안전한 멀티캐스트 통신을 위한 효율적인 그룹키 관리 프로토콜

준회원 이 현 종*, 정회원 김 진 철*, 오 영 환**

Effective group key management protocol for secure multicast communication

Hyeon-Jong Lee* *Associate Member*
Jin-Chul Kim*, Young-Hwan Oh** *Regular Members*

요 약

유니캐스트와는 달리 멀티캐스트 환경에는 상당히 많은 전송 링크가 존재하기 때문에 그 만큼 보안을 위협하는 요소들이 많이 존재한다. 따라서 안전한 멀티캐스트 통신을 위해서는 멀티캐스트 트래픽을 보호하는 메커니즘이 필수적이고 이러한 메커니즘의 핵심적인 요소 기술은 그룹키를 효율적으로 그룹 구성원들에게 분배하는 것이다. 멀티캐스트 환경에서 보안성을 제공하기 위해 최근까지 진행되어 오고 있는 연구들은 대부분 그룹키 분배에 관련된 것들이다.

본 논문에서는 멀티캐스트 환경에서 키 서버의 트래픽 집중을 효율적으로 분산시킬 수 있는 그룹키 관리 구조를 제안하였다. 제안한 프로토콜은 분산된 키 관리 구조와 서브 그룹 단위로 그룹키를 갱신한다. 시뮬레이션을 통하여 제안한 멀티캐스트 그룹키 관리 프로토콜을 기존의 연구들과 비교한 결과 가입 및 탈퇴 지연시간과 데이터 전송 지연 시간 측면에서 향상된 성능을 나타냄을 알 수 있었다.

Key Words : Group key; Multicast; Key Management.

ABSTRACT

Unlikely unicast transmission, there are many elements that threaten security. Thus, key management of creating and distributing group keys to authorized group members is a critical aspect of secure multicast operations. To offer security in multicast environment, the recent reserches are related to most group key distribution.

In this thesis, we propose a group key management protocol for efficient, scalable, and multicast operation. This proposed protocol architecture can distribute traffic centralized to the key server. since the group key rekeyed by sub-group manager. The detailed simulation compared with other group key management protocol show that the proposed group key management protocol is better for join, leave, and data latency.

* 현대 제이콤 기술 연구소 (hjlee@jcomm.co.kr), ** 광운대학교 전자공학부
논문번호 : 030189-0502, 접수일자 : 2003년 5월 2일

Ⅰ. 서 론

인터넷의 급속한 성장에 따라 인터넷의 사용 영역은 그 범위가 점차 확대되고 있다. 그에 따라 다양한 인터넷의 서비스가 요구되었고 그 중에서 데이타(Data), 영상(Video), 그리고 음성(Audio)을 특정 사용자 그룹에게 전송하는 멀티캐스트 데이터 전송기술은 음성 및 영상회의, 주문형 서비스(pay per view), CSCW (Computer-supported collaborative work) 그리고 주기적인 정보전달 등의 서비스를 가능하게 한다. 최근 인터넷상에서 이러한 멀티캐스트 데이터 전송에 대한 관심이 증가함에 따라 멀티캐스트 환경에서의 은밀한 데이터 전송에 대한 문제가 대두되게 되었다.

그러나, 멀티캐스트 통신은 그룹 억세스 관리나 그룹키 공유문제 등의 기술적 어려움으로 인해 유니캐스트 통신에 비해 보안성이 떨어진다. 현재 인증과 기밀성을 제공하기 위한 IPSec이나 양 종단간의 키 관리를 위한 ISKMP, Oakley, IKE 등 인터넷에 대한 대부분의 보안 표준들은 주로 하나의 송신자와 하나의 수신자 사이에서의 유니캐스트 데이터 전송에 관련된 것이어서 멀티캐스트 데이터 전송에는 적합하지가 않다. 따라서 폭넓은 확장성과 인터넷상에서 효율적인 기능을 제공하는 멀티캐스트 기술이 주요한 흐름으로 등장하기 위한 필수 조건은 안전한 멀티캐스트 데이터 전송을 보장하는 보안 기술 개발이 선행되어야 한다.

그룹 기반 보안에 대한 키 관리 문제는 오랫동안 연구되어 왔으나 이전에 대부분의 연구들은 계층적 구조나 컨퍼런스에 대한 암호학적인 접근에 초점을 맞추어 왔다[8,9,10]. 그러나 이러한 연구들의 대부분이 이론적인 한계를 가지고 있었으며 광범위한 환경에서의 구현에 대한 방법이 없었다. 최근에 실제적인 IP 멀티캐스트 환경에서 그룹키 관리에 대한 많은 제안들이 있었지만, 이러한 대부분의 제안들도 확장성이나 기타 여러 가지 구조적인 문제들을 가지고 있다[1,3,4,5,6].

따라서 본 논문에서는 멀티캐스트 환경에서 키 서버의 트래픽 집중을 효율적으로 분산시킬 수 있는 키 관리 구조를 제안하였다. 분산된 구조로 하나의 키 서버로 집중되는 트래픽과 부하를 분산시키며 서브 그룹단위로 그룹키를 갱신한다.

본 논문의 구성은 Ⅱ장에서 관련이론들을 그리고 Ⅲ장에서 기존의 연구들에 대해서 분석하고 Ⅳ장에서 멀티캐스트 환경에서의 효율적인 키 분배 프로토콜에 대해서 제안하였다. Ⅴ장에서는 제안한 이론을 시뮬레이션을 통해 검증하였고, Ⅵ장에서는 결론 및 향후 연구방향에 대해 제시하였다.

