해 탈퇴하는 멤버가 알고있는 이전의 키들도 역시 갱신되어져야 한다. 그림 3과 프로시져-3은 멤버 탈퇴에 대한 키-트리의 갱신과 키 갱신 과정을 나타낸다. 키-트리의 갱신을 위해 탈퇴하는 멤버의 형제노드를 루트 노드로 하는 서브트리의 레벨을 한 단계 상승시킨다. 단계 3에서 호출되는 프로서져-1에서 서브트리의 루트노드가 단말노드인 경우 해당 멤버는 자신의 비밀 랜덤값을 새로 생성하여 단말노드의 블라인트 키를 갱신하여 키갱신 프로토콜을 수행하도록 한다. 예를 들어, 그림 3에서 멤버 M_3가 그룹에서 탈퇴하는 경우, 남아있는 멤버들은 노드 $\langle 2,1 \rangle$를 삭제하고 노드 $\langle 2,0 \rangle$를 루트노드로 하는 서브트리를 한 단계 위로 이동시킴으로써 키트리를 갱신하고 새로운 키 $k_{\langle 1,0 \rangle}$와 $k_{\langle 0,0 \rangle}$는 프로시져-3에 따라 갱신된다. 이때 M_1과 M_2는 자신들의 단말노드 $\langle 2,0 \rangle$와 $\langle 2,1 \rangle$에 각각 새로운 비밀값을 할당하며, 그룹키의 갱신과정에서 더 이상 탈퇴한 멤버 M_3의 공유값은 사용되지 않으므로 M_3는 새로운 그룹키를 계산할 수 없게된다.

프로시져-2. : 멤버 가입(Join)
1. M_{n+1} → Group : <join_request, ID_{n+1}, $B_{n+1}=[r_{n+1}]P$>
2. 모든 멤버들은 자신의 키트리를 갱신:
1) M_{n+1}을 가장 얕은 노드에 $\langle l, i \rangle$로 할당
2) 노드 $\langle l-1, \lceil \frac{i}{2} \rceil \rangle$의 기존 멤버의 노드 $\langle l, i-1 \rangle$로 변경
3) 노드 $\langle l, i-1 \rangle$의 멤버가 갱신된 키-트리의 이미지를 새 멤버 M_{n+1}에게 전달
3. M_{n+1}의 키 경로상의 모든 키를 갱신:
for ($1 \le j \le l$)
1) $l' = l-j$; $v = \lceil \frac{i-1}{2^j} \rceil$
2) $k_{\langle l',v \rangle}$를 프로시져-1에 의해 갱신

