Ad-hoc환경에서의 2-라운드 비대칭 키공유 기법

이원희, 구재형, 이동훈*

*고려대학교, 정보보호기술연구센터

Asymmetric 2-rounds Key Agreement for Ad-hoc networks

Won Hui Lee, Jae Hyung Koo, Dong Hoon Lee*

*Center for Information Security Technologies(CIST) Korea Univ.

요 약

Diffie-Hellman(DH) 키공유 기법[1]이 제안된 후 많은 종류의 키공유 기법들이 연구되었으며, 특히 특정한 그룹 내의 구성원들이 안전한 통신을 할 수 있게 하기 위한 그룹 기반의 키공유 기법들에 대한 많은 방식들이 제안되었다. 이러한 방식들은 PKI 기반구조 등을 사용하는 여러 응용들에서 쉽게 사용될 수 있지만, Ad-hoc통신망이라는 특정한 기반구조를 사용할 수 없는 환경에서는 적용하기가 쉽지 않다. [2]에서는 Ad-hoc 환경에서의 그룹(conference) 기반의 키공유 기법을 제안하였지만, (n+2)-라운드의 통신량이 요구되며, 그룹 구성원들의 위치정보를 알고 있어야 한다. 본 논문에서는 ad-hoc 환경에서의 효율적이며, 그룹 구성원들의 위치정보를 알 필요 없는 2-라운드 키공유 기법을 제안한다.

I. 서론

신뢰할 수 없는 채널 상에서 안전한 통신을 제공하려면 서로 공유된 키를 사용하여 암호통신을 하여야 한다. 안전한 키공유를 위해 DH 키공유 기법[1]이 제안되었으며, 이후 상호간에 전달되는 데이터에 인증을 추가한 기법 등 다양한 DH 키공유 기법들이 제안되었다. 그러나 ad-hoc통신망의 다음과 같은 특성 때문에 PKI와 같은 기반구조를 사용하는 기법들은 ad-hoc통신망 환경에 적용하기 힘들다.

Ad-hoc통신망은 AP(Access Point)와 같은 기반구조 없이 각 무선호스트들 사이에 데이터 전송이 가능한 망이다. 각 무선호스트들은 각 호스트의 무선통달거리 내에 존재하는 호스트들과 직접 데이터 전송이 가능하며, 라우터 기능을 가지는 무선 호스트들이 존재할 경우 원격 호스트(multi-hop host)와도 통신이 가능하다. 또한 무선 호스트들의 잦은 위치변화로 망구조가 유동적이며, 고정된 기반구조를 사용할 수 없는 전장이나 천재지변으로 기반구조를 사용할 수 없는 환경에서 사용가능하다.[4][5]

이와 같은 ad-hoc통신망의 특징 때문에 ad-hoc 통신망에서는 인증된 키공유를 위해 패스워드 기반의 키공유 기법을 사용할 수 있다.

패스워드는 일반적으로 사람이 기억할 수 있는 정보이다. 이러한 정보는 매우 제한된 집합 내에서 선택된 것이므로 낮은 엔트로피를 갖게 된다. 따라서 패스워드로 암호화된 데이터를 갖고 있는 공격자의 경우 사전공격(dictionary attack)을 통해 올바른 패스워드를 찾을 수 있다. 따라서 낮은 엔트로피를 가지는 패스워드로부터 높은 엔트로피를 가지는 세션키를 생성할 수 있는 기법이 필요하다.

패스워드 기반의 키공유의 경우 다음과 같은 성질을 만족해야한다.

Perfect forward Secrecy : 패스워드를 알고 있는 사용자만이 해당 세션키를 만들 수 있어야 하며, 사용된 패스워드가 유출되더라도 이전의 세션에서 사용된 세션키를 알 수 없어야 한다.

Contributory key agreement : 세션키가 모든

참여자의 참여값으로 생성되는 기법을 contributory 기법이라 부른다. Contributory 기법에서는 특정 사용자 또는 사용자들에 의해 세션키가 제한된 키공간에서 선택될 수 없어야 한다.

Tolerance to disruption attempts : 전송되는 데이터에 대한 수정이나 삭제는 할 수 없으나 삽입은 가능한 공격자에 대해서 안전해야 한다.

[2]에서는 이러한 성질을 만족하는 패스워드 기반의 키공유기법을 제안하였다. 그러나 그룹구성원들이 그룹세션키를 만들기 위해 순차적으로 각자의 비밀값을 지수승 함으로써 라운드 수가 그룹구성원의 수에 따라 선형적으로 증가하며, 다른 그룹구성원들의 정보를 알고 있어야 한다. 본 논문에서는 이러한 단점을 보완한 패스워드 기반의 2-라운드 비대칭 그룹키공유 기법을 제안하고자 한다.

