情報保護學會論文誌
第14券 第1號, 2004. 2

효율적인 패스워드 기반 그룹 키 교환 프로토콜*

황 정 연,[a)†] 최 규 영,[a)] 이 동 훈,[a)‡] 백 종 명[b)]
고려대학교,[a)] 유비넷(주)[b)]

Efficient Password-based Group Key Exchange Protocol

Jung Yeon Hwang,[a)†] Kyu Young Choi,[a)] Dong Hoon Lee,[a)‡] Jong Myung Baik[b)]
Korea University,[a)] Ubinet[b)]

요 약

패스워드 기반 인증된 그룹 키 교환 프로토콜은 (안전하지 않다고 여겨지는) 공개 네트워크 상에서 인간이 기억 가능한 패스워드를 공유한 참가자 그룹에게 멀티 캐스트 데이터 무결성과 비밀성을 위해 사용될 세션 키를 제공해 준다. 본 논문에서는 패스워드 기반 인증된 그룹 키 교환 프로토콜을 제시하고 결정적 DH 문제와 계산적 DH 문제의 어려움에 근거하여 랜덤 오라클 모델과 이상적 암호화 모델에서 안전성을 증명한다. 이 프로토콜은 상수 번의 통신 라운드가 요구되며 키 공유를 위해 단지 사용자마다 O(1)번의 모듈라 지수 승을 요구한다. 따라서 매우 효율적이며 참가자 집합의 크기에 독립적이다. 또한 이 프로토콜은 전방향 안전성을 제공한다.

ABSTRACT

Password-based authenticated group key exchange protocols provide a group of user, communicating over a public (insecure) channel and holding a common human-memorable password, with a session key to be used to construct secure multicast sessions for data integrity and confidentiality. In this paper, we present a password-based authenticated group key exchange protocol and prove the security in the random oracle model and the ideal cipher model under the intractability of the decisional Diffie-Hellman(DH) problem and computational DH problem. The protocol is scalable, i.e. constant round and with O(1) exponentiations per user, and provides forward secrecy.

Keyword: *password-based group key*

Ⅰ. 서 론

1.1 배 경

상업적 회의나 모임들에서 또는 인터넷을 통해 기업은 인증된 다수의 이용자에게 E-비지니스(business)를 제공한다. 군사 작전이나 응급구조와 같은 임무위주의 응용분야, 소수 그룹의 사용자들이 협력해야하는 개별적인 네트워킹 등은 기저 자원이 충분하지 않는 환경에서 종종 운영되어진다. 이런 경우들에서는 종종 단순한(비용 효과적인) 인증 메커니즘이 요구되어지는 반면에 안전한 멀티캐스트 세션이 요구된다. 효율적인 비밀 키 관리의 요구나 기저 자원의 부족으로 인해서 패스워드를 이용하여 인증된 그룹 키 교환 프로토콜들은 유용하게 위의 구현에 이용될 수 있다. 특히 공개키 기반 구조(Public Key Infrastructure, PKI)의 사용이 불가능하거나 이용되는 데 어려움이 많은 다양한 환경에서 활용될 수

접수일: 2003년 10월 7일; 채택일: 2004년 1월 5일
* 본 연구는 (주)유비넷 연구과제 지원으로 수행하였습니다.
† 주저자, videmot@cist.korea.ac.kr
‡ 교신저자, donghlee@korea.ac.kr

있다.

패스워드 기반 인증된 그룹 키 교환 프로토콜은 안전하지 않다고 여겨지는 공개 네트워크 상에서 단지 복잡도가 매우 낮은(인간이 기억할 수 있을 정도의) 패스워드만을 사용하여, 프로토콜에 관여된 참가자의 신원에 대한 어떤 확신을 제공하고 동시에 참가자들 사이에 안전한 멀티캐스트 세션을 위한 암호학적으로 강한 공유키를 생성하는 프로토콜이다.

인간이 기억 가능한 패스워드란 기억하기 쉬운 반면 상대적으로 엔트로피가 낮아 가능한 스트링의 공간의 크기는 작고 전수 조사 공격에 취약하게 된다. 따라서 패스워드 기반 프로토콜의 설계에서는 패스워드의 낮은 안전성의 수준을 암호학적으로 충분히 강한 수준으로 높이는 확장 구조가 요구되며 이는 그 설계의 어려움의 요인으로 작용된다.