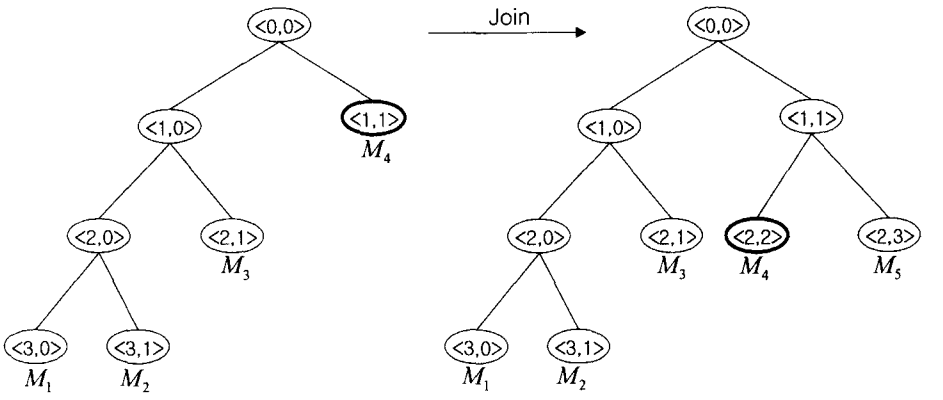


그림 2. 멤버 참여에 대한 키-트리 갱신

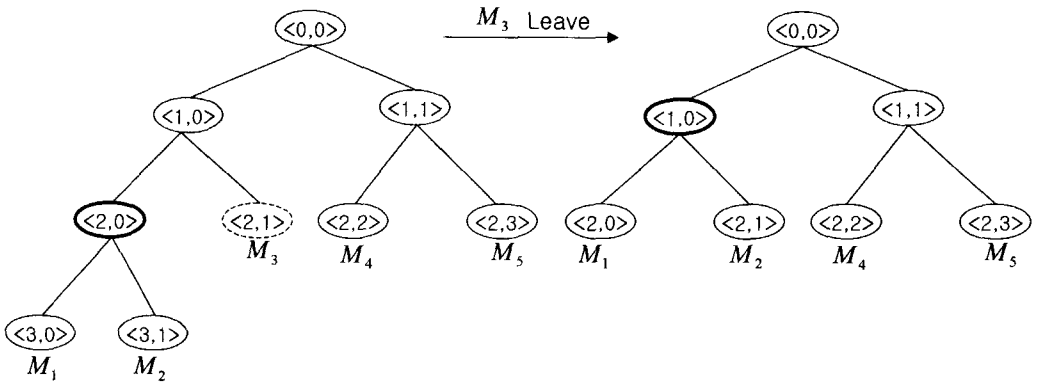


그림 3. 멤버 탈퇴에 대한 키-트리 갱신

프로시져-3. : 멤버 탈퇴(Leave)
1. 노드 $\langle l, i\rangle$의 멤버 M_j가 그룹을 탈퇴 2. 모든 멤버들은 자신의 키트리를 갱신: 2-1. 노드 $\langle l, i\rangle$를 삭제 2-2. 노드 $\langle l, i-1\rangle$의 서브트리의 레벨을 한 단계 위로 이동 3. M_j의 이전 키 경로상의 모든 키들을 갱신: *for* ($1 \le j \le l$) 1) $l' = l - j$; $v = \lceil \frac{i}{2^j} \rceil$ 2) $k_{\langle l', v\rangle}$를 프로시져-1에 의해 갱신 ; 이때 서브트리의 루트노드가 단말노드인 경우 해당 멤버는 새로운 비밀 랜덤값을 사용함

4. 제안 프로토콜 분석

4.1 안전성 분석

본 논문에서 제안된 프로토콜은 Chen과 Kulda에 의해 제안된 두 개체간의 ID 기반의 키 합의 프로토콜[9]을 키-트리 기법에 적용하여 그룹키 관리 기법으로 확장한 프로토콜이다. 따라서 키-트리에서 각 노드키에 대한 보안성은 [9]에서 분석된 ID 기반의 키 합의 프로토콜의 안전성을 따르게 되며, 그룹키의 보안과 관련하여 제안된 프로토콜은 키에 대한 묵시적 키 인증(implicit key authentication)과 전방 비밀성과 후방 비밀성을 제공한다.

묵시적 키 인증(implicit key authentication) : 수동적인 공격자(passive adversary)는 멤버들간에 교환되는 메시지를 통해 키-트리상의 어떠한 키도 획득할 수 없다.

제안 프로토콜에서의 묵시적 키 인증은 Diffie-Hellman 문제와 이산대수 문제[17]에 의해 제공된다. PKG의 마스터키 s가 노출되지 않았다고 가정할 때, 각 멤버들은 자신들의 임시 개인키로 비밀값 r_i를 선택하고 r_i는 멤버들간에 직접적으로 전달하지 않으므로 어떤 공격자가 멤버들간에 교환되는 메시지를 모두 수신했다고 할지라도 최하위 레벨 d에 위치한 노드의 키 $k_{\langle d, v\rangle}$를 계산할 수 없다. 보안 파라미터들이 올바르게 설정된 경우, 이산대수 문제의 어려움에 의해 Procedure-1의 공개값 $B_{v_L} = [k_{v_L}]Q_L$과 $B_{v_R} = [k_{v_R}]Q_R$로부터 하위 노드키 k_{v_L}이나 k_{v_R}을 계산할 수 없으며, 더욱이 공격자가 레벨 l과 d사이의 어떠한 중간노드의 키도 알지 못하면 다음 상위 레벨의 키를 계산할 수 없으므로 이산대수 문제와 Diffie-Hellman 문제로 인해 Procedure-1의 키 $k_{\langle l, v\rangle}$를 계산할 수 없다. 따라서 적어도 어떤 중간노드의 비밀키 $k_{\langle l, v\rangle}$를 알지 못하고서는 최상위 노드의 그룹키를 계산할 수 없다.

