안전하고 효율적인 그룹 키 분배에 관한 연구

김정윤*, 최형기**
*성균관대학교 휴대폰학과
**성균관대학교 정보통신공학부
e-mail:steal83@ece.skku.ac.kr
hkchoi@ece.skku.ac.kr

A Study on Secure and Efficient Key Distribution for Group Communication

Jung-Yoon Kim*, Hyoung-Kee Choi**
*Dept. Mobile Systems Engineering, Sungkyunkwan University
**School of Information and Communication Engineering, Sungkyunkwan University

요 약

최근 네트워크 기술의 발전에 의해 VoIP, IP-TV 등 다양한 서비스들이 등장하였다. 이러한 실시간 서비스들은 품질을 보장하면서 통신 내용을 안전하게 보호할 수 있는 보안 메커니즘이 필수적이다. 우리는 VoIP를 이용한 다자간 통화 및 화상 회의, 그리고 IP-TV와 같은 그룹 기반의 서비스들을 안전하고 효율적으로 보호하기 위한 그룹 키 분배 프로토콜을 제안한다. 제안하는 프로토콜은 빠르고 효율적인 연산만으로 구성되었으며, 그룹 내부 및 외부의 공격으로부터 그룹 키를 안전하게 보호한다. 성능평가 및 분석 결과는 제안하는 프로토콜이 최근에 연구된 다른 프로토콜들에 비해 안전하고 효율적임을 증명하였다.

키워드 : 키분배, 그룹키, 그룹통신, 네트워크보안, 보안프로토콜

I. 서론

최근 네트워크 기술이 발전함에 따라, VoIP, IP-TV 등 인터넷을 기반으로 하는 다양한 서비스들이 등장하였다. VoIP는 기존 인터넷 망을 활용함으로써 저렴한 비용의 전화 서비스가 가능하기 때문에, 기존 PSTN 기반의 전화를 대체할 새로운 서비스로 기대되고 있다. IP-TV는 기존의 방송 시스템과 달리, 서비스 제공자와 사용자가 상호 작용함으로써 양방향 서비스를 제공할 수 있다. 이러한 양방향 통신은 사용자에 따라 차별화 된 서비스를 제공한다.

한편, VoIP 및 IP-TV 서비스는 모두 오픈 네트워크인 인터넷을 기반으로 제공되기 때문에, 각종 위협에 노출되어 있다. 예를 들어, VoIP의 경우, 통화 주체들 외의 다른 사용자들이 통화 내용을 도청할 수 없도록 통화 내용을 안전하게 보호해야 한다. 특히, 다자간 통화나 화상 회의의 경우, 여러 사용자들이 통화에 참석할 수 있기 때문에, 권한이 없는 사용자가 통화에 참여하거나 통화 내용을 도청할 수 없도록 접근 제어를 수행해야 한다. 그러나 이러한 접근 제어나 통화 내용에 대한 암호화를 수행하게 되면, 통화 내용의 송수신에 지연이 발생할 수 있으며, 이는 통화의 품질을 떨어뜨리는 원인이 된다. 다자간 통화나 화상 회의의 경우에는 통화 주체가 셋 이상이 될 수 있기 때문에, 보안에 소요되는 오버헤드가 일반 통화에 비해 더욱 클 수 밖에 없다. 따라서, 다자간 통화 및 화상 회의의 경우는 1:1 통화에 비해 통화 품질의 저하를 최소화 시키면서 통화 내용을 안전하게 보호할 수 있는 방법이 더욱 절실히 요구된다.

IP-TV의 경우, 서비스 제공자는 과금 처리가 완료된 합법적인 사용자만이 서비스를 수신할 수 있도록 하는 접근 제어를 필요로 한다. 즉, 합법적인 사용자들만이 공통으로 알고 있는 그룹 키를 이용하여 컨텐츠를 암호화 함으로써 컨텐츠를 보호할 수 있다. 이 경우, 그룹 키의 효율적인 재분배 문

제를 해결해야 한다. 예를 들어, 과금 기간이 만료되거나, 혹은 가입 해지를 함으로써 더 이상 컨텐츠의 수신 권한이 없는 사용자가 발생할 경우, 서비스 제공자는 해당 사용자가 더 이상 컨텐츠를 수신할 수 없도록 기존의 그룹 키를 갱신해야 하며, 갱신된 그룹 키를 합법적인 사용자들에게 다시 전송해야 한다. 이러한 그룹 키 재분배를 안전하게, 그리고 서비스 품질의 저하를 최소화 하면서 수행하기 위한 방안이 연구되어야 한다.