Ⅱ. 멀티캐스트 환경에서의 그룹키 분배

멀티캐스트 전송은 특정한 한 명 이상의 사용자 그룹에 똑같은 데이터를 전송하는 것을 말한다. 멀티캐스트는 유니캐스트에 기반한 그룹 통신에서처럼 n 명의 그룹 구성원에 대해 n번 전송하는 대신에 한번의 전송으로 모든 구성원에게 데이터를 전송할 수 있기 때문에 훨씬 효율적이다. 그러나 이러한 멀티캐스트 전송은 정보의 전송에 있어서 매우 효율적이지만 유니캐스트 전송에 비해 보안성이 떨어지기 때문에 특정한 사용자 그룹에 제한된 전송을 할 수 없다. 따라서 의도된 수신자만이 데이터를 수신할 수 있도록 보장하는 것이 가장 중요하다.

2.1 보안 메커니즘

현재 인터넷에서 지원하는 멀티캐스트 서비스는 특정한 수신자에게 제한된 데이터전송을 할 수 있는 어떠한 규정도 제공하지 못하고 있다. 어떠한 수신자도 IGMP(Internet Group Management Protocol) 메시지를 로컬 라우터에 보냄으로서 자유롭게 멀티캐스트 그룹에 가입과 탈퇴를 할 수 있다. 이러한 자유로운 멀티캐스트 환경에서 멀티캐스트 데이터의 흐름을 특정한 사용자 그룹으로 제한하기 위해서는 일반적으로 암호 알고리즘으로 알려진 보안 메커니즘 적용이 필수적이다. 즉, 송신자에 의해 암호화된 메시지는 그룹 멤버들에게 전송되고 오직 그룹 멤버들만이 암호화된 메시지를 해독할 수 있어야 한다.

암호 알고리즘은 전송되는 데이터를 암호화하여 기밀성을 제공하는 메커니즘이다. 송신자는 전송하고자 하는 데이터와 적당한 암호 키를 암호 알고리즘에 입력하고 암호화를 한 후, 암호문을 전송한다. 수신자는 수신된 암호문을 해당 복호키를 이용하여 암호문을 해독하게 된다.

암호 알고리즘은 대칭키 방식과 공개키 암호화 방식으로 크게 나눌 수 있다. 대칭키 암호화 방식은 암호키와 복호키가 동일한 방식이다. 반면에 공개키 암호화 방식은 암호키와 복호키가 동일하지 않은 개인키와 공개키로 구성된다. 그룹 통신에서는 대칭키 암호화 방식을 이용한다.

2.2 그룹 키 관리

은밀한 멀티캐스트 환경에서 그룹키를 그룹 구성원들에게 은밀하게 전송하는 키 관리 메커니즘은 가장 중요한 부분이다. 따라서, 키 관리 메커니즘은 멀티캐스트 보안 문제를 해결하기 위한 여러 제안들에 대한 효율적인 비교와 평가를 가능하게 하는 기준이 된다.

전송형태에 관계없이 그룹의 보안을 유지하기 위해서 적어도 하나 이상의 그룹 관리자가 존재해야 한다. 그룹 관리자는 사용자가 정당한 사용자인지를 확인하는 인증 기능과 그룹 구성원 이외의 사용자의 접근을 제한하는 접근제어 기능을 수행해야 한다. 또한, 그룹 구성원간의 키의 동기화를 위한 신뢰할 수 있는 키의 분배와 분배된 키의 보안성을 유지하기 위한 메커니즘을 수행해야 한다.

Ⅲ. 멀티캐스트 키 관리 프로토콜

현재 제안된 그룹키 관리 프로토콜은 크게 세 가지로 분류된다. 하나의 그룹 관리자가 존재하는 집중된 키 관리 구조, 전체의 그룹을 여러 개의 서브 그룹으로 나누고 각각의 서브 그룹에 그룹 관리자를 두는 분산된 서브 그룹 구조, 마지막으로 그룹 관리자가 존재하지 않고 모든 멤버들이 접근 제어와 그룹 키 분배를 수행하는 분산된 구조가 있다. 이번 장에서는 대표적인 몇 가지의 그룹키 관리 프로토콜을 살펴본다.

3.1 Iolus

Iolus는 Mittra에 의해서 제안되었다. Iolus에서 멀티캐스트 그룹은 독립적인 서브 그룹으로 구성된다. 각각의 서브그룹은 낮은 수준의 서브 그룹으로 나뉘어 서브 그룹의 계층이 형성된다. 예로, [그림 1]에서 멀티 캐스트 그룹 G는 4개의 서브 그룹으로 구성되어 있다.

Iolus는 전체 그룹에 대한 전역 DEK(Data Encryption Key)가 존재하지 않는다. 각각의 서브 그룹은 자신의 DEK를 가지고 새로운 멤버는 서브 그룹에 가입함으로서 자동적으로 전체 그룹에 가입할 수 있다. 그리고 단지 서브 그룹의 DEK만이 전송된다. 멤버가 멀티캐스트 그룹으로부터 제거될 때는 로컬 서브 그룹으로부터 제거하고 해당 서브 그룹의 DEK를 갱신하면 된다. 다른 서브 그룹에는 영향을 미치지 않는다.

각각의 서브 그룹에서 메시지는 서브 그룹의 DEK로 암호화되기 때문에 단지 해당 서브 그룹만이 이해할 수 있다. 이러한 암호화된 데이터를 다른 서브 그룹에 전송하기 위해서는 암호화된 데이터를 복호화하고 다시 목적지 서브 그룹의 DEK로 암호화되어야 한다.

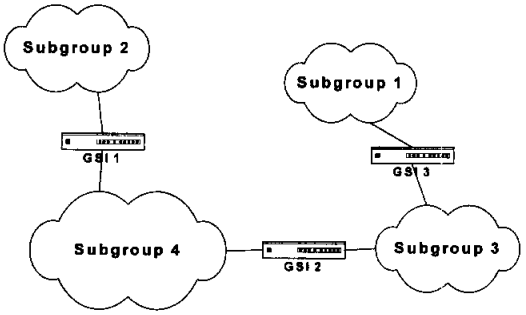

그림 1. Iolus 구조

이러한 작업은 Group Security Interfaces (GSIs)라고 하는 특정한 멤버들에 의해 수행된다. 각각의 GSI는 각각의 서브 그룹 사이에 위치하며 인접한 두 서브 그룹의 키를 모두 가지고 있다. 예를 들어 [그림 1]에서 GSI 1은 서브 그룹 2와 서브 그룹 4의 멤버이고 두 그룹 사이에서 변환 역할을 한다.