1.2 본 논문의 결과

본 논문에서는 매우 효율적인 패스워드 기반 인증된 그룹 키 교환 프로토콜을 제안한다. 제시된 프로토콜은 이수미 외 저자들에 의해 제안된 부분 그룹의 키 공유 프로토콜[1]에 기반 한다. 그리고 결정적 DH 문제와 계산적 DH 문제의 어려움에 근거하여 랜덤 오라클 모델과 이상적 암호화 모델에서 안전성을 증명한다. 이 프로토콜은 2번의 통신 라운드를 요구하며 키 공유를 위해 단지 사용자마다 O(1)번의 모듈라 지수 승이 요구된다. 이런 의미에서 프로토콜은 참가자 집합의 크기에 독립적이다. 또한 제시된 프로토콜은 장기간 사용되는 패스워드의 노출에도 이전의 세션 키가 노출되지 않는다는 의미에서의 전방향 안전성을 제공한다.

1.3 관련된 연구들

Bellovin과 Merritt[2]에 의해 제기된 패스워드 기반 2자간 세션 키 교환문제 이후로 PKI의 사용 없이 위의 문제를 해결하려는 연구가 많이 진행되어 왔다. 특히, Bellare 외 저자들은 이 문제에 대한 형식적인 모델을 제시하였고 이상적 암호화(ideal cipher) 모델에서 안전성이 증명된 프로토콜들을 제안하였다.[3] Boyko 외 저자들은 다자간모의실험 기법을 이용하여 랜덤 오라클 모델에서 안전성이 증명된 프로토콜들을 제안하였다.[4] 최근에 Katz 외 저자들[5]과 Goldreich 외 저자들[6]이 제안한 프로토콜들은 표준모델에서 안전성이 증명되었다.

2자간 패스워드 기반 키 교환 프로토콜들에 대한 활발한 연구와는 대조적으로 다자간의 경우는 많은 연구가 이루어지지 못하였다. 최근 Bresson 외 저자들은 최초로 패스워드 기반 인증된 그룹 Diffie-Hellman 키 교환 스킴에 대한 형식적인 연구를 제시하였다.[7] 이 스킴은 [3]의 2자간 형태를 다자간 참여로 확장시킨 형태이고 [8]의 스킴에 기반 하였다. 하지만 통신라운드가 프로토콜 참가자들에 대해 선형적으로 증가하고 키를 계산하는 비용이 많이 요구되는 단점을 가졌다.

1.4 본 논문의 구성

본 논문의 구성은 다음과 같다. Ⅱ장에서는 패스워드 기반 인증된 키 교환 프로토콜의 형식적 모델을 살펴보고 Ⅲ장에서는 암호학적 가정들을 살펴본다. Ⅳ장에서는 패스워드 기반 그룹 키 공유 프로토콜들을 제안하고 안전성을 증명한다. Ⅴ장에서는 다른 프로토콜과 효율성을 비교하고 결론을 맺는다.

Ⅱ. 모 델

본 장에서는 패스워드 기반 그룹 키 교환 프로토콜의 형식적인 모델을 살펴본다. 다음 내용 중 더 세부적인 부분은 [7]를 참조한다.

2.1 프로토콜 참가자들과 인터페이스

그룹 Diffie-Hellman 키 교환 프로토콜에 참가하는 참가자들의 집합을 U로 고정하자. 참가자 $U_i \in U$는 프로토콜의 서로 구별되고 병렬적인 실행에 관여되는 *오라클*이라고 불리워지는 많은 인스턴스들을 갖는다. 여기서는 참가자 U_i 의 t-번째 인스턴스를 Π_i^t로 나타내기로 하자.

패스워드 기반 그룹 키 교환 프로토콜은 패스워드 생성 및 키 교환 알고리즘들로 구성된다. 패스워드 생성 알고리즘은 보안 상수를 입력으로 받아 크기 N인 패스워드들의 집합에서 고르게 분포된 패스워드를 출력한다. 이 알고리즘을 통해 참가자들은 사전에 낮은 엔트로피를 갖는 패스워드를 공유한다. 키 교환 알고리즘은 공통의 패스워드를 소유한 그룹 내의 참가자 오라클에게 세션 키를 제공하는 다자간 상호적 프로토콜이다.

2.2 쿼리들(Queries)

공격자는 다양한 쿼리를 통해 프로토콜 참가자들과 상호 작용한다. 다음은 이러한 쿼리들의 형태와 각 쿼리들이 갖는 능력을 설명한다.

