제19회 한국정보처리학회 춘계학술발표대회 논문집 제10권 제1호 (2003. 5)

멀티캐스트의 효율적 키 분배 및 보안성 향상 구축

라영주*, 전정훈, 김범근, 김도문, 전문석
숭실대학교 컴퓨터학과
e-mail:tolizal@empal.com

Efficent Key distribution and security of Multicast elevation construction

Young-Joo Ra*, Jung-Hun Jun, bum-Gun Kim, Do-Moon Kim, Moon-Seog Jun
Dept of Computing, Soong-Sil University

요 약

대부분의 인터넷 서비스는 일대일 전송방식의 best-effort를 지향하는 유니캐스트(Unicast)가 보편화 되어있다. 하지만, 다자간 통신 서비스는 고려하지 않아 망 자원 이용측면에서 매우 비 효율적이다. 최근, 인터넷방송이나 소프트웨어 분배, 원격 화상회의, 다중사용자 게임, 증권시세 정보서비스 등 다자간 멀티미디어 서비스가 주요 인터넷 사업으로 각광을 받으면서, 멀티캐스트(Multicast) 전송기술의 사용범위가 점차 증가되고 있다. 멀티캐스트는 그룹참가자의 가입과 탈퇴가 빈번한 특징이 있어 키 전달 과정에서 네트워크의 과부하를 초래한다. 본 논문에서는 빈번하게 생성되는 그룹 키의 길이를 축소시켜 메시지의 생성과정을 단축하고, 독립된 그룹간 통신에 사용되는 유니캐스트에 IPSec(Internet Protocol Security Protocol)을 적용시켜 보다 안전하게 구간의 접근제어와 무결성 및 기밀성을 보장하는 SDKD(Secure Dynamic Key Distribution)를 제안한다.

Ⅰ. 서론

1. 개요

기존의 유니캐스트(Unicast) 전송기법은 일대일 방식으로써 하나의 송신자가 하나의 수신자에게 데이터패킷을 보냄으로써 일대다 혹은 다대다의 서비스를 적용하기에는 대역폭의 낭비가 심하다.

최근 인터넷방송이나 소프트웨어 분배, 원격 화상회의, 다중사용자 게임 등의 출현으로 다수의 사용자를 요구하는 서비스의 효율적인 전송을 위해서 제안된 전송기법이 멀티캐스트(Multicast)이다.[1, 2]

멀티캐스트는 그룹관리를 통하여 하나의 그룹에 데이터패킷이 전송된 후 그룹 참가자들에게 데이터 패킷의 복사본이 전송되므로 대역폭의 효율성을 높일 수 있다. 하지만, 그룹관리에 필요한 참가자들의 빈번한 가입과 탈퇴로 인해 네트워크의 오버헤드를 초래한다. 게다가 그룹간 통신에 필요한 키 전달과정에서의 무결성 및 기밀성 문제를 내포하고 있다.

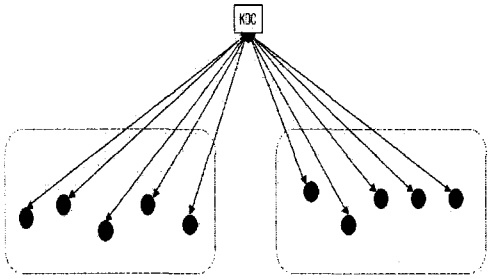
본 논문에서는 이러한 문제점들을 개선하여 세션을 안전하게 보호하기 위해 키 재분배 과정에서의 오버헤드를 줄일 수 있는 방안으로 독립 그룹 내에서 가입과 탈퇴가 원활이 이루어지며 키에 대한 보안성을 향상시키는 효율적인 SDKD (Secure Dynamic Key Distribution)를 제안한다.