Iolus는 이해하기 쉽고 강력한 확장성을 제공한다. 실패의 한점(The single point of failure)이 존재하지 않으며 그룹 멤버들이 관리해야 할 키는 하나이다. Iolus는 또한 여러 문제점들을 가지고 있다. 첫째, 어떻게 서브 그룹을 형성할 것인가가 정의되지 않았다. 각각의 서브 그룹이 자체적으로 그룹을 형성하는게 어렵기 때문에 특정한 프로토콜과 툴이 필요하다. 둘째, 어떤 메커니즘은 각각의 사용자들이 가입하는 서브 그룹을 알 필요가 있다. 셋째, 만일 사용자가 다중 서브 그룹에 가입이 허용된다면, 멤버쉽의 동기화가 문제가 된다. 예를 들어, 한 멤버가 전체 그룹으로부터 추방된다면 모든 서브 그룹은 그 멤버를 배제해야 한다. 넷째, Iolus는 메시지를 모든 서브 그룹에 전송하기 위해 여러 번 암/복호화를 해야한다.

3.2 계층적 트리 구조

계층적 트리 구조에서, 전체 그룹은 하나의 DEK를 공유한다. 그러나 각각의 멤버들은 여러 개의 키를 기억해야 한다. 모든 키는 Key Server(KS)에

의해 생성되고 모든 키를 계층적으로 편성한다. DEK를 바꾸기 위해서 N개의 메시지 대신에 단지 $O(\log N)$ 키 갱신 메시지가 필요하다.

[그림 2]에서 Wallner가 제안한 계층적 트리 구조의 멀티캐스트 그룹은 16 멤버로 구성되어 있고, 각각의 멤버는 다섯 개의 키를 가진다. KS에서 모든 키는 다섯 개의 레벨로 분류된다. 이 경우, DEK는 루트 (Root)에서의 키이다(Key 15). 각각의 멤버는 자신과 KS에게만 알려진 Pairwise 키를 가진다. 또한, 멤버와 루트 사이의 네 개의 Key를 안다. 새로운 멤버가 그룹에 가입할 때, KS는 가입자를 서브 그룹에 위치시키고 KS는 다섯 개의 키를 전송
한다. 예를 들어, 만일 호스트 2가 가입을 한다면, 호스트 2는 key 1, 9, 13, 15와 자신의 유일한 Pairwise 키를 수신한다. 멤버가 그룹을 떠날 때, DEK는 변경이 필요하다.

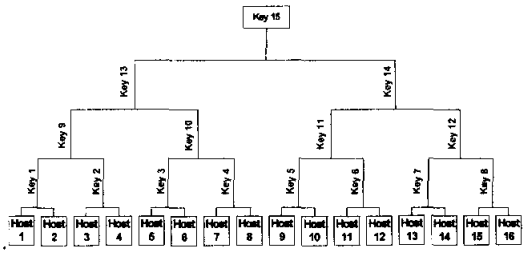

그림 2. 계층적 트리 구조

키 갱신을 수행하기 위해서 몇몇의 메시지가 전송되어야 한다. 예를 들어, 호스트 2가 그룹을 떠난다면, 호스트 2가 알고 있는 모든 키는 갱신되어야 한다. 우선 KS는 새로운 키 Key 1을 호스트 1의 Pairwise 키로 암호화하여 호스트 1에게 전송한다. 그런 다음 새로운 키 Key 9를 각각 Key 1과 Key 2로 암호화하여 해당 호스트에 전송한다. 여기서 Key 2로 암호화하는 것은 호스트 3과 호스트 4를 위해서이다. 다음, 같은 방식으로 키 Key 13을 각각 Key 9와 Key 10을 이용하여 해당 호스트에 전송한다. 마지막으로 그룹키 Key 15를 각각 Key 13과 Key 14를 이용하여 전체 그룹 멤버에게 전송한다. 그래서 키를 갱신하는데 전체적으로 7개의 메시지가 이용된다.

계층적 트리 구조는 키 갱신 메시지의 수를 N에서 $O(\log N)$으로 크게 줄일 수 있다. 그러나 각각의 사용자는 여러 개의 키를 가지고 있어야 한다. 보다 중요한 것은 실패의 한점을 피하지 못했다는 것이다. KS는 키 관리 프로세스의 모든 측면과 연관이 되어 있다. 이러한 특성은 확장성과 강건성에 중대한 영향을 미친다.

3.3 Core Based Tree

Core Based Tree (CBT) 구조는 A. Ballardie에 의해서 제안되었다. 이 방식에서 그룹 생성자 (Group Initiator)는 코어 (Core) 의 멤버를 설계한다. 또한, 이러한 코어를 연결하는 라우팅 트리를 코어 트리 (Core Tree)라고 한다.

[그림 3]은 CBT 멀티캐스트 그룹을 나타낸다. 각각의 코어는 그룹 생성자로부터 멤버 포함 목록 (Member Inclusion List)와 멤버 배제 목록 (Member Exclusion List)를 포함하는 멤버쉽 제어 정보 (Membership Control List)를 수신한다. 코어 트리가 설정된 후, 정당한 멤버는 여러 코어 중 하나의 코어에 가입 요청을 하여 그룹에 가입할 수 있다. 가입 요청이 승인되면 멤버와 코어 트리에 연결된 패스는 그룹 멀티캐스트의 새로운 부분으로 추가된다. 계층적 트리 구조와 유사하게 이 방법은 KDC (Key Distribution Center)를 이용하지만 방법은 다르다. 그룹이 생성될 때, KDC는 두 개의 새로운 키 DEK (Data Encryption
Key)와 KEK (Key Encryption Key)를 생성한다. 새로운 멤버가 가입할 때마다 이러한 두 개의 키가 전송되고 DEK를 갱신하기 위해서 KDC는 KEK로 암호화 한 새로운 DEK를 그룹 멤버들에게 전송한다. 미리 정의된 시간 간격이 지나면 멤버들은 새로운 키를 메시지를 암호화하는데 적용한다.