- 전송 Send(Π_i^t, m) : 이 쿼리는 공격자가 사용자 오라클 Π_i^t에게 메시지 m을 보내는 상황을 모델링 한다. 그 사용자 오라클 Π_i^t는 공격자에 의해서 조작된 통신 메시지를 받고 프로토콜에 의해 기술된 방식으로 반응한다.
- 실행 Execute(U) : 이 쿼리는 공격자가 단순한 도청에 의해 정직한 실행에 대한 접근을 얻는 수동적인 공격을 모델링 한다. 이 경우 공격자는 참가자들 사이의 정직한 프로토콜 실행에 의해 얻어지는 전달 메시지들을 돌려받는다.
- 유출 Reveal(Π_i^t) : 이 쿼리는 세션 키의 손실이 다른 세션 키의 노출을 일으켜서는 안 된다는 개념을 모델링 한다. 만일 사용자 오라클이 세션 키를 가지고 승인했었다(accepted)면 공격자에게 그 세션 키 sk가 되돌려진다.
- 손상 Corrupt(U) : 이 쿼리는 상대적으로 장기간 사용되는 키인 패스워드의 노출로부터 다른 세션들을 보호하고자 하는 전방향 안전성(forward secrecy)의 개념을 모델링 한다. 공격자는 패스워드를 돌려받는다.
- 테스트 Test(Π_i^t) : 이 쿼리는 스킴의 의미론적(semantic) 안전성에 대해 공격자의 성공 능력을 측정하기 위해 이용된다. (이것은 실제의 공격 능력에 대응되지는 않는다.) 사용자 오라클 Π_i^t가 세션 키 sk를 가지고 승인했었을 때, 다음을 실행한다. 동전 가 던져진 후 만일 b=0이면 sk가 b=1이면 랜덤한 키가 되돌려진다. 이 쿼리는 한번만 이용될 수 있고 $Fresh$(아래 내용 참조)한 사용자 오라클 Π_i^t에 대해서만 가능하다.

2.3 안전성 개념

2.3.1 파트너링(Partnering)

제시된 프로토콜에서 모든 메시지는 프로토콜에 참가하고 있는 모든 참가자 그룹에게 브로드 캐스트 된다. 여기서 브로드 캐스트의 의미는 단지 개별적인 링크를 통하여 참가자들이 그룹의 모든 구성원에게 같은 메시지를 보낼 수 있음을 나타낸다. sid_i^t를 오라클 Π_i^t에 의해 보내고 받은 모든 메시지들을 라운드와 참가자 아이디의 순서로 연접하여 나열된 문자열 형태라고 하자. 그리고 pid_i^t를 Π_i^t가 통신하기 원했던 그룹 내의 참가자들의 아이디 집합이라고 하자. 만일 $pid_i^t = pid_j^s$이고 $sid_i^t = sid_j^s$라면 사용자 오라클 Π_i^t와 Π_j^s는 *파트너 되어졌다*라고 불리워진다.

2.3.2 Freshness와 전방향 안전성

만일 공격자가 테스트-쿼리를 던지기 전에 손상-쿼리를 던지지 않았고, 오라클 Π_i^t가 세션 키 sk($\neq$ NULL)를 계산하였고 Π_i^t은 물론 이의 파트너들에게 유출-쿼리를 던지지 않았다면 오라클 Π_i^t은 $Fresh$하다고 정의된다.

본 논문에서는 전방향 안전성에 대해서도 고려한다. 현실적인 환경에서, 공격자는 실제로 패스워드 입력 장면의 목격, '트로이 목마'등의 삽입, 또는 해킹 등으로부터 사용자의 패스워드를 알아 낼 수 있다. 이런 현실적인 안전성의 개념은 공격자의 손상-쿼리를 통해 모델화된다. 전방향 안전성이 요구되지 않는 경우라면 위의 Freshness의 정의에서 손상-쿼리에 대한 조건은 필요하지 않다.

2.3.3 AKE 안전성

키 비밀성에 관한 안전성의 개념은 공격자의 존재하에서 프로토콜을 실행하는 환경에서 발생한다. 다음과 같은 모의게임 **게임**^{ake}(A,P)를 통해 고려해 보자.

게임은 패스워드 생성 알고리즘, 공격자 (알고리즘) A, 모든 오라클 Π_i^t 에게 확률분포에 관련된 동전을 제공함으로써 초기화된다. 그리고 다음을 진행된다.