제안된 기법은 다음과 같은 제한된 환경을 가정한다.

회의장에서 ad-hoc통신망을 구성할 수 있는 무선 호스트를 사용하여 소규모 그룹이 회의를 하고자 한다. 올바른 그룹 구성원들은 회의장에 들어오기 전에 안전한 방법으로 그룹 간에 공유된 패스워드를 알 수 있다. 물론 Ad-hoc통신망만이 구성될 수 있기 때문에 상호인증을 위해서 인증서나 신뢰할 수 있는 KDC(Key Distribution Center)를 사용할 수 없다. 그룹 내에는 회의 주최 측의 무선 호스트가 그룹리더로 존재하며, 이 호스트는 좀더 강력한 계산능력을 가질 수 있다.

II. 본론

1. 패스워드 기반의 그룹키 공유

Asokan과 Ginzboorg는 [2]에서 Bellovin과 Merrit가 [3]에서 제안한 EKE(Encrypted Key Exchange)를 다중 사용자가 기여(contributory multi-party)하여 그룹 세션키를 구성할 수 있도록 변형시킨 기법을 제안하였다.

M_1부터 M_n까지 n명의 그룹원이 존재하고 M_n은 그룹의 리더이며, 이들은 사전에 동일한 패스워드를 공유하고 있다. S_i는 세션키를 만들기 위한 M_i의 참여값인 난수다. M_n은 비대칭키 암호화 방식의 키쌍(E,D)을 가지고 있다.

(1) $M_n \rightarrow ALL$: M_n, $P(E)$

(2) $M_i \rightarrow M_n$: M_i, $P(E(R_i,S_i))$, i=1,...,n-1

(3) $M_n \rightarrow M_i$: $R_i(\{S_j, j=1,...,n\})$, i=1,...,n-1

(4) $M_i \rightarrow M_n$: M_i, $K(S_i,H(S_1,S_2,...,S_n))$

위의 과정을 통해서 세션키 $K = f(S_1,S_2,...,S_n)$가 도출된다, 여기서 f()은 n개의 입력값을 갖는 일방향 함수이다.

그룹구성원 간에 공유된 패스워드인 P로 암호화된 공개키 E로 S_i와 R_i가 전달되므로 패스워드를 알지 못하는 사용자의 경우 단계(3)에서 올바른 S_j, j=1,...,n을 알 수 없게 되고, 결과적으로 패스워드를 알고 있는 사람만이 올바른 세션키를 만들 수 있게 된다. 단계(4)는 키 확인의 역할을 한다.

2. 패스워드 기반의 인증된 Diffie-Hellman 그룹키 교환

[2]에서는 [6]에서 제안한 다중 사용자 환경에서의 인증된 DH 그룹키 분배 기법을 사용하여 다중 사용자 환경에서 그룹키를 생성하는 기법을 제안하였다.

N명의 그룹구성원 $M_1,M_2,...,M_n$이 사전에 공유된 패스워드 P를 가지고 있을 때, 각 구성원 M_i는 난수 S_i를 생성한다. [2]에서 제안하고 있는 기법은 위와 같은 상황에서 P를 알고 있는 구성원들만이 그룹 세션키 $K = g^{S1S2...Sn}$를 공유할 수 있도록 구성되어있다. 이 기법은 세 부분으로 구성되는데, 첫 번째 부분(단계(1),(2))에서 $M_1,M_2,...,M_{n-1}$은 n-1번의 과정 후에 중간키인 $\pi = g^{S1S2...Sn-1}$ 을 생성한다. 이 때 M_n은 π에 자신의 비밀값인 S_i를 지수승 함으로써 그룹세션키 K를 구할 수 있다. 두 번째 부분(단계(3),(4))에서 각 구성원 M_i는 P를 키로 하여 단계(2)에서 받은 값에 자신의 은닉요소인 $\hat{S}_i/S_i$을 지수승한 값 $C_i = \pi^{Si/Si}$를 그룹리더인 M_n에게 보내고, M_n은 이 값에 자신의 비밀값인 S_n을 지수승하여 각 M_i에게 보낸다. M_i들은 M_n으로부터 받은 값에서 자신의 은닉요소를 제거함으로써 그룹세션키인 K를 구할 수 있다. 마지막 세 번째 부분(단계(5))은 키 확인 과정이다.