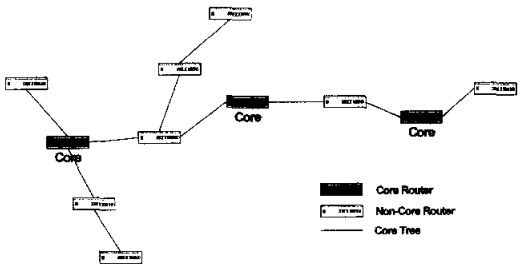

그림 3. Core Based Tree 구조

CBT는 코어 트리 상의 멤버들이 멤버쉽 인증과 키 분배를 수행하기 때문에 매우 확장적이다. KDC는 새로운 키를 생성하도록 요구되고 여러 개의 KDC가 존재할 경우, 실패의 한점의 문제는 피할 수 있다.

그러나, CBT의 사용에 약간의 제한이 있다.

CBT는 모든 소스로부터 코어로 향한 트래픽이 동일한 링크를 이용하기 때문에 코어 근처에서 트래픽 집중현상이 발생한다. 또한, 이러한 병목현상은 데이터 전송의 지연을 발생하기 때문에 멀티미디어 데이터 전송에 심각한 문제를 야기할 수 도 있다.

Ⅳ. 제안한 멀티캐스트 키 분배 프로토콜

3장에서 설명한 Iolus, 계층적 트리구조, CBT구조에서 의 키관리 문제점을 해결하기 위해 본 논문에서는 새로운 그룹키 관리 프로토콜을 제안한다. 제안한 방식은 분산된 구조를 가지고 있기 때문에 하나의 키 서버로 집중되는 트래픽과 부하를 분산시켰으며 하나의 키 서버를 사용하는 방식에서 단일서버로 인해 발생할 수 있는 문제점들을 해결하였다. 또한 네트워크 상으로 그룹키를 직접적으로 전달하지 않기 때문에 전송중에 그룹키가 노출될 수 있는 문제점을 해결하였다. 제안한 프로토콜의 네트워크 구성도는 [그림 4]와 같다.

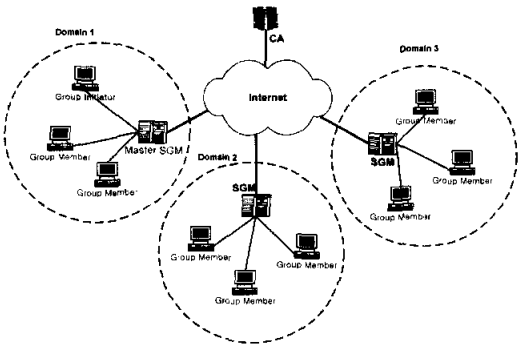

그림 4. 제안한 알고리즘의 네트워 구성도

4.1 멀티캐스트 그룹 키 및 그룹 생성 프로토콜

4.1.1 그룹 키 생성

그룹키 생성은 초기 그룹을 생성할 때와 가입과 탈퇴가 발생했을 때, 가입과 탈퇴가 발생한 해당 SGM으로부터 전송된 IV(Initial Vector)와 해당 SGM의 식별자, SGM_ID를 이용하여 다음과 같은 방식으로 생성된다. 각각의 SGM에서 생성된 그룹키는 서브 그룹 사용자들에게 전송한다. 네트웍 상으로 직접 그룹 키 KG를 전송하지 않기 때문에 보다 은밀하게 그룹 키를 갱신할 수 있다. 그룹키의 생성은 다음과 같은 방식으로 생성한다.

$$K_G = E_{SGM\_ID}(IV)$$

여기서, E는 암호 알고리즘이다. SGM_ID는 키 갱신이 일어나는 SGM의 식별자이고 IV는 임의의 난수이다. 보안 알고리즘은 어플리케이션의 요구에 따라 적절하게 선택할 수 있다.

4.1.2 그룹 생성 프로토콜

멀티캐스트 그룹을 생성하고자 하는 그룹 생성자는 그룹 생성자가 속해있는 도메인의 SGM에게 그룹 생성 요구 메시지와 인증에 관련된 정보를 전송한다. 그룹 생성자가 속해있는 도메인의 SGM은 (Master) SGM이 되고 SGM을 관리하는 기능을 가진다. 그리고 차후에 그룹 해제의 권한을 갖는다. [그림 5]는 그룹 생성자와 SGM간에 상호 인증을 통해 키를 생성하는 프로토콜이다.

(1) GI → (Master) SGM : make group request,
Group Address, ACL
(2) GI ↔ (Master) SGM : authenticate GI
(3) (Master) SGM : generate a new subgroup key K_S
distribute subgroup key K_S
randomly generate IV
generate a new group key K_G with IV, SGM_ID
randomly generate a new key
encryption key (KEK)
(4) (Master) SGM → GI : $\{K_G\}_{K_S}$

그림 5. 그룹 생성 프로토콜

4.1.3 SGM 등록

SGM이 서브 그룹으로부터 최초 join request 메시지를 받았을 경우 (Master) SGM에 등록되지 않은 SGM은 등록 절차를 거쳐야 한다. [그림 6]은 (Master) SGM과 SGM간의 등록 프로토콜을 나타낸다. 인증 절차가 수행되면 (Master) SGM은 KEK를 IKE나 기타 키 교환 알고리즘을 이용하여 안전하게 SGM에게 전송하고, 사용자 등록에 따른 그룹키 갱신을 위해 생성한 SGM_ID를 KEK로 암호화 하여 전송한다.