(1) 패스워드 생성 알고리즘을 이용하여 패스워드를 생성하고 참가자들에게 분배한다.
(2) 모든 오라클 Π_i^t을 초기화한다.
(3) 공격자 A를 초기화하고 오라클 Π_i^t에 대한 접근과 쿼리들을 허용한다. 그리고 질의된 쿼리에 대해 적절한 응답을 돌려준다.
(4) 게임이 끝나는 시점에서, A는 테스트-쿼리에 관련된 비트 b의 추측 값 b'을 출력한다.

만일 알고리즘 A가 Fresh한 참가자 U에 단일한

테스트-쿼리를 질의하고 **게임**^{ake}(A,P)에서 사용된 비트 b를 올바르게 추측한다면 A는 그 게임에서 승리한다. 이 사건을 Succ로 나타내기로 하자. AKE 이점은

$$Adv_P^{ake}(A) = |2\Pr[Succ]-1|$$

으로 나타낸다. 여기서 확률공간은 공격자와 모든 오라클들의 랜덤한 동전에 대해 취해진다. 또한 A가 프로토콜에 대한 공격으로 패스워드에 대한 공격을 시도하여 성공할 사건을 $Succ_{pw}$라 하고 그 확률을 $\Pr[Succ_{pw}]$로 나타내기로 하자.

Ⅲ. 정의와 암호학적 가정들

여기서는 본 논문에서 제안할 프로토콜의 안전성에 관련된 암호학적 개념과 가정들을 간단히 살펴본다.

3.1 랜덤 오라클(Random Oracle) 모델과 이상적 암호화(Ideal Cipher) 모델

랜덤 오라클 (또는 이상적 해쉬) 모델에서 암호학적 해쉬 함수는 안전성 분석을 위해 공개 랜덤 함수로 간주된다. 이 모델에서 어떤 확률 공간으로부터 랜덤 함수를 선택하는 것은 유한한 문자열의 집합을 치역으로 갖는 랜덤 함수를 선택하는 것을 의미한다.

이상적 암호화 모델에서 어떤 확률 공간으로부터 랜덤 함수를 선택하는 것은 완전한 방식으로 문자열들을 암호화하는 방법을 주어진 프로토콜에 제공해 준다. 즉, 주어진 키 값 K에 대해 선택된 함수 E_K를 랜덤한 1-1 함수로 간주한다.

3.2 결정적(Decisional) Diffie-Hellman (DDH) 문제

DDH 파라미터 생성 알고리즘 IG_{DDH}은 보안 상수 1^k을 입력 값으로 받아 위수 q인 곱셈 연산 군 $\vec{g}$를 출력한다. DDH 문제는 두 가지 형태, (g, g^a, g^b, g^{ab})과 (g, g^a, g^b, g^c)을 구분하는 문제이다. IG_{DDH}에 대한 알고리즘 A의 이점은 다음과 같이 정의 된다:

$$|\Pr[A(G,g,g^a,g^b,g^{ab})=1| \; G \leftarrow IG_{DDH}(1^k);\ g \leftarrow G; a,b \leftarrow Z_q\,] -$$
$$\Pr[A(G,g,g^a,g^b,g^c)=1| \; G \leftarrow IG_{DDH}(1^k);\ g \leftarrow G; a,b,c \leftarrow Z_q\,]|.$$

만일 어떤 확률적 다항식 시간 알고리즘 A도 DDH 문제를 해결하는데 주목할만한(non-negligible) 이점을 얻지 못한다면 IG_{CDH}는 DDH 가정을 만족한다고 불리워진다.

3.3 계산적(Computational) Diffie-Hellman(CDH) 문제

CDH 파라미터 생성 알고리즘 IG_{CDH}은 보안 상수 1^k을 입력 값으로 받아 위수 q인 덧셈 연산 군 $\vec{g}$를 출력한다. CDH 문제는 주어진 (G, g, g^a, g^b)에 대해 g^{ab}을 계산하는 문제이다. IG_{CDH}에 대한 알고리즘 A의 이점은 다음과 같이 정의 된다:

$$|\Pr[g^{ab} \leftarrow A(G,g,g^a,g^b)| \; G \leftarrow IG_{CDH}(1^k);\ g \leftarrow G; a,b \leftarrow Z_q\,]|.$$

만일 어떤 확률적 다항식 시간 알고리즘 A도 CDH 문제를 해결하는데 주목할만한(non-negligible) 이점을 얻지 못한다면 IG_{CDH}는 CDH 가정을 만족한다고 한다.