(1) $M_i \rightarrow M_{i+1}$: $g^{S1S2...Si}$, i=1,...,n-2

(2) $M_{n-1} \rightarrow ALL$: $\pi = g^{S1S2...Sn-1}$

(3) $M_i \rightarrow M_n$: $P(C_i)$, i=1,...,n-1

(4) $M_n \rightarrow M_i$: $(C_i)^{Sn}$, i=1,...,n-1

(5) $M_i \rightarrow ALL$: M_i, $K(M_i, H(M_1,M_2,...,M_n))$

Ad-hoc환경에서 위 기법은 두 가지 단점을 가지고 있다.

첫 번째 단점은 단계(1)에서 $g^{S1S2...Si}$의 계산이

노드들 간에 순차적인 전달에 의해서 이루어지기 때문에 M_i는 다음 노드인 M_{i+1}에 관한 정보(주소, 위치)를 알고 있어야 한다. 즉, 각 노드들은 어떠한 노드들이 그룹의 구성원으로 참여하고 있으며, 데이터가 전달되는 순서와 다음 노드의 위치정보를 저장하고 있어야 한다. 이것은 노드들의 위치가 고정적이지 않고, 데이터의 전달이 다른 호스트의 도움으로 이루어지는 ad-hoc환경에서는 상당한 오버헤드로 작용하게 된다.

두 번째로 단계(1)에서 한노드에서 다음 노드로 순차적으로 데이터를 전달하므로 세션키를 공유하기까지 총 (n+2)-라운드가 요구되는 단점이 있다.

3. Ad-hoc환경에서의 2-라운드 비대칭 키공유 기법

본 논문에서 제안하는 기법은 각 구성원들이 단계(1)에서 자신의 비밀값인 S_i를 랜덤하게 선택하여 생성된 값 g^{Si}을 그룹구성원들 간에 사전 공유된 값인 패스워드 P로 암호화 하여 그룹리더에게 전송한다. 이후 그룹리더는 각 구성원들로부터 받은 g^{Si}에 자신의 비밀값인 S_n을 지수승한 값을 곱하므로 $K' = g^{S1Sn} \cdot g^{S2Sn} \cdot \ldots \cdot g^{Sn-1Sn}$를 구하고, 이 값을 해쉬시킴으로써 그룹세션키 $K = H(K')$를 구할 수 있다. 여기서 H()는 일방향 해쉬함수이다. K'를 알게 된 그룹리더 M_n은 자신의 공개값 g^{Sn}을 P로 암호화 시킨 값과 각 노드들과 공유될 g^{SiSn}으로 K'를 나눈 값 C_i를 각 구성원 M_i에게 전달한다. 이 값을 받은 각 구성원들은 g^{SiSn}을 구해 C_i에 곱한 값을 해쉬시키는 것을 통해 그룹세션키 $K = H(Ci \cdot g^{SiSn})$를 구할 수 있게 된다.

(1) $M_i \rightarrow M_n$: $P(g^{Si})$, i=1,...,n-2

(2) $M_n \rightarrow M_i$: $P(g^{Sn})$, $C_i = K'/g^{SiSn}$, i=1,...,n-1

(3) $M_i \rightarrow ALL$: $M_i, K(M_i)$

단계(1)에서 순차적으로 비밀값을 지수승을 하는 [2]와는 달리 각 구성원들은 그룹리더에게 자신의 공개값을 바로 전달하기 때문에, 각 그룹구성원들은 그룹리더의 정보만을 가지고 있으면 된다. 또한 모든 그룹세션키를 생성하기 위한 모든 단계가 그룹리더와 그룹구성원 사이에서만 이루어지기 때문에, 그룹 구성원의 수와 관계없이 라운드수가 일정하게 된다. 또한 그룹리더를 제외한 나머지 구성원들에 대한 계산량은 [2]에서 보다 적게 요구된다.

4. 안전성 분석

1) Perfect forward secrecy

만일 그룹 간에 공유된 패스워드 P가 유출될 경우, 공격자는 P로 암호화된 메시지인 $g^{S1},\ldots,g^{Sn}$은 알 수 있지만 Diffie-Hellman 문제에 의해서 g^{SiSn}를 구할 수 없으므로 해당 세션의 그룹키는 알 수 없게 된다.

2) Contributory key agreement

세션키가 $K = H(g^{S1Sn} \cdot g^{S2Sn} \cdot \ldots \cdot g^{Sn-1Sn})$와 같이 모든 구성원들의 참여값이 사용되고, 계산된 값이 해쉬되므로 contributory를 만족한다.