(1) SGM → (Master) SGM : *registration request*
(2) SGM ↔ (Master) SGM : authentication SGM,
Distribute KEK
allocate SGM Identification Number(SGM_ID)
(3) (Master) SGM → SGM : $\{IV, \{SGM\_ID\}\}_{KEK}$

그림 6. SGM 등록 프로토콜

4.2 그룹 가입 및 그룹 키 갱신 프로토콜

새로운 가입자가 발생하면 새로운 가입자는 기존의 데이터에 접근해서는 안된다. 이러한 접근을 방지하기 위해 가입자가 발생시 SGM은 그룹키를 갱신해야 한다. [그림 7]은 그룹 가입 및 키 갱신 절차를 나타내고 있다.

(1) U → $SGM_{(x)}$: join request, Group Address
(2) U ↔ $SGM_{(x)}$: authentication U and Distribute K_S
(3) $SGM_{(x)}$: randomly generation IV
(4) $SGM_{(x)}$ → {SGM} : key_update_x, $\{IV\}_{KEK}$
(5) {SGM}: generate a new group key K_G with IV, $SGM\_ID_{(x)}$
(6) {SGM} → {U} : $\{K_G\}_{K_S}$

그림 7. 그룹 가입 및 그룹키 갱신 프로토콜

4.3 그룹 탈퇴 및 그룹 키 갱신 프로토콜

그룹의 가입자가 탈퇴할 때도 마찬가지로 탈퇴자가 그룹키와 서브 그룹 키를 이미 알고 있기 때문에 탈퇴 후, 접근이 가능하므로 그룹 키와 서브 그룹키를 동시에 갱신해야 한다. [그림8]은 멤버의 요청에 의한 탈퇴 및 그룹키 갱신 절차를 나타낸다.

(1) U → $SGM_{(x)}$: $\{leave\ request\}_{K_S}$
(2) $SGM_{(x)}$ → U : $\{leave\ granted\}_{K_S}$
(3) $SGM_{(x)}$: randomly generate IV randomly generate a new subgroup key K_S
(4) $SGM_{(x)}$ → {SGM} : key_update_x, $\{IV\}_{KEK}$
(5) {SGM} : generate K_G with IV, $SGM\_ID_{(x)}$
(6) {SGM} → {U} : $\{K_G\}_{K_S}$
$SGM_{(x)}$ → {U} : $\{K_G, K_S\}_{K_U}$
* K_U는 사용자 공개키

그림 8. 그룹 탈퇴 및 그룹키 갱신 프로토콜

V. 성능평가 및 고찰

본 장에서는 제안한 멀티캐스트 키관리 기술을 Iolus, 계층적 트리 구조 키관리 기술과 같은 다른 멀티캐스트 키 관리 기술들과 성능 트레이드 오프(Trade-off) 관계를 평가하였다.

5.1 성능 평가 기준

제안한 멀티캐스트 그룹키 관리 프로토콜은 멤버의 가입과 탈퇴의 빈도에 따른 가입 및 탈퇴의 지연시간과 데이터 전송 지연시간을 성능 평가의 기준으로 했다. 이러한 평가 기준은 그룹키 관리 프로토콜의 구체적인 수행능력을 판단하는 기준이 된다. 또한, 그룹의 멤버 수가 증가함에 따라 그룹 가입 및 탈퇴 빈도수가 늘어나기 때문에 확장성 측면까지 평가할 수 있다.

표 1. 메시지 및 용어 정리

용 어	내 용
Mk_Grp	그룹 생성 요청 프로토콜
Grp_ Addr	그룹 주소
Join_Req	가입 요구 메시지
Leave_Req	탈퇴 요구 메시지
Leave_grant	탈퇴 승인 메시지
key_update_x	특정 SGM으로부터의 그룹키 갱신을 알림
ACL	접근 제어 목록 (Access Control List)
SGM(x)	이벤트가 발생한 특정 SGM을 지칭
SGM_ID(x)	특정 SGM의 식별자
GI	그룹 생성자 (Group Initiator)
KEK	SGM 간의 메시지 암호화에 이용되는 키
K_G	그룹키
K_S	서브 그룹키

표 2. 성능 평가에 이용된 파라메터

파라메터	입력값
Encryption algorithm	DES3
Key length	168 bits
Key encryption time	1.64 ms
Key generation time	2.8 ms
Data packet size	500 bytes
Signature algorithm	RSA
Signature length	512 bits
Authentication rate	1506 KB/s
Signature rate	367 KB/s

가입 및 탈퇴 지연 시간은 가입자나 탈퇴자가 가입 및 탈퇴 요구 메시지를 전송하고 키를 수신하는데 소요되는 시간이고 다음과 같은 사항들을 포함한다.

- 요구(Request)와 응답(Response)에 관련된 패킷 에 대한 네트웍 지연
- 서버에서의 요구의 수신과 인증에 대한 계산, 새로운 키의 생성과 암호화, 메시지 Digest의 생성, 응답에 대한 서명과 그리고 멤버에게 전송하는 시간과 관련된 지연
- 서버에서의 큐잉(Queueing) 지연
- 요구와 응답의 손실로 인한 지연

데이터 전송 지연 시간은 소스(Source)로부터 전송된 데이터가 그룹 멤버들이 수신할 때까지의 평균 네트웍 지연이다. 제안한 메커니즘과 계층적 트리 구조에서 멀티캐스트 데이터 전송은 프로토콜과 무관하다. 그러나 Iolus에서, 소스에서 전송된 각각의 패킷은 서브 그룹 관리자에 의해 서브 그룹키로 다시 암호화되어 서브 그룹 멤버들에게 전송된다. 이러한 추가적인 지연은 데이터 패킷 지연에 반영되어야 한다.