우리는 본 논문을 통해, Pseudo Random Function (PRF)과 XOR을 이용한 안전하고 효율적인 그룹 키 분배 프로토콜을 제안한다. 우리가 제안하는 프로토콜은 PRF의 성질을 이용하여 그룹 내부 및 외부의 공격자로부터 그룹 키를 안전하게 보호할 수 있다. 그리고 PRF의 반복 사용에 의해 발생할 수 있는 전방향 안전성 문제를 XOR의 적용으로 극복하였다. 뿐만 아니라, 제안하는 프로토콜은 PRF, XOR과 같이 매우 빠른 연산만을 이용하여 그룹 키를 분배함으로써, VoIP나 IP-TV의 성능을 극대화할 수 있다.

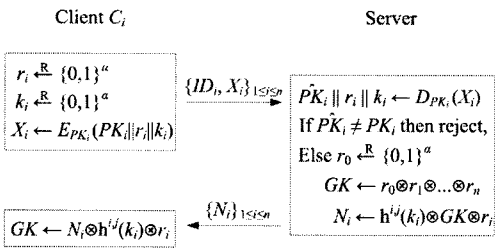
본 논문의 이후 구성은 다음과 같다: 2장에서는 그룹 키 분배에 관련된 연구들을 소개하며, 3장에서는 제안하는 프로토콜에 대해 설명한다. 4장에서는 제안하는 프로토콜의 성능을 평가하고 분석한다. 끝으로, 5장에서는 향후 연구계획을 제시하면서, 결론을 내린다.

II. 관련 연구

단말에게 그룹 키를 분배하기 위한 방식에는, 키 전달 프로토콜과 키 동의 프로토콜이 존재한다. 키 전달 프로토콜은, 키 관리 서버가 그룹 키를 생성 및 전달하는 방식을 의미한다. 키 동의 프로토콜은, 키 관리 서버 없이 단말들이 메시지를 송수신하여 그룹 키를 공유하게 되는 방식을 의미한다. 우리는 기존에 연구된 키 전달 프로토콜과 키 동의 프로토콜을 분석하였고, 그 설명은 다음과 같다.

1. 키 동의 프로토콜

Sherman 등은 머클 트리 (Merkle Tree)를 이용한 그룹 키 분배 프로토콜인 One-way Function Trees (OFT)를 제안하였다 [1]. OFT의 뿌리 노드 (root node)는 그룹 키를 의미하고, 종단 노드 (leaf node)는 단말과 키 관리 서버가 공유하는 비밀 값을 의미한다. 단말은 그룹 키를 획득하기 위해, 먼저 종단 노드에 해당하는 비밀 값을 해쉬 함수로 연산한 값과, 형제 노드 (sibling node)에 해당하는 비밀 값을 해쉬 함수로 연산한 값을 XOR 함으로써, 부모 노드 (parent node)를 갖게 된다. 그리고 이 부모 노드에 대해서도 위와 같은 과정을 거쳐, 궁극적으로 그룹 키인 뿌리 노드가 계산된다. OFT는, 단말의 가입이나 탈퇴가 발생할 때마다, 대칭 키 기반 암호화 알고리즘과 해쉬 함수를 각각 $\log2(n+1)$ 씩 수행해야 한다.



〈그림 1〉 제안하는 그룹 키 분배 프로토콜의 기본적인 동작 과정

Jung은 연산 자원이 제한된 단말들이 효율적으로 그룹 키를 공유하기 위한 키 동의 프로토콜을 제안하였다 [2]. Jung의 프로토콜은 디피-헬만 (Diffie-Hellman) 기반의 그룹 키 동의 방식을 사용하고 있으며, XOR, 해쉬 등 가벼운 연산만으로 구현이 가능하다. 그러나, Lee 등은 Jung의 프로토콜에 보안 취약점이 존재한다는 사실을 증명하였다 [3]. Lee 등에 따르면, Jung의 프로토콜은 내부 공격자에 의한 서비스 거부 공격이 발생할 수 있다.