Ⅳ. 패스워드 기반 그룹 키 교환 프로토콜

본 장에서는 효율적인 패스워드 기반 그룹 키 교환 프로토콜을 제안하고 안전성을 증명한다. 제시된 프로토콜은 모든 참가자가 균등하게 키 생성에 참여하여 그룹 키를 생성하는 대칭형의 키 교환 프로토콜이다. 다음은 이 후 설명에서 사용할 기호 표현에 대한 정의이다.

- U_i : i 번째 사용자.
- G : 위수가 q인 곱셈 연산군.
- g : 곱셈 연산군 G의 생성자.
- pw : 프로토콜 참여자들이 소유한 패스워드.
- $E_{pw}(m)$: 패스워드(pw)를 사용하여 메시지 (m)를 암호화.

- $h : \{0,1\}^* \rightarrow \{0,1\}^{\ell}$ 인 해쉬 함수.
- r_i : i 번째 사용자가 선택한 랜덤한 값.
- γ: 프로토콜 실행 중 전달되는 메시지의 집합.
- K : 세션 그룹키.

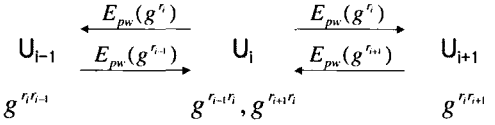
4.1 패스워드 기반 그룹 키 교환 프로토콜

모든 참가자들은 순번이 순환이 되도록 정해져 있으며, 즉 사용자 인덱스는 모듈라 n(=사용자의 수)의 규칙을 따르며, U_i ($1 \le i \le n$)는 그룹의 i번째 프로토콜 사용자를 나타낸다. 다음에서는 크기 N인 패스워드 집합 $Pass$에서 고르게 선택된 pw가 모든 참가자들 사이에 공유되었다고 가정한다.

프로토콜에서는 ℓ비트 크기의 q 값을 위수로 갖는 덧셈 군 G, 해쉬 함수 $h: \{0,1\}^* \rightarrow \{0,1\}^{\ell}$ 그리고 블록 암호 함수들이 이용된다. 블록 암호 함수들은 다음과 같이 키가 부여된 순환(keyed permutation)들의 집합이다: $E_k : G \rightarrow G \mid k \in Pass$. 복호화 함수는 D_k로 나타내기로 한다.

〔프로토콜 PW-GKE〕

〈라운드 1〉 각 사용자 U_i는 r_i값을 임의로 선택하고 g^{r_i}을 계산한다. 패스워드를 이용하여 암호화하고 이 값 $E_{pw}(g^{r_i})$을 브로드 캐스트 한다.



〈라운드 2〉 각 사용자 U_i는 각각 U_{i-1}와 U_{i+1}로부터 수신된 값 $E_{pw}(g^{r_{i-1}})$와 $E_{pw}(g^{r_{i+1}})$을 자신의 비밀 패스워드로 복호화 한다. 그리고 $g^{r_i r_{i-1}}$와 $g^{r_i r_{i+1}}$을 계산하고 $z_i = h(g^{r_i r_{i-1}}) \oplus h(g^{r_i r_{i+1}})$를 계산한 후 z_i값을 브로드 캐스트 한다.

〈키 계산〉 각 사용자 U_i는 모든 사용자로부터 수신된 전달 메시지들을 이용하여 다음과 같이 계산한다:

$$h(g^{r_{i+1}r_{i+2}}) = z_{i+1} \oplus h(g^{r_i r_{i+1}}),$$
$$h(g^{r_{i+2}r_{i+3}}) = z_{i+2} \oplus h(g^{r_{i+1}r_{i+2}}),$$
$$\vdots$$
$$h(g^{r_{i-2}r_{i-1}}) = z_{i-2} \oplus h(g^{r_{i-3}r_{i-2}}).$$

U_i는 $h(g^{r_1 r_2})$, $h(g^{r_2 r_3})$,..., $h(g^{r_n r_1})$ 값들을 얻은 후, 그룹 키 K를 다음과 같이 생성한다:

$$K = h(h(g^{r_1 r_2}) \| h(g^{r_2 r_3}) \| \cdots \| h(g^{r_n r_1})).$$

4.2 안전성 증명

우리는 랜덤 오라클 모델과 이상적 암호화 모델을 사용하여 능동적인 공격자에 대하여 제안한 프로토콜 PW-GKE의 안전성을 증명한다. 증명의 아이디어는 다음과 같다. 공격자는 프로토콜 PW-GKE에 대해 패스워드로부터 이점을 얻거나 얻지 못하는 형태의 2가지로 공격을 할 수 있다. 우리는 이 경우들에 대해서 각각 DDH 문제 또는 CDH 문제 해결의 성공 확률로 공격의 성공 확률이 적절히 제한되어짐을 보인다.