5.2 성능 평가 환경

성능 평가는 Berkeley의 Event-based Network Simulator ns-2[14]를 이용했고 성능 평가에 사용될 토폴로지는 Tiers 네트웍 토폴로지 제너레이터[15]를 사용했다. 참고로 Tiers는 WANs, MANs 그리고 LANs과 같은 3 레벨의 계층적 네트웍을 생성하고 가장 현실성 있는 네트웍 토폴로지를 생성하는데 많이 이용된다[16]. 성능 평가에 사용된 네트웍은 하나의 WAN, 10개의 MAN 그리고 50개의 LAN에 걸쳐 총 360개의 노드로 구성되어 있다. 각각의 LAN은 5개의 호스트를 포함한다. WAN, MAN 그리고 LAN 링크의 대역폭은 각각 2.5 Gb/s, 155 Mb/s 그리고 100 Mb/s이다. 평균 링크 전송 지연은 WAN 링크에 대해 60 ms, MAN 링크에 대해 17 ms 그리고 LAN에 대해 1ms이다. 성능 평가는 멀티 캐스트 어플리케이션에 관련된 트래픽의 흐름과 그룹키를 관리하기 위한 제어 트래픽 만을 고려했다. 큐잉 지연을 고려하기 위해 각각의 패킷은 노드에서 0 에서 2 ms 사이의 균일 분포(Uniform distribution)를 가지는 지연을 가진다.

멀티캐스트 환경을 영상 및 음성회의에 적합한 N-to-N 시나리오로 가정했다. 멀티캐스트 그룹 멤버들은 각각의 LAN에 분포되어 있다. 가입자들은 각각 독립적으로 그룹 멤버 상태와 그룹 멤버가 아닌 상태를 교차한다. 이러한 상태들에 머물러 있는

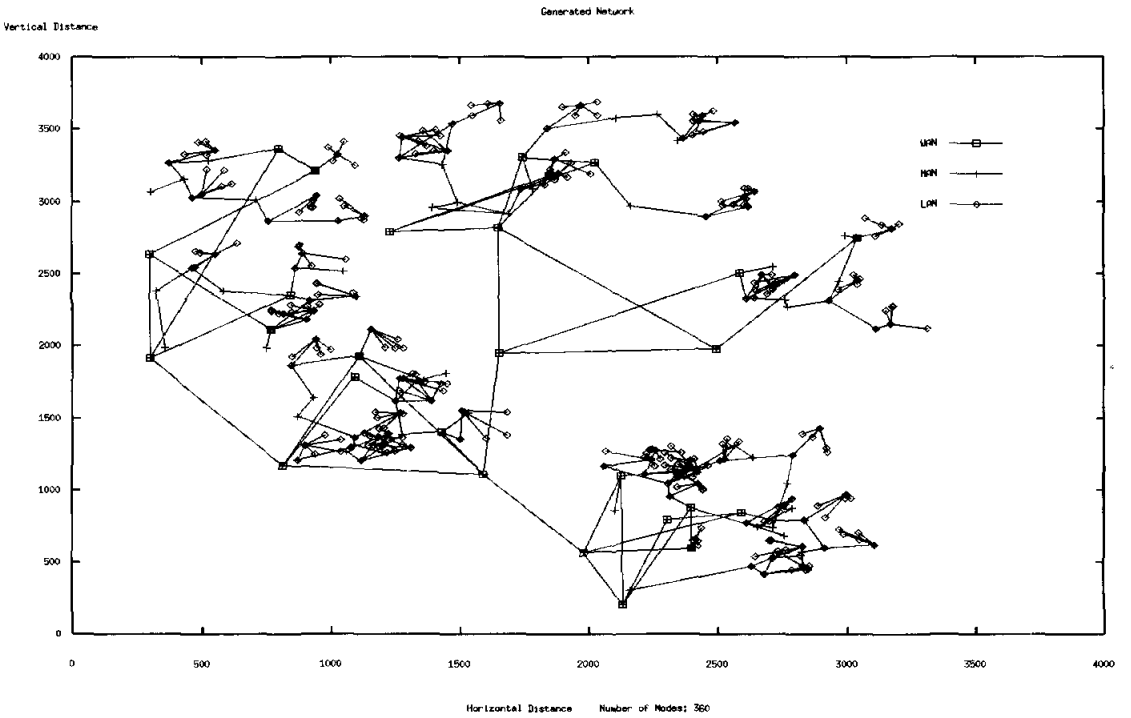

그림 9. 성능평가에 이용된 네트웍

시간을 지수적으로 분포되어 있다고 가정했다. 또한, 그룹 가입과 탈퇴의 빈도에 따른 지연시간을 측정하기 위해 호스트가 그룹의 멤버인 시간($1/\mu$)을 2초부터 12초까지로 설정했다. 그룹 멤버가 많아질수록 키 갱신 빈도수가 늘어나는데 광범위한 네트웍에 대한 측정의 어려움으로 인하여 네트웍 크기를 줄이는 대신 키 갱신 빈도를 늘렸다. 성능 평가 시간은 70초로하고 시작 10초 후부터 측정을 시작했다. 하나의 CBR 소스 (Constant Bit Rate Traffic source)는 224Kb/sec로 그룹에 데이터를 전송한다. 전송자의 멤버 비율은 10% ~ 40%이고 지역 전체에 균일하게 분포한다. 각각의 CBR 소스는 Sending state와 Quite state를 교차하고 각각의 상태에서 0 ~ 12초 사이에 균일하게 분포되어 있다.

성능 평가를 실행하기 위한 파라메터들은 [표 2]과 같다. 사용된 파라메터값은 Chang, Schneier[17, 18]가 보고한 결과값을 근거로 하고 있다.

5.3 결과 및 고찰

본 절에서는 Iolus, 계층적 트리구조 그리고 제안한 멀티캐스트 그룹키 관리 프로토콜을 5.2에서 기술한 성능 평가 환경에 기반하여 실험을 실시하고 그 결과를 분석하였다.