2. 관련연구

2.1 KDC(Key Distribution Center) 구조

KDC를 이용한 방식은 모든 그룹 참가자들이 하나의 KDC에서 그룹 키를 받는다. 이 모델은 참가자의 수가 적은 그룹에서는 구현이 쉽고 참가자의 관리가 수월하지만, 모든 참가자가 하나의 KDC에만 의존하므로 참가자의 수가 많아지면 병목현상이 발생하고, 동적인 멀티캐스트 세션연결에 부적절하며,

확장성이 없다는 단점이 있다. 그림1과 같이 2개 그룹에 10명의 참가자에 대해 그룹키 관리를 KDC가 처리해야하는 부담이 있다.[8]

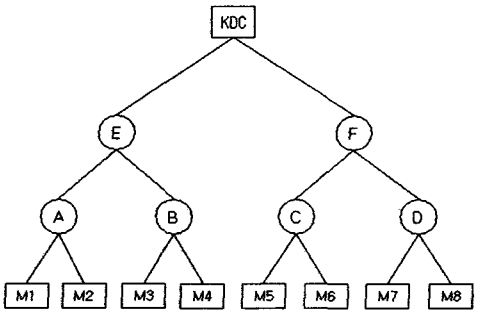


[그림 1. KDC 그룹관리]

2.2 계층적 트리 구조(Hirerarchical Tree Archite cture)

계층적 트리 구조는 트리의 루트(KDC)에서 멀티캐스트 트래픽을 보호하기 위한 그룹키를 관리하고, 참가자들은 각자 다른 자신의 키를 가지고 트리의 leaf노드가 된다. 참가자들은 자신으로부터 루트까지의 모든 키들을 저장해야 한다. 그림 2의 M1은 자신의 키와 A, E, 루트까지의 key를 알고 있어야 한다.

계층적 트리 구조는 트리의 한 부분에서 키 재분배가 일어날 때, 다른 반대부분으로의 영향이 적기 때문에 적은 메시지 트래픽만으로 참가자들의 키 재분배 과정이 가능하며, 확장성이 우수하다. 그림 2에서와 같이 M1~M4 참가자가 가입이나 탈퇴가 발생할 경우 M5~M8에는 영향이 적다. 하지만, 참가자들은 "루트까지의 pass key + 1"개의 키가 필요하고 루트가 공격당할 경우 키가 유출되거나 오류 발생 시 치명적 일 수 있다.

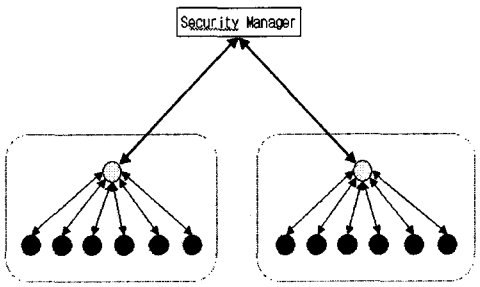


[그림 2. 트리구조의 키관리]

2.3 GKMP(Group Key Management Protocol)

GKMP는 Security manager(KDC)에 집중적이던 키에 대한 액세스제어, 키생성, 키분배의 관리를 각 그룹의 신뢰 할 수 있는 세션 참가자를 "그룹조정자"로 선출하여 KDC의 권한을 위임받아 수행을 한다.[3, 4]

Security manager에 대한 트래픽을 각 그룹별로 분산시켰기 때문에, 키 분배의 효율이 향상된다. 하지만 그림3에서 보듯이, 하나의 그룹에서 참가자들이 많아지면 KDC방식과 마찬가지로 그룹조정자가 많은 참가자의 가입과 탈퇴에 대한 키 생성 및 키 분배에 대한 역할을 해야 하기 때문에 상당한 부하가 발생하게 된다. 또한, 모든 참가자들 중 "그룹조정자"를 선발하는 방법도 문제가 된다.



[그림 3. GKMP의 그룹관리]

Ⅱ. 본 문

1. SDKD(Secure Dynamic Key Distribution)

본 논문에서 제안한 SDKD는 GKMP와 같이 키 매니져가 각 그룹의 키분배를 관리한다. 그림4는 SDKD의 전체적인 흐름를 도식화 한 것이다. 각 그룹은 그룹의 참가자중 한명을 선발하여 그룹 조정자의 역할을 담당하며, 키 매니져와의 통신을 통해 다른 그룹키의 정보를 교환한다.