PW-GKE의 안전성을 증명하기 위해 먼저 PW-GKE의 실행 중 패스워드 인증 부분이 빠진 프로토콜의 안전성을 먼저 증명하자. 편의상 이 프로토콜을 L-GKE로 표시하자. 즉, L-GKE의 1-라운드에서 사용자는 g^{r_i}을 브로드 캐스트 하고 이웃하는 참가자와 DH 키 교환을 한 후, 2-라운드에서 $z_i = h(g^{r_i r_{i-1}}) \oplus h(g^{r_i r_{i+1}})$를 브로드 캐스트 한 후 PW-GKE와 동일하게 키 값을 계산한다.

〔정리1〕 프로토콜 L-GKE는 DDH 가정에 기반하여 전방향 안전성을 제공하는 안전한 그룹키 교환 프로토콜이다. 다시 말해서:

$$Adv_{L-GKE}^{GKA-fs}(t, q_{ex}) \le 2|U| Adv_G^{DDH}(t).$$

(증명) 먼저 수동적인 안전성을 위해 전달메세지에 대해 다음을 관찰해 보자. 1라운드에서 주어진 전달 메시지 $\{g^{r_i} \mid 1 \le i \le n\}$ 들로부터는 유용한 정보를 얻을 수 없다고 가정할 수 있다. 또한 2라운드에서 주어지는 $\{z_i \mid 1 \le i \le n\}$ 값들에 대해서도 역시 대응되는 키와 정보이론적인 관점에서 독립적임을 다음과 같이 알 수 있다. 다음과 같이 주어진

$$z_i = h(g^{r_{i-1}r_i}) \oplus h(g^{r_i r_{i+1}}),$$

$$z_{i+1} = h(g^{r_i r_{i+1}}) \oplus h(g^{r_{i+1} r_{i+2}}),$$
$$\vdots$$
$$z_{i-1} = h(g^{r_{i-2} r_{i-1}}) \oplus h(g^{r_{i-1} r_i})$$

에 대하여, $X_i = h(g^{r_i r_{i+1}})$이라 두면,

$$z_i = X_{i-1} \oplus X_i,$$
$$z_i = X_{i+1} \oplus X_{i+2},$$
$$\vdots$$
$$z_i = X_{i-2} \oplus X_{i-1}$$

이 된다. 이 경우 주어진 식을 만족하는 해 $(X_1, \ldots, X_i, \ldots, X_n)$의 형태는 다음과 같이 나타낼 수 있다:

$$X_1 = z_1 \oplus z_2 \oplus \cdots \oplus z_{n-1} \oplus X_n,$$
$$X_2 = z_2 \oplus \cdots \oplus z_{n-1} \oplus X_n,$$
$$\vdots$$
$$X_{n-1} = z_{n-1} \oplus X_n,$$
$$X_n.$$

따라서 주어진 독립변수 X_n 값이 취하는 집합의 크기인 2^{ℓ}만큼 해가 존재하게 된다.

그러므로 개별적인 메시지 정보로부터 얻는 공격자의 이점은 무시할만하다고 가정할 수 있다. 다음은 g^{r_i}와 $z_i = X_{i-1} \oplus X_i$에 대한 연계된 정보에 대한 프로토콜 L-GKE의 안전성을 표준적인 하이브리드(hybrid) 논의에 의해 증명한다. L-GKE의 경우 장기적인 키 사용이 없으므로 손상-쿼리는 무시할 수 있다. 여기서는 공격자 A가 단지 한번의 실행-쿼리를 생성한다고 가정한다. n을 그룹 내의 참가자의 수라 하고 $\varepsilon(t) = Adv_G^{DDH}(t)$라 하자. 다음에서 일련의 변경된 프로토콜들의 메시지들의 분포들 $Fake_i$ $(0 \le i \le n)$을 고려한다. i=0인 경우는 실제(Real) 프로토콜 PW-GKE에 대응된다.