2. 키 전달 프로토콜

Dondeti 등은 그룹의 계층 구조에 기반하여 그룹 키를 분배하는 Dual-Encryption Protocol (DEP)을 제안하였다 [4]. 이 연구에 의하면, 그룹 키를 효율적으로 갱신하기 위해서는, 하나의 그룹을 다수 개의 서브 그룹으로 분할해야 하며, 각 서브 그룹의 그룹 키를 관리하는 서브 그룹 매니저가 존재해야 한다. 또한, 서브 그룹 매니저가 메시지를 수신할 권한이 없는 경우, 서브 그룹 매니저의 접근을 제한하기 위해 키 서버는 그룹 가입자에게만 알려진 그룹 키를 이용하여 메시지를 암호화 한다. DEP는 서브 그룹 매니저의 접근 제어가 필요한 상황에는 적합하지만, 그렇지 않은 경우에는 불필요한 암호화에 의한 오버헤드가 발생한다.

Sun 등은 Pay-TV에 적합한 새로운 CAS를 제안하였다 [5]. 그들은 그룹 키 전달에 사용되는 모든 값들을 사전에 오프라인으로 전달하는 방식을 사용하였다. 즉, 저자들은 단말마다 저장해야 하는 정보를 늘리는 대신, 전송 오버헤드를 감소시키는 그룹 키 분배 프로토콜을 제안하였다. 시스템에 존재하는 모든 단말 C_i ($1 \le i \le n$)에는 각 단말과 관련된 고유 정보 I_i ($1 \le i \le n$)가 있으며, I_i는 C_i를 제외한 나머지 n-1개의 단말들이 모두 알고 있는 값이다. 만약 단말 C_i가 탈퇴하면, 나머지 n-1개의 단말들은 I_i를 기존 키에 XOR 함으로써 새로운 키를 획득하게 된다. 따라서, 탈퇴한 단말은 갱신된 키를 알 수 없고, 탈퇴한 단말을 제외한 모든 단말은 갱신된 키를 알 수 있다.

III. 제안하는 그룹 키 분배 프로토콜

〈그림 1〉은 우리가 제안하는 프로토콜의 기본적인 동작 과정을 나타낸다. r_i 및 k_i는 단말 C_i ($1 \le i \le n$)가 생성하는 랜덤

값이고, PK_i는 단말 C_i와 서버가 공유하고 있는 비밀 키이다. 이 비밀 키는 사전에 공유하고 있는 값이거나, 혹은 공개키 등을 이용하여 사전에 전달한 값이 될 수 있다. a는 보안 파라미터 (security parameter)이며, 생성하는 랜덤값의 크기(bit)를 나타낸다. n은 현재 그룹에 존재하는 단말의 수를 나타낸다. GK는 그룹 키 (Group Key)를 나타내며, $h^{i,j}(x)$는 단말 C_i가 지금까지 그룹 키를 전송받은 횟수가 j-1번일 때, $h^{i,j-1}(x)$를 입력값으로 하여 PRF를 1회 수행한 값이다. 즉, $h^{i,j}(x)$는 입력값 x에 대해 단말 C_i가 PRF를 총 j번 수행한 결과를 의미한다. 그리고 ⊗은 XOR 연산을 나타낸다.

〈그림 1〉에서 볼 수 있듯이, 단말 C_i는 먼저 랜덤값 r_i 및 k_i, 그리고 서버와 공유하고 있는 비밀 키인 PK_i를 함께 암호화 한다. 이 때 사용되는 암호화 키는 PK_i이기 때문에, 결국 PK_i로 PK_i 자신을 암호화 하는 self-encryption 기법이 이용된다. Self-encryption은 비밀정보의 노출을 최소화 하면서 동시에 사용자 인증을 가능하게 하는 기법으로서, challenge-response 기법에서 사용되는 넌스 (nonce) 혹은 타임스탬프 (timestamp)의 전송이 불필요하기 때문에, 효율적인 인증을 가능하게 한다. 그러나 Self-encryption은 재전송 공격에 취약하다는 단점이 있다. 우리가 제안하는 프로토콜에서는 그룹 키 분배에 사용되는 랜덤값을 self-encryption의 입력값에 포함시킴으로써, 결국 비밀정보의 노출을 최소화 하는 동시에 불필요한 전송값을 최소화 하고 사용자 인증을 가능하게 한다.