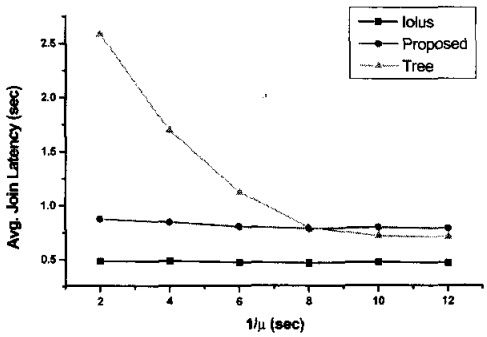

그림 10. 가입 지연

5.3.1 가입 및 탈퇴 지연 시간

[그림 10]에서 그룹키 갱신 빈도 수에 대해 가입자가 그룹을 가입하는데 소요되는 가입 지연 시간을 그래프로 나타내었다. 계층적 트리 구조는 Iolus나 제안한 프로토콜에 비해 가입 지연 시간이 크다. Iolus나 제안한 프로토콜의 가입 지연 시간은 그룹키 갱신 빈도 수에 거의 영향을 받지 않는다는 것을 알 수 있고, 제안한 방식이 다른 서브 그룹의 SGM에게 IV(Initial Vector)를 생성하여 전송하기 때문에 가입 지연이 Iolus보다 크게 나타난다. 또한, 그룹키 갱신 빈도수가 증가함에 따라 계층적 트리 구조는 지연 시간이 큰 폭으로 증가하고 그룹키 갱신 빈도수가 낮아질수록 그래프는 수평에 가까워짐을 알 수 있다. 이러한 특성은 계층적 트리 구조가 안정성 면에서 떨어진다는 것을 보여 준다.

낮은 그룹키 갱신 빈도수에서 제안한 방식이 가장 나쁜 특성을 보이나 성능을 판별할 만큼의 큰 차이는 아니다. 이 그림에서 주목해야 할 점은 그룹키 갱신의 빈도수가 높을 때이다. 계층적 트리구조는 상당히 큰 가입 지연을 보인다. 그러나 본인이 제안한 방식과 Iolus는 갱신 빈도수에 거의 영향을 받지 않는다는 것을 알 수 있다. 이것은 확장성이 뛰어나다는 것을 의미한다.

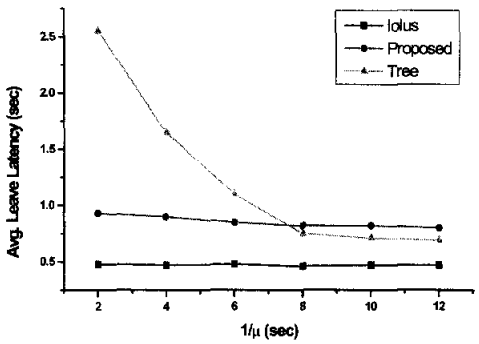

그림 11. 탈퇴 지연

[그림 11]에서는 그룹키 갱신 빈도 수에 따라 그룹 멤버가 그룹을 탈퇴하는데 소요되는 탈퇴 지연 시간을 그래프로 나타내었다. [그림 11]에서 보는 바와 같이 탈퇴 지연시간은 가입 지연 시간과 거의 유사함을 알 수 있다. [그림 10]에서처럼 계층적 트리 구조가 탈퇴 지연시간이 가장 크고 그룹키 갱신 빈도수가 증가함에 따라 역시 지연 시간이 큰 폭으로 증가한다. 탈퇴 시, 서브 그룹키를 갱신해야 하는 점 때문에 제안한 프로토콜의 탈퇴 지연 시간이 가입 지연 시간에 비해 약간 증가했음을 알 수 있다.

그룹 멤버에 대한 가입 및 탈퇴 지연 시간은 요구와 응답에 대한 네트웍 지연과 키 관리 지연이다. 네트웍 지연은 그룹 멤버와 키 관리자사이의 네트웍상의 거리에 관련이 있고, 키 관리 지연은 키 생성 및 전송과 같은 계산적인 부하와 그룹 멤버의

수 그리고 그룹 멤버의 변동에 관련이 있다.

5.3.2 데이터 전송 지연 시간

[그림 12]에서는 그룹키 갱신 빈도수에 따른 CBR 소스에 의해 전송되는 멀티 캐스트 데이터 패킷의 전송 지연 시간을 나타낸다. 계층적 트리 구조나 제안한 프로토콜에서의 데이터의 전송은 그룹키 관리 프로토콜과 무관하다. 따라서 그룹키 갱신 빈도수에 영향을 받지 않는다. 반면에 Iolus는 각각의 서브 그룹에서 서로 다른 그룹키를 사용하기 때문에 서브 그룹 멤버들이 멀티캐스트 데이터를 복호화 할 수 있도록 변환 과정이 필요하다. 따라서, 계층적 트리 구조나 제안한 프로토콜보다 매우 큰 데이터 전송 지연 시간을 갖는다.

성능 평가에서처럼, Iolus는 서브 그룹 별로 그룹키를 관리하기 때문에 가입 및 탈퇴의 지연이 본인이 제안한 방식보다 뛰어나다. 그리고 확장성 측면에서는 거의 유사한 성능을 가지고 있다. 하지만 데이터의 전송 측면에서 Iolus는 서브 그룹마다 서로 다른 그룹키를 사용하기 때문에 데이터를 전송할 경우 GSI에서 데이터를 전송하는 서브 그룹의 그룹키를 이용하여 전송 데이터를 복호화하고 데이터를 수신하는 서브 그룹의 그룹키를 이용하여 다시 암호화하여 전송해야 하기 때문에 멀티캐스트 데이터의 전송 측면에서는 가장 좋지 않은 성능을 보인다. 본인이 제안한 방식은 가입과 탈퇴의 측면에서 계층적 트리 구조의 문제점을 해결하고 데이터의 전송 측면에서 Iolus의 문제점을 보완했다. 따라서 가입과 탈퇴의 지연과 데이터 전송 지연을 모두 고려할 때, 제안한 프로토콜이 가장 우수함을 알 수 있다.