3) Tolerance to disruption attempts

단계(1)에서 공격자가 임의의 값 g^r을 M_n에게 보냈다고 가정하자. 그러면 M_n은 g^r을 P로 복호화하고 이 값에 자신의 비밀값을 지수승하여 그룹세션키 $K = g^{S1Sn} \cdot g^{S2Sn} \cdot \ldots \cdot g^{Sn-1Sn} g^{r'Sn}$를 구성할 수 있다. 그리고 단계(2)에서 자신에게 데이터를 보낸 모든 노드에게 C_i를 전달한다. 올바른 그룹구성원들은 $g^{r'Sn}$에 상관없이 그룹세션키 K를 구할 수 있지만, 공격자의 경우는 올바른 $g^{r'Sn}$를 구할 수 없으므로 그룹세션키 K를 구할 수 없게 된다.

5. 효율성 분석

표 1은 위에서 언급한 각 기법에서 그룹세션키가 공유되기까지의 라운드수와 각 노드별 계산량을 비교한 것이다. 이 표에서 볼 수 있는 것처럼 본 논문에서 제안하고 있는 기법의 경우 [2]와는 달리 구성원의 수에 관계없이 라운드횟수가 2로 고정될 수 있으며, 계산량의 경우 [2]에서의 일반 구성원들 M_i은 2번의 지수승과 3번의 곱셈연산, 1번의 암호화 과정이 필요하지만, 본 논문에서 제안하고 있는 기법에서의 일반 구성원들 M_i은 1번의 지수승과 1번의 곱셈연산, 1번의 암호화 과정과 1번의 해쉬 과정이 요구된다. 그룹리더인 M_n의 경우 [2]에서는 n-1번의 지수승과 n-1번의 암호화 과정이 요구되며, 본 논문의 기법의 경우 n-1번의 지수승과 n번의 암호화 과정, 2(n-1)번의 곱셈연산, 1번의 해쉬 과정이 요구된다. 일반 그룹구성원의 경우 계산량이 기존 기법에 비해 적으며, 그룹리더의 경우 비슷한 계산량을 요구한다.

III. 결론

본 논문에서는 [2]에서 그룹구성원들이 그룹 세션키를 만들기 위해 순차적으로 각자의 비밀값을 지수승 함으로써 발생하는 라운드 수를 줄이기 위해, 그룹리더에게 그룹구성원 각자의 공개값을 보내도록 하여, 그룹 세션키를 $g^{S1Sn} \cdot g^{S2Sn} \cdot \ldots \cdot g^{Sn-1Sn}$와 같은 형태로 구성함으로써 그룹리더의 위치정

	기법1	기법2
라운드수	(n+2) 라운드	2 라운드
노드별 계산량	M_i : - 1 지수승 (단계1) - 1 지수승, 1 나눗셈, 1 암호화 (단계3) - 2 곱셈 (단계4) Mn : - n-1 복호화 (단계3) - n-1 지수승 (단계4)	M_i : - 1 지수승, 1 암호화, 1 곱셈, 1 해쉬 (단계3) M_n : - n-1 복호화, 1 암호화, n-1 지수승, n-1 곱셈, 1 해쉬 (단계2) - n-1 나눗셈 (단계3)

표 1: 기법 간 성능비교

※ 기법1 - Diffie-Hellman 그룹키 공유

※ 기법2 - 2-라운드 비대칭 키공유

보만을 가지고 2-라운드에 그룹세션키를 구성할 수 있는 기법을 제안하였다. 또한 제안된 기법은 그룹리더를 제외한 일반 그룹구성원들의 계산량이 [2]보다 한번의 지수승 만큼 적게 요구된다. 본 논문에서 제안된 기법을 사용하여 ad-hoc환경에서 좀더 효율적인 그룹키공유를 제공할 수 있다.

참고문헌

[1] W. Diffie and M. Hellman, "New directions in cryptography," IEEE Trans. Inform. Theory, vol. IT-22, pp 644-654, Nov 1976

[2] N.Asokan and Philp Ginzboorg. "Key Agreement in Ad-hoc Networks," Computer Communications, vol 23:1627-1637, 2000

[3] Steven M. Bellovin and Michael Merrit. "Encrypted key exchange : Password-based protocols secure against dictionary attacks," In Proceedings of the IEEE Symposium on Research in Security and Privacy, May 1992

[4] Carlo Kopp, "Ad Hoc Networking," Published in 'System', June 1999, pp 33-40

[5] 김동완, "이동 Ad Hoc망 기술 개요," http://kmh.ync.ac.kr/Network2/mobile/2000/itr03-20000300.htm

[6] Michael Steiner, Gene Tsudik and Michael Waidner. "Private communication," Unpublished work described in slides of presentation made at the Third ACM CCS conference, March 1996.