단말 C_i가 서버에게 자신의 ID와 함께 PK_i, r_i, k_i를 안전하게 전송하면, 서버는 그것을 복호화 하고, 해당 단말의 PK_i와 전송된 PK_i를 비교함으로써 단말 C_i를 인증할 수 있다. 만약 인증에 성공하면, 서버는 랜덤값 r_0를 생성하여 비밀리에 보관하고, 그룹 키 $GK = r_0 \otimes r_1 \otimes \dots \otimes r_n$를 계산한다. 그리고 서버는 $h^{i,1}(k_i) = h(k_i)$를 수행하고, 그 결과를 저장한다. 만약, 추후에 새로운 단말이 그룹에 가입하게 되면 그룹 키 갱신이 필요하게 되는데, 이 때 그룹 키 갱신을 위해 서버는 단말 C_i로부터 k_i를 받는 대신, 저장해둔 $h^{i,1}(k_i)$을 입력값으로 하여 PRF를 수행한다. 그리고 그 결과인 $h^{i,2}(k_i) = h(h^{i,1}(k_i)) = h(h(k_i))$를 저장한다. 즉, $h^{i,j}(k_i)$는, 그룹 키 갱신이 발생할 때마다 PRF를 1회씩 수행하고 그 결과를 다음 번 그룹 키 갱신의 입력값으로 사용함으로써, 궁극적으로 j번째 그룹 키 갱신이 발생할 때 수행된 PRF의 결과를 의미한다. 따라서 각각의 그룹 키 갱신 과정에서의 PRF 횟수는 1회이다.

$h^{i,j}(k_i)$를 계산한 서버는 $N_i = h^{i,j}(k_i) \otimes GK \otimes r_i$를 계산하고, N_i를 단말 C_i에게 전달한다. 이를 수신한 단말 C_i는 $GK = N_i \otimes h^{i,j}(k_i) \otimes r_i$를 계산함으로써 그룹 키 GK를 획득할 수 있다. 이 때, 단말은 이전 그룹 키 갱신 과정에서 계산되었던 $h^{i,j-1}(k_i)$를 저장하고 있다가, 그 값을 입력값으로 하여 PRF를 1회 수행함으로써 $h^{i,j}(k_i)$를 획득할 수 있다. 따라서 단말이 수행하는 PRF의 횟수는 1회이며, 저장하고 있는 값은 $h^{i,j-1}(k_i)$ 뿐이다. 키 갱신이 완료되면, 단말은 $h^{i,j-1}(k_i)$를 삭제하고 $h^{i,j}(k_i)$를 저장한다.

한편, 새로운 단말 C_{n+1}이 그룹에 가입할 경우, 서버는 〈식 1〉 및 〈식 2〉와 같은 절차를 통해 그룹 키 GK와 공개 값 N_i를 각각 GK′과 N_i'으로 갱신하고, 단말 C_i에게 갱신된 N_i'을 전송한다 ($1 \le i \le n+1$).

$$GK' = GK \otimes r_0 \otimes r_0' \otimes r_{n+1} \quad \text{〈식 1〉}$$

$$N_i' = h^{i,j+1}(k_i) \otimes GK' \otimes r_i \quad \text{〈식 2〉}$$

r_0'는 서버가 생성한 비밀 랜덤값이다. $h^{i,j+1}(k_i)$의 경우, 이전 그룹 키 분배 시 계산하고 저장해두었던 $h^{i,j}(k_i)$를 입력값으로 하여 PRF를 1회 수행하면 되기 때문에, 결과적으로 PRF 1회로 획득할 수 있는 값이다.

만약, 기존에 가입했던 단말 C_v가 그룹으로부터 탈퇴할 경우, 서버는 〈식 3〉 및 〈식 4〉와 같은 절차를 통해 그룹 키 GK와 공개 값 N_i를 각각 GK″과 N_i''으로 갱신하고, 단말 C_i에게 갱신된 N_i''을 전송한다 ($1 \le i \le n$, $i \ne v$).

$$GK'' = GK \otimes r_0 \otimes r_0'' \otimes r_v \quad \text{〈식 3〉}$$

$$N_i'' = h^{i,j+1}(k_i) \otimes GK'' \otimes r_i \quad \text{〈식 4〉}$$

r_0''는 서버가 생성한 비밀 랜덤값이다. $h^{i,j+1}(k_i)$의 경우, 이전 그룹 키 분배 시 계산하고 저장해두었던 $h^{i,j}(k_i)$를 입력값으로 하여 PRF를 1회 수행하면 되기 때문에, 결과적으로 PRF 1회로 획득할 수 있는 값이다.