Real=

$$\left\{\begin{array}{l} t_1, t_2, \ldots, t_n \leftarrow G;\ g^{t_1}, g^{t_2}, \ldots, g^{t_n}; \\ z_1 = h(g^{t_n t_1}) \oplus h(g^{t_1 t_2}), \ldots, \\ z_n = h(g^{t_{n-1} t_n}) \oplus h(g^{t_n t_1}); \\ \Im = (g^{t_1}, g^{t_2}, \ldots, g^{t_n}, z_1, z_2, \ldots, z_n); \\ K = h(h(g^{t_1 t_2}) \| h(g^{t_2 t_3}) \| \cdots \| h(g^{t_n t_1})) : (\Im, K) \end{array}\right\}$$

계속해서 변조(Fake) 전달 메시지 $\Im$의 분포와 이에 대응하는 그룹 키 K는 다음과 같다.

Fake_1 =

$$\left\{\begin{array}{l} r_{1,2}, t_1, t_2, \ldots, t_n \leftarrow G;\ g^{t_1}, g^{t_2}, \ldots, g^{t_n}; \\ z_1 = h(g^{t_n t_1}) \oplus h(g^{r_{1,2}}), z_2 = h(g^{r_{1,2}}) \oplus h(g^{t_2 t_3}), \ldots, \\ z_n = h(g^{t_{n-1} t_n}) \oplus h(g^{t_n t_1}); \\ \Im = (g^{t_1}, g^{t_2}, \ldots, g^{t_n}, z_1, z_2, \ldots, z_n); \\ K = h(h(g^{r_{1,2}}) \| h(g^{t_2 t_3}) \| \cdots \| h(g^{t_n t_1})) : (\Im, K) \end{array}\right\}$$

이런 일련의 과정을 거쳐 최종적으로 다음과 같은 변조 전달 메시지의 분포를 얻을 수 있다.

Fake_n =

$$\left\{\begin{array}{l} r_{1,2}, r_{2,3}, \ldots, r_{n,1}, t_1, t_2, \ldots, t_n \leftarrow G;\ g^{t_1}, g^{t_2}, \ldots, g^{t_n}; \\ z_1 = h(g^{t_n t_1}) \oplus h(g^{r_{1,2}}), z_2 = h(g^{r_{1,2}}) \oplus h(g^{r_{2,3}}), \ldots, \\ z_n = h(g^{r_{n-1,n}}) \oplus h(g^{r_{n,1}}); \\ \Im = (g^{t_1}, g^{t_2}, \ldots, g^{t_n}, z_1, z_2, \ldots, z_n); \\ K = h(h(g^{r_{1,2}}) \| h(g^{r_{2,3}}) \| \cdots \| h(g^{r_{n,1}})) : (\Im, K) \end{array}\right\}$$

[주장1] *Real*과 $Fake_1$을 구별하는 임의의 PPT 알고리즘 A의 능력은 DDH 문제의 어려움에 의해 제한된다:

$$\left| \Pr[\Im \leftarrow Real; K \leftarrow Real: A(\Im, K) = 1] - \Pr[\Im \leftarrow Fake_1; K \leftarrow Fake_1: A(\Im, K) = 1] \right| \le \varepsilon(t).$$

(증명) 주어진 알고리즘 A를 이용하여 DDH 문제를 푸는 알고리즘 B를 다음과 같이 구성할 수 있다. DDH 문제 (g, g^x, g^y, g^z)가 B의 입력 값으로 주어졌다고 하자. B는 이 값을 이용하여 다음을 계산한다.

$$t_1, t_2, \ldots, t_n \leftarrow G;\ set\ g^{t_1} = g^x, g^{t_2} = g^y, g^{t_3}, g^{t_4}, \ldots, g^{t_n};$$
$$z_1 = h(g^{t_n t_1}) \oplus h(g^z), z_2 = h(g^z) \oplus h(g^{t_2 t_3}), \ldots,$$
$$z_n = h(g^{t_{n-1} t_n}) \oplus h(g^{t_n t_1});$$
$$\Im = (g^x, g^y, g^{t_3}, \ldots, g^{t_n}, z_1, z_2, \ldots, z_n);$$
$$K = h(h(g^z) \| h(g^{t_2 t_3}) \| \cdots \| h(g^{t_n t_1})) : (\Im, K)$$

B는 위에서 계산되어 나온 결과 값 $\Im, K$를 A의

입력 값으로 준다. 만일 A가 주어진 입력 값을 받아 ξ의 확률로 K가 *Real*에서 생성된 것인지 아니면 $Fake_1$생성된 것인지 구별하여 결과 값을 출력한다면 B는 적어도 ξ 보다 큰 확률로 DDH 문제를 해결할 수 있게 된다. 다시 말해서 $\varepsilon(t)=Adv_G^{DDH}(t)$라 하면 결국 A는 기껏해야 $\varepsilon(t)$의 성공확률을 가지게 된다.