묵시적 키 인증에 의해 오직 프로토콜에 관여한 두 멤버만이 키를 계산할 수 있으므로 키에 대한 비밀성도 제공할 수 있게 된다. 또한 이산대수 문제와 Diffie-Hellman문제의 어려움을 가정할 때, 자신의 키 경로상의 노드를 제외한 다른 어떤 노드의 비밀키도 계산할 수 없으므로 키의 독립성도 제공하게 된다.

제안 프로토콜에서 키-트리는 상향식(bottom-up)으로 구성되고 각각의 멤버는 자신의 키 경로상에 있는 노드의 비밀키들만 알고 있으므로 멤버의 참여

와 탈퇴에 대해 키-트리와 키를 갱신함으로써 전방 비밀성과 후방비밀성을 제공할 수 있다.

후방 비밀성 : 새로운 멤버가 그룹에 참여하는 경우, 새 멤버는 이전 키-트리에 대한 비밀키를 획득할 수 없다.

새로운 멤버 M_{n+1}이 그룹에 참여하는 경우, M_{n+1}은 자신의 비밀 랜덤값에 대한 공개값 $B_{n+1}=[r_{n+1}]P$를 제공하게 되고, M_{n+1}의 키 경로상의 키들은 M_{n+1}가 제공한 값을 포함하여 갱신되게 된다. 그러므로 M_{n+1}이 수신하게 되는 키 경로상의 키들은 M_{n+1}의 공개값을 포함하여 갱신된 키들이므로 새로 가입한 멤버가 현재 획득한 키를 통해 이전의 키를 계산해내는 것이 어려우므로 키의 비밀성이 보장된다고 가정하면, M_{n+1}이 현재의 키를 이용해서 이전의 그룹 메시지에 접근할 수는 없다.

전방 비밀성 : 어떤 멤버가 그룹에서 탈퇴하는 경우, 탈퇴한 멤버는 새로 갱신된 비밀키를 획득할 수 없다.

노드 $\langle l, v\rangle$에 위치한 멤버 M_j가 그룹에서 탈퇴하는 경우 M_j의 노드 $\langle l, v\rangle$는 키-트리에서 삭제되며 $\langle l, v\rangle$의 부모 노드는 $\langle l, v\rangle$의 형제노드의 서브트리로 대체되고 M_j의 이전 키 경로상의 모든 키들은 새로 갱신된다. 이때 갱신된 키들은 M_j가 제공한 어떠한 정보도 포함하지 않으며, M_j가 현재 남아있는 멤버들간에 교환되는 정보를 통해 새로운 키를 계산할 수는 없다. 따라서 M_j는 더 이상 그룹 통신에 참여할 수 없게 된다.

키 독립성 : 그룹키 집합 K의 적당한 부분집합 K'을 알고있는 공격자가 다른 어떠한 그룹키 $\overline{K} \in (K - K')$를 계산할 수 없다.

어떤 멤버 M_k의 키 경로상의 키들($K_k \subseteq K$)이 공격자에게 노출되는 경우, M_k의 키 손상에 대한 그룹키 갱신은 일종의 M_k의 탈퇴에 대한 그룹키 갱신과 동일하게 취급될 수 있으므로 공격자가 획득한 키 집합 K_k로부터 갱신된 새로운 그룹키를 획득할 수 없다.