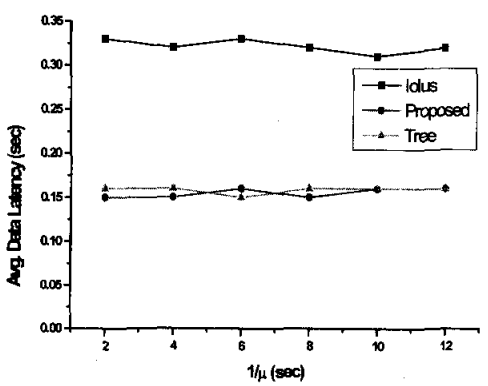

그림 12. 데이터 전송 지연

VI. 결 론

본 논문에서는 은밀한 멀티캐스트를 위한 확장성과 보안성을 고려한 새로운 그룹 키 관리 프로토콜을 제안하였다. 제안된 프로토콜은 분산된 키 관리 구조를 가지고 있기 때문에 하나의 키 서버로 집중되는 트래픽을 분산시킬 수 있으며, 보다 낮은 확장성을 제공한다. 또한, 서브 그룹 별로 그룹키를 갱신하기 때문에 계산적인 부하를 분산시킬 수 있고 네트웍 상으로 그룹키를 전송하지 않기 때문에 보다 낮은 보안성을 제공한다.

성능 평가를 통해서 Iolus, 계층적 트리 구조 그리고 제안한 프로토콜을 비교 분석하였다. 성능 평가 결과는 가입과 탈퇴 지연 시간 측면에서 Iolus가 가장 좋은 성능을 보였고 다음으로 제안한 프로토콜이 대부분의 어플리케이션에서 수용 가능한 만족스러운 성능을 나타냈다. 그리고 데이터 전송 지연 시간 측면에서는 계층적 트리 구조와 함께 가장 좋은 성능을 보였다. 그룹의 멤버수가 증가함에 따라 그룹 가입 및 탈퇴 빈도수가 늘어나기 때문에 확장성 측면에서도 우수하다. 이러한 두 가지 평가 기준과 확장성 측면을 모두 고려해 볼 때, 제안한 프로토콜이 가장 우수함을 알 수 있었다.

제안한 멀티캐스트 그룹키 관리 프로토콜은 보안성을 요구하는 N-to-N 구조를 가진 영상회의나 음성회의 어플리케이션에 적합하다. 이것은 인터넷의 발전에 따라 더 많은 다양성을 요구하는 멀티캐스트 어플리케이션을 충족시킬 수 있으며 나아가 더 좋은 품질의 멀티캐스트 어플리케이션을 구현할 수 있을 것이다.

참 고 문 헌

[1] T. Hardjono, B. Cain, and I. Monga. Intra-Domain Group Key Management Protocol. Internet Draft, draft-ietf-ipsec-intragkm-00.txt, Nobember 1998.

[2] R. Housley, W. Ford, W. Polk, and D. Solo. Internet X. 509 Public Key Infrastructure Certificate and CRL Profile. RFC 2459, January 1999.

[3] H. Harney and C. Muckenhirm, Kua, "Group Key management protocol (GKMP) specification," RFC 2093, IETF, 1996

[4] T. Ballardie, "Scalable multicast key distribution,"

RFC 1949, IETF, 1996
[5] S. Mittra, "The Iolus framework for scalable secure multicasting," in *Proceedings of ACM SIGCOMM'97*, pp. 277-288, ACM, 1997
[6] C.K. Wong, "Secure group communications using key graphs," in Proceedings of ACM SIGCOMM'98, ACM, September, 1998.
[7] B. Schneier. *Applied Cryptography Second Edition: protocols, algorithms, and source code in C.* John Wiley & Sons, Inc., 1996. ISBN 0-471-11709-9.
[8] G.-H Chou and W.-T Chen, "Secure broadcasting using the secure lock," IEEE Trans. Software Eng., vol.15, pp.929-934, 1989
[9] A.Fiat and M.Naor, "Broadcast encryption," in Advances in Cryptology, CRYPTO'93, D.R.Stinson, Ed. Berlin, Germany: Springer Verlag, 1994, vol.773, Lecture Notes in Computer Science, pp.480-491.
[10] D.R.Stinson, "On some methods for unconditionally secure key distribution and broadcast encryption," Designs, Codes Cryptography, vol.12, no.3, pp.215-243, 1997.
[11] C. Wong, M. Gouda and S.S. Lam, "Secure Group Communication Using Key Graphs", *Proceedings of ACM SIGCOMM'98*, 1998.
[12] A. Ballardie, "Scalable multicast key distribution", RFC 1949, 1996.
[13] A. Ballardie, "Core Based Trees multicast routing architecture", RFC 2201, 1997.
[14] M. Doar, "A Better Model for Generating Test Networks", *Proc. of Global Internet*, IEEE, November 1996.
[15] C. Hanle, " Performance Comparision of Reliable Multicast Protocols using the Network Simulator ns-2 ", *Proc. of the Annual Conference on Local Computer Networks (LCN)*, IEEE, October 1998.
[16] UCB/LBNL/VINT Network Simulator, ns version 2, http://wwwmesh.cs.berkeley.edu/ns/.
[17] I. Chang, R. Engel, D. Kandlur, D. Pendarakis, and D. Saha, "A Toolkit for Secure Internet Multicast", Manuscript, 1998.
[18] B. Schneier, "*Applied Cryptography: Protocols, Algorithms, and Source Code in C*" John Wiley & Sons, 2nd edition, December 1995.

이 현 종(Hyeon-Jong Lee) 준회원

1990년 2월 : 광운대학교 전자공학부 졸업
1992년 2월 : 광운대학교 전자통신공학과 석사
1992년 3월~현재 : 현대 제이콤 기술 연구소

<주관심분야> 네트워크 보안, 무선 네트워크 보안, 전자상거래

김 진 철(Jin-Chul Kim) 정회원

1995년 2월 : 광운대학교 전자통신공학과 졸업
1997년 2월 : 광운대학교 전자통신공학과 석사
1996년 12월~현재 : 한전 KDN
2000년 3월~현재 : 광운대학교 전자통신공학과 박사과정

<주관심분야> PKI, 네트워크 보안, 무선네트워크 보안

오 영 환(Young-Hwan Oh) 정회원

한국통신학회 논문지 제28권 제3B호 참조