IV. 성능평가 및 분석

제안하는 그룹 키 분배 프로토콜은 PRF 및 XOR만을 사용하여 매우 빠른 연산이 가능하며, 특히 단말 측에서는 그룹 내 가입자의 수와 관계 없이 PRF 1회와 XOR 2회만으로 그룹 키 분배가 가능하다. 뿐만 아니라, 이 연산들 중 PRF 1회와 XOR 1회는 그룹 키 분배와 상관없이 미리 연산이 가능(pre-computable)하다.

〈표 1〉 제안하는 프로토콜과 다른 그룹 키 분배 프로토콜의 성능 비교

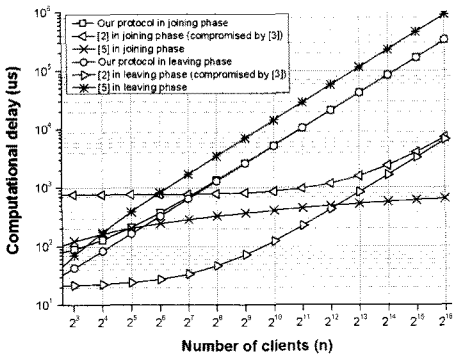
			연산 오버헤드	통신 오버헤드	저장 오버헤드
제안하는 프로토콜	가입 시	서버	D+(n+1)H+(2n+5)X	n	3n+4
		가입한 단말	E+H+2X		3
		나머지 단말	H+2X		3
	탈퇴 시	서버	(n-1)H+(2n+1)X	n	3n-2
		나머지 단말	H+2X		3
		탈퇴한 단말	No operations		1
[2]	가입 시	가입한 단말	2M+3H+nX	2n	3
		나머지 단말	3H+nX		3
	탈퇴 시	나머지 단말	3H+nX	2n	3
		탈퇴한 단말	No operations		0
[5]	가입 시	서버	$Log_2(n)H+Log_2(n)E$	$Log_2(n)$	n+2
		가입한 단말	$Log_2(n)D$		$Log_2(n)+2$
		나머지 단말	No operations		$Log_2(n)+2$
	탈퇴 시	서버	$Log_2(n)H+X$	1	n+2
		나머지 단말	$(2^{\log_2(n)+1}-2)\log_2(n)-2)H$		$Log_2(n)+2$
		탈퇴한 단말	No operations		$Log_2(n)+1$

〈표 1〉은 제안하는 프로토콜과 다른 그룹 키 분배 프로토콜의 연산 횟수, 송신 메시지 개수, 저장 메시지 개수를 비교한다. 제안하는 프로토콜에서 미리 계산이 가능한 연산들도 연산 횟수에 모두 포함시켰다. 〈표 1〉에서 E와 D는 각각 대칭 키 기반 암호화, 복호화를 나타내며, H는 PRF를 나타낸다. 그리고 X는 XOR 연산을 나타내며, M은 모듈러 지수 연산을 나타낸다.

[2]의 경우 그룹 키 분배에 서버가 필요 없다는 장점이 있고, 연산 성능이 매우 뛰어나다는 장점이 있지만, 관련 연구에서도 설명했듯이 [3]에 의해 취약점이 증명되었다.

[5]에서 제안하는 프로토콜의 단말 측 연산 오버헤드는 우리가 제안하는 프로토콜보다 크다는 것을 알 수 있고, 단말의 저장 오버헤드도 O(1)이 아니기 때문에 확장성에 문제가 발생할 수 있다. 그러나 [5]에서 제안하는 프로토콜의 서버 측 연산 오버헤드 및 통신 오버헤드는 머클 트리를 이용하기 때문에 매우 낮다.