제안 프로토콜은 멤버의 참여와 탈퇴에 대해 약한(weak) 후방비밀성과 전방비밀성을 제공한다. 만일 PKG의 마스터 비밀키 s가 노출된 경우, s를 획득한 공격자는 $e(B_{v_l}, Q_{v_R})^s \cdot e(Q_{v_L}, B_{v_R})^s$를 계산함으로써 Procedure-1에서의 키 $k_{\langle l,v\rangle}$를 계산할 수 있게된다. ID 기반의 암호시스템에서는 PKG가 자신의 마스터 개인키 s를 이용하여 각 사용자의 ID에 대한 공개키/개인키쌍을 발급하므로 PKG의 마스터 개인키의 노출은 보안상 심각한 문제를 야기할 수 있으나, 이러한 문제에 대한 대처 방안으로 PKG의 마스터 키 s에 대해 비밀 분산법(secret sharing)을 이용하여 해결할 수 있으며, 제안 프로토콜은 PKG에 대한 마스터 비밀키 s의 노출로부터 비밀성을 제공하기 위해 [9]에서 제안된 Chen과 Kulda의 변형된 키 설정 프로토콜을 이용하여 해결할 수도 있다.

4.2 성능 분석

본 절에서는 제안 프로토콜을 Perrig에 의해 제안된 AGKA-G와 Reddy와 Nalla에 의해 제안된 ID-AGKA 프로토콜과 통신량과 계산량에 대한 성능을 비교 분석한다. 표 5는 n명의 그룹 멤버에 대한 그룹키 합의 프로토콜에 대한 비교 결과를 나타낸다.

통신량에 대한 비용은 그룹키를 설정하기 위해 교환되는 메시지의 개수로 평가되고, 계산량에 대한 비용은 그룹키를 계산하기 위한 주요 연산의 횟수로 평가한다. 표에서 $d=\lceil \log n \rceil$는 키-트리의 깊이를 나타낸다. AGKA의 주요 연산은 법-지수(EXP) 연산과 법-곱셈(MUL) 연산이며, ID-AGKA와 제안 프로토콜에서의 주요 연산은 pairing 연산(PR)과 타원곡선상의 점들에 대한 덧셈(PADD)과 스칼라 곱셈(PMUL) 연산이며, 특히 pairing의 연산이 많은 비중을 차지하게 된다.

통신비용에서 AGKA-G의 통신비용은 키 확인(key confirmation) 메시지를 포함한 비용이며 ID-AGKA와 제안 프로토콜에 키 확인 메시지가 추가되더라도 매 라운드당 한 번의 메시지가 추가되므로 AGKA-G보다 통신 비용에서는 효율적이다. ID-AGKA는 멤버의 송신 메시지는 적으나 중간노드의 ID에 대한 개인키를 해당 서브트리의 멤버들에게 제공하기 위해 PKG가 참여함으로써 PKG의 부가적인 연산과 메시지 전송이 요구되며, 수신 메시지의 처리에 있어서는 제안 프로토콜이 ID-AGKA보다 효율적이다. 또한 ID-AGKA는 키트리의 중간노드들에 대한 개인키가 반드시 요구되며 이 키들은 PKG가 생성해서 전달해야하므로 PKG에 시스템상에 오류가 발생하거나 통신의 단절은 그룹키 관리 프로토콜

자체를 수행할 수 없게 한다. 제안기법은 중간 노드에 대한 키를 PKG를 통해 획득하지 않고 그룹키 프로토콜 과정에서 멤버들간에 설정함으로써 PKG의 노드키 계산에 대한 부담을 줄일 수 있고 그룹 멤버들은 PKG로부터 중간 노드의 키를 수신해야 하는 통신 오버헤드를 줄일 수 있다.

표 5의 비교에서 암호화와 복호화는 대칭키 암호 시스템의 연산을 의미하며, 제안 프로토콜은 매 라운드에서 설정된 중간노드의 키를 프로토콜에 관여한 멤버가 서브트리에 속한 다른 멤버들에게 안전하게 전달하기 위해 암호화를 수행하고 다른 멤버들은 이를 복호화 하여 설정된 키를 획득하게 된다. ID-AGKA의 경우 멤버에 의한 암호화는 없으나 중간노드의 ID에 대한 개인키를 안전하게 해당 서브트리에 속한 멤버들에게 전달하기 위해 PKG에 의해 암호화가 요구되고 그룹 멤버들은 중간노드의 개인키를 얻기 위해 복호화를 수행한다.