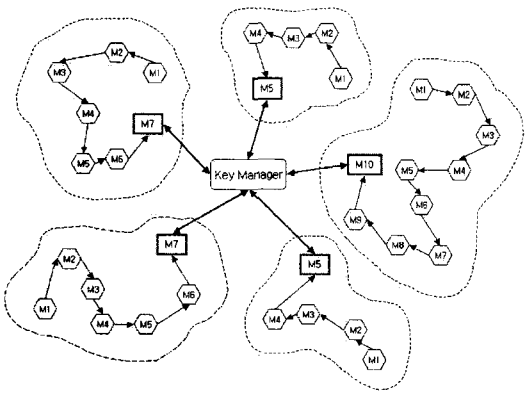
1.1 그룹생성

그룹을 생성하기 위한 그룹 키는 각 그룹 조정자와 그룹 참가자들의 키를 조합하여 생성한다. 단, 키 매니져의 선발문제는 마지막에 참가한 참가자로 선발한다.

각 참가자들이 자신만이 사용하는 mKey (member Key)를 사용하여 그룹 내에서 메시지 암·복호화에 사용되는 GMEK(Group Message Encryption Key)를 생성한다.

GMEK = h(m1_mKey + m2_mKey + ,,, mn_Key)

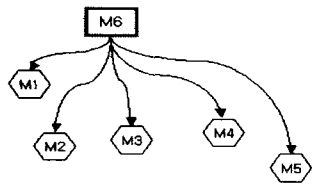
GMEK의 생성은 각 참가자들의 mKey를 이웃한 참가자에게 전달하고 받은 참가자는 자신의 mKey와 연산 후, 다시 이웃한 참가자에게 전달한다. 이와 같은 과정으로 마지막에 참가하여 그룹 조정자로 선발된 참가자에게 전달되면, 연산하여 GMEK가 생성된다.[7]



[그림 4. SDKD 전체흐름도]

1.2 그룹 키 분배

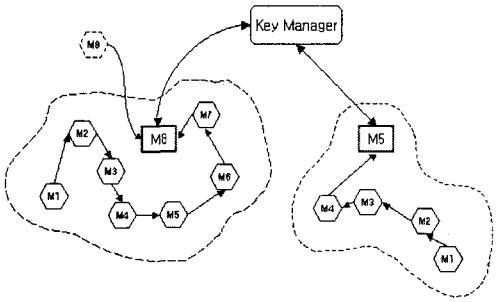
키 매니져는 각 참가자들에게 GMEK를 브로드캐스트 한다. 이때, 생성된 키는 참가자의 수가 많을수록 키의 길이가 커지기 때문에 사용하기에 부적절한 큰 키를 GMEK로 사용한다는 것은 키 저장문제와 메시지의 경우에도 연산과정에서 처리를 하게 되고 키 매니져의 키 분배 시에도 대역폭의 낭비가 될 수 있다. 이러한 문제점들은 해쉬 알고리즘을 사용해 키의 길이를 간략화 시킴으로써 기밀성과 무결성을 제공하게 된다. 그림5는 그룹조정자가 참가자에게 GMEK를 분배하는 과정을 보여준다.



[그림 5. SDKD 키 분배 과정]

1.3 참가자 가입

새로운 참가자가 조인요청을 보내면, 그룹 조정자는 저장하고 있던 참가자들의 mKey를 새로운 참가자에게 넘겨준다. 새로운 그룹조정자는 받은 mKeys를 자신의 mKey와 연산하여 새로운 GMEK를 생성하여 각 참가자들에게 전달한다. 그림6 에서와 같이 이전의 그룹조정자 M7이 갖고있던 GMEK를 M8이 전달받아 자신의 mKey와 연산후 새로운 GMEK가 생성되므로 키 생성시간을 단축할 수 있다.



[그림 6. SDKD 참가자 가입]

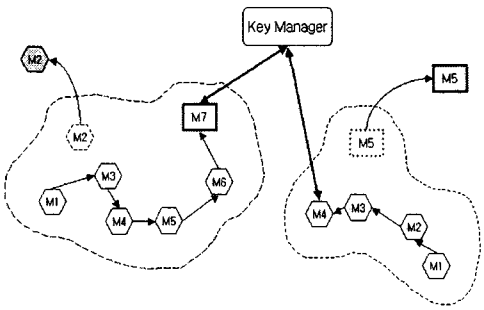
1.4 참가자 탈퇴

참가자의 탈퇴 시 키관리(분배)는 두 가지로 볼 수 있다.

- 참가자가 탈퇴하는 경우에는, 키 생성과정과 같이 이웃하는 참가자에게 다시 자신의 mKey를 전달하여 그룹조정자가 연산 후 키의 재분배가 이루어진다.