비록 제안하는 프로토콜은 그룹 가입자의 수에 비례하여 서버 측 연산 오버헤드 및 통신 오버헤드, 서버 측 저장 오버헤드가 증가하지만, 각각의 연산 자체는 매우 빠른 PRF 및 XOR로 구성되어 있다. 그리고 단말 측에서는 가입자 수에 상관 없이 고정된 횟수의 연산 및 고정된 크기의 저장 공간만을 필요로 한다. 한편, 제안하는 프로토콜의 메시지 개수는 비록 O(n)이라는 시간복잡도를 갖지만, 각 메시지의 크기는 작다. 만약 [5]와 같이, 제안하는 프로토콜에 트리 기반의 키 분배 기법을 적용한다면, 시간복잡도를 O(log2(n))으로 줄일 수 있고, 이 경우 다른 프로토콜보다 연산 및 통신 효율이 향상될 수 있다. 우리는 제안하는 프로토콜에 트리 기법을 적용하는 방안을 연구 중이다.



〈그림 2〉 제안하는 프로토콜과 다른 그룹 키 분배 프로토콜의 연산 시간

〈그림 2〉는 가입자의 수 n에 따른 제안하는 프로토콜과 다른 그룹 키 분배 프로토콜의 연산 오버헤드를 나타낸다. 〈그림 2〉에 의하면 [2]에서 제안하는 프로토콜이 성능 면에서 가장 우수함을 알 수 있다. 그러나 위에서도 언급했듯이 [2]는 [3]에 의해 취약점이 증명되었다. [5]에서 제안하는 프로토콜의 경우, 단말의 가입 시 성능은 우수한 반면, 단말의 탈퇴 시 오버헤드는 매우 크다. 따라서 그룹의 가입 및 탈퇴가 빈번하게 발생하는 환경에서 안전성 및 성능을 모두 고려하였을 때, 우리가 제안하는 프로토콜이 가장 우수함을 알 수 있다.

V. 결론

우리는 본 논문을 통해 안전하고 효율적인 그룹 키 분배 프로토콜을 제안하였다. 제안하는 프로토콜은 PRF를 이용하여 내부 및 외부 공격자에 의한 키의 노출을 차단하였으며, 반복적인 PRF의 사용 시 발생할 수 있는 전방향 안전성의 문제도 PRF와 함께 XOR을 사용함으로써 극복하였다. 그리고 그룹 키 분배를 위해 PRF 및 XOR과 같이 수행 속도가 매우 빠른 연산만을 사용하며, 단말 측에서는 그룹 가입자의 수에 상관 없이 상수 (constant) 횟수의 연산만을 수행하면 되므로 매우 빠르고 효율적인 그룹 키의 분배가 가능하다. 또, 가벼운 연산만을 사용함으로써 단말의 배터리 소모량도 최소화 하였고, 단말에서 저장해야 하는 정보의 크기도 매우 작기 때문에 실제 구현에 소요되는 비용이 매우 적다.

향후 연구과제는, 다른 그룹 키 분배 프로토콜과 같이 머클 트리 기반의 키 분배 기법을 제안하는 프로토콜에 적용하는 것이며, 현재 해당 연구를 진행하고 있다. 또한, 제안하는 프로토콜의 안전성을 더욱 상세히 분석하고, 더욱 다양한 그룹 키 분배 프로토콜과의 성능을 비교하는 것이 또 다른 향후 연구과제가 될 것이다.

참고문헌

[1] A. T. Sherman and D. A. McGrew, "Key Establishment in Large Dynamic Groups Using One-Way Function Trees," IEEE Trans. Software Engineering, vol. 29, no. 5, pp. 444-458, May. 2003.

[2] B. E. Jung, "An Efficient Group Key Agreement Protocol," IEEE Commun. Letters, vol. 10, no. 2, pp. 106-107, Feb. 2006.

[3] S. M. Lee and D. H. Lee, "Analysis of an Efficient Group Key Agreement Protocol," IEEE Commun. Letters, vol. 10, no. 8, pp. 638-639, Aug. 2006.

[4] L. R. Dondeti, S. Mukherjee, and A. Samal, "Scalable Secure One-to-many Group Communication using Dual Encryption," Computer Commun., vol. 23, no. 17, pp. 1681-1701, Nov. 2000.

[5] H. M. Sun, C. M. Chen, and C. Z. Shieh, "Flexible-Pay-Per-Channel: A New Model for Content Access Control in Pay-TV Broadcasting Systems," IEEE Trans. Multimedia, vol. 10, no. 6, pp. 1109-1120, Oct. 2008.