제안기법은 매 라운드마다 각 서브트리에서 가장 왼쪽에 위치한 단말노드의 멤버를 스폰서로 선택하게 되므로 그룹키를 설정하기 위해 그룹 멤버는 모든 라운드(d 번)의 프로토콜에 관여하거나 혹은 최소한 1번만 관여하게 된다. 그러나 ID-AGKA의 경우 매 라운드마다 상위 노드의 키를 설정하기 위해 멤버들이 서브트리의 블라인드 키를 브로드캐스트를 통해 전달하여 모든 멤버들이 그룹키를 계산하는 방법을 사용하므로 표 5에서 나타낸 ID-AGKA의 연산은 모든 멤버들에게 똑같이 적용된다. 제안기법은 그룹키 관리를 멤버들에게 분담하는 측면에서 비용분담(cost sharing)의 특징을 가지지만 멤버들간의 협력에 의한 그룹키 합의는 여러 번의 통신 횟수($O(\log n)$)를 요구하므로 중소규모의 멤버들로 구성되는 그룹의 키 관리에 적당하다.

5. 결 론

본 논문에서는 중소규모의 동적 피어 그룹 환경을 위한 ID 기반의 그룹키 관리 프로토콜을 제안하였다. 제안 기법에서 PKG는 멤버들의 공개키/개인쌍의 발급만 담당할뿐 실질적인 그룹키 관리 프로토콜에는 관여하지 않고 멤버들의 자율적인 참여와 협력에 의해 키를 관리함으로써 PKG의 오류에 대한 단일지점 오류를 피할 수 있다. 또한 피어그룹의 동적인 성질을 고려하여 멤버의 가입과 탈퇴에 대한 동적인 그룹키 관리를 지원한다. 본문에서 그룹의 분할(Partition)과 병합(Merge)에 대해서는 언급하지 않았지만 제안기법도 멤버의 가입과 탈퇴에 대한 프로시져를 확장하여 기존의 키-트리의 분할과 병합을 동일하게 처리할 수 있다.

표 5. 프로토콜 성능 비교 분석

	AGKA-G [14]	ID-AGKA [15]		제안 기법	
Communication cost (min, max)					
라운드	d	d		d	
전송메시지 / M_i	$(2, 4d)$	$(1, d)$		$(1, 2d)$	
전송메시지 / PKG	-	$n-1$		-	
수신메시지 / M_i	$(d+3, 4d)$	$2d-1$		d	
Computation cost (min, max)					
주요 연산 횟수 / M_i	$(5, 5d)$ [EXP] $(2, 2d)$ [MUL]	PR	$(2, 2d)$	PR	$(1, d)$
		PMUL	$(2, 2d)$	PMUL	$(2, 2d)$
		PADD	-	PADD	$(1, d)$
암호화	$(1, d-1)$	$n-2$ (by PKG)		$(1, d-1)$	
복호화	$d-1$	$d-2$		$d-1$	

$d = \log n$: depth of key-tree
[EXP] : modular exponentiation, [MUL] : modular multiplication
[PR] : pairing, [PMUL] : point multiplication, [PADD] : point addition

최근에는 paring 연산의 성질을 이용하여 삼자간 (tripartite) 키 합의 프로토콜에 대해서도 연구가 되고 있으며 삼자간 키 합의 프로토콜을 키-트리에 적용하는 경우 트리의 레벨을 줄임으로써 그룹키 합의를 위한 키-트리의 계산량을 줄일 수도 있을 것이다.

참 고 문 헌

[1] D. Balenson, D. McGrew and A. Sherman, "Key management for large dynamic groups, One-way function trees and amortized initialization", IETF Internet Draft: draft-balensongroupkeymgmt-oft-00.txt, Feb. 1999.