- 그룹조정자가 탈퇴하는 경우에는, 참가자들 중 그룹조정자에게 마지막으로 mKey를 전달했던 참가자가 컨트롤러의 역할을 하게 된다.

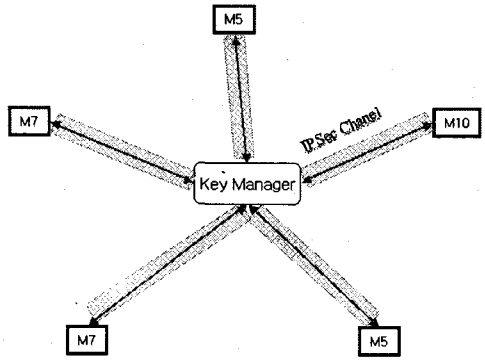
그림7 에서 왼쪽그룹은 참가자 M2가 탈퇴한 후의 키 재생성 과정을 보여주고 있고, 오른쪽 그룹은 그룹조정자였던 M5가 탈퇴한 후에 M4가 그룹조정자의 역할을 하는 과정을 보여준다.



[그림 7. SDKD 참가자 탈퇴]

1.5 그룹 간 통신

각 그룹은 독립적인 GMEK를 갖고 있기 때문에 목적지 그룹의 GMEK를 알지 못하면 통신은 불가능하다. 키 매니져는 각 그룹의 GMEK를 알고있고, 각 그룹조정자에게 그룹별 GMEK를 제공하여 각 그룹간에 연결자 역할을 함으로써 서로 다른 그룹간 통신을 가능하게 한다. 이때, 키 매니져와 그룹조정자 사이의 키 정보 전송은 유니캐스트로 전송되고, IPSec을 사용하여 구간의 접근제어와 무결성 및 기밀성을 보장한다. 그림8은 키 매니져와 각 그룹조정자간에 IPSec을 이용한 안전한 통신을 나타낸다.[5, 6].



[그림 8. 키 매니져와 각 그룹조정자간의 통신]

Ⅲ. 결론

본 논문에서의 KDC(Key Distrbute Center), 계층적 트리 구조(Hirerarchical Tree Architecture), GKMP(Group Key Management Protocol)는 중앙집중형으로써 키 관리(생성 및 분배) 과정에서의 과부하 문제와 KDC나 루트로 키 분배 권한이 집중되어 외부로부터 공격당할 경우 치명적인 보안성 문제를 발생한다. 그 문제점의 대안으로 멀티캐스트(Muticast) 환경에서의 키 관리를 KDC나 루트에 집중시키지 않고 각 그룹에 그룹조정자를 이용하여 그룹별 키 관리를 분산시킴으로써, 외부 공격의 피해를 그룹별로 최소화시키고, 키 생성자(그룹조정자)의 부담을 줄이며, 여러 그룹들간 전송에 IPSec을 적용시켜 서로 다른 그룹간 그룹키를 전달하는 과정에서 접근제어와 무결성 및 기밀성을 보장하여 보안성을 해결하는 효율적인 방안을 제안하였다.

향후 멀티캐스트 전송기법을 이용한 서비스 구현시 참가자 증가에 대한 확장성과 서로 다른 그룹참가자들 간의 통신에서 안전한 서비스 제공이 기대된다.

참고문헌

[1] P.Pessi, "Secure Multicast", Tik-110.501 Seminar on Ntwork Security, 1995.

[2] T. Hardjono and G. Tsudik, "IP Multicast Security: Issues and Directions", Annales de Telecom, July-August 2000, pp 324-340.

[3] H. Harney, C. Muckenhirn "Group Key Management Protocol(GKMP) Specification" RFC2093 July 1997.

[4] H. Harney, C. Muckenhirn "Group Key Management Protocol(GKMP) Architecture" RFC2094 July 1997.

[5] S.Kent, R Atkinson "IP Authentication Header" RFC2402 November 1998.

[6] S.Kent, R Atkinson "IP Encapsulating Security Payload" RFC2406 November 1998.

[7] Michael Steiner, Gene Tsudik "Key Agreement in Dynamic Peer Groups" IEEE Transactions on Parallel and Distributed Systems, August 2000.

[8] D. Wallner, E. Harder and R. Agee, Key Management for Multicast : Issues and Architectures, Internet-Draft, draft-wallner-key-arch-01.txt, September 1998.