[2] Y. Kim, A. Perrig and G. Tsudik, "Communication-Efficient Group Key Agreement", in Proceedings of the 16th international conference on Information security: Trusted information: the new decade challenge, IFIP-SEC 2001, pp.229-244, 2001.

[3] Y. Kim, A. Perrig and G. Tsudik, "Simple and Fault-Tolerant Key Agreement For Dynamic Collaborative Groups", 7th ACM Conference on Computer and Communications Security, pp.235-244, 2000.

[4] C. Wong, M. Gouda and S. Lam, "Secure Group Communications Using Key Graphs", ACM SIGCOMM 98, pp.68-79, 1998.

[5] A. Shamir, "Identity-based cryptosystems and signature schemes". In Advances in Cryptology-CRYPTO '84, LNCS 196, Springer-Verlag, pp.47-53, 1984.

[6] D. Boneh, M. Franklin, "Identity-based encryption from the Weil pairing", In Advances in Cryptology-CRYPTO 01, LNCS 2139, Springer-Verlag, pp.213-229, 2001.

[7] F. Hess, "Efficient identity based signature schemes based on pairings", in Proceedings of the Ninth Annual Workshop on Selected Areas in Cryptography. SAC 2002, pp.310-324. 2003.

[8] K. G. Paterson, "ID-based signatures from pairings on elliptic curves", Cryptology ePrint Archive Report 2002/004, available at http://eprint.iacr.org/2002/004/.

[9] L. Chen and C. Kudla, "Identity-based authenticated key agreement protocols from pairings", Cryptology ePrint Archive Report 2002/184, available at http://eprint.iacr.org/2002/184

[10] K. Shim, Efficient "ID-based authenticated key agreement protocol based on Weil pairing", Electronic Letters, Vol. 39, No.8, pp.653-654, 2003

[11] N. P. Smart, "An identity based authenticated key agreement protocol based on the Weil pairing", Electronics Letters, Vol. 38, pp.630-632, 2002.

[12] G. Frey, M. Muller, and H. Ruck, "The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems", IEEE Transactions on Information Theory, Vol. 45, No.5 pp.1717-1719, 1999.

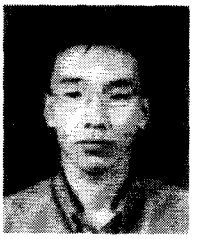
[13] S. Galbraith, "Supersingular curves in cryptography", in Advances in Cryptology-Asiacrypto '01, LNCS 2248, Springer-Verlag, pp.495-513, 2001.

[14] A. Perrig, "Efficient collaborative key management protocols for secure autonomous group communication", in Proceedings of the 1999 International Workshop on Cryptographic Techniques and E-Commerce (CrypTEC'99), pp.192-202, 1999.

[15] K. C. Reddy and D. Nalla, "Identity based authenticated group key agreement protocol", INDOCRYPT 2002, LNCS 2551, Springer-Verlag, pp.215-233, 2002.

[16] C. Günther, "An identity-based Key Exchange Protocol", Advances in Cryptology, EUROCRYPT 89, LNCS 434, Springer-Verlag, pp.29-37, 1989.

[17] A. J. Menezes, T. Okamoto and S. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field", IEEE Transaction on Information Theory, Vol. 39, pp.1639-1646, 1993.



박 영 호

2000년 2월 부경대학교 전자계산학과 이학사
2002년 2월 부경대학교 대학원 전자계산학과 이학석사
2002년 3월~현재 부경대학교 대학원 정보보호학과 박사과정
관심분야 : 네트워크 보안, 암호프로토콜, 암호 키 관리기술, Ad-hoc 네트워크

이 경 현

1982년 경북대학교 수학교육과 학사
1985년 한국과학기술원 응용수학과 이학석사
1992년 한국과학기술원 수학과 이학박사
1982년~1993년 3월 한국전자통신연구소 선임연구원
1993년 3월~현재 부경대학교 전자컴퓨터정보통신 공학부 부교수
관심분야 : 암호이론, 멀티미디어 정보보호, 네트워크 보안, 암호프로토콜, 재시도 대기체계론