주장 1과 비슷하게 임의의 A에 대하여 우리는 아래와 같은 식이 만족함을 쉽게 알 수 있다.

$$\begin{aligned}&|\Pr[T\leftarrow Fake_1; K\leftarrow Fake_1; A(T,K)=1]\\&-\Pr[T\leftarrow Fake_2; K\leftarrow Fake_2; A(T,K)=1]|\leq\varepsilon(t)\\&\vdots\\&|\Pr[T\leftarrow Fake_{n-1}; K\leftarrow Fake_{n-1}; A(T,K)=1]\\&-\Pr[T\leftarrow Fake_n; K\leftarrow Fake_n; A(T,K)=1]|\leq\varepsilon(t)\end{aligned}$$

위 $Fake_n$의 분포에서 얻은 K값은 T와는 독립적으로 생성되기 때문에 다음과 같은 식을 만족한다:

$$\begin{aligned}&|\Pr[T\leftarrow Fake_n; K\leftarrow Fake_n; A(T,K)=1]\\&=|\Pr[T\leftarrow Fake_n; K\leftarrow Random; A(T,K)=1]|\end{aligned}$$

결국 우리는 아래와 같은 식을 얻을 수 있다.

$$\begin{aligned}&|\Pr[T\leftarrow Real; K\leftarrow Real; A(T,K)=1]-\\&\Pr[T\leftarrow Real; K\leftarrow Random; A(T,K)=1]|\leq 2n\varepsilon(t)\end{aligned}$$

그리고 $n<|U|$ 이므로

$$Adv_{B-GKA}^{GKA-fs}(t,1)=2|U|\varepsilon(t).$$

〔정리2〕 프로토콜 PW-GKE는 DDH와 CDH 가정에 기반하여 전방향 안전성을 제공하는 안전한 패스워드 기반 그룹 키 교환 프로토콜이다. 다시 말해서:

$$\begin{aligned}&Adv_{PW-GKE}^{PGKE-fs}(t,q_{ex},q_s)\leq\\&q_T\cdot Adv_{L-GKE}^{GKE-fs}(t,1)+\frac{n(q_h-1)}{2}Adv_G^{CDH}(t)+\frac{q_s}{N}\end{aligned}$$

그리고

$$\begin{aligned}&Adv_{PW-GKE}^{PGKE-fs}(t,q_{ex},q_s)\leq\\&2q_Tn\cdot Adv_G^{DDH}(t)+\frac{n(q_h-1)}{2}Adv_G^{CDH}(t)+\frac{q_s}{N}.\end{aligned}$$

여기서 q_s, q_{ex}, q_h는 각각 전송-쿼리, 실행-쿼리, 해쉬-쿼리의 횟수를 나타내고 $q_T=q_{ex}+q_s$, N는 패스워드 공간의 크기 그리고 n은 프로토콜 참가자의 수이다.

(증명) B를 PW-GKE를 공격하는 능동적인 공격자라 하자. B는 프로토콜의 인증부분인 패스워드를 공격하여 얻는 이익이 있으며 또는 전달 메시지의 변조 없이 프로토콜을 공격하여 얻는 이익이 있다.

먼저 우리는 공격자가 패스워드를 공격하여 패스워드를 알게 됨으로써 프로토콜을 공격한다고 가정하자. 이때 B가 얻는 이익은 기껏해야 $Succ_{pw}(t)$정도의 성공확률을 가지게 된다.

그 다음으로 우리는 공격자 B가 전달 메시지의 변조 없이 프로토콜을 공격한다고 가정하자. 우리는 B를 이용하여 오직 한번의 실행-쿼리를 만들어 L-GKE를 공격하는 수동적인 공격자 A를 구성할 수 있음을 보인다. 초기에 A는 손상-쿼리를 통해 패스워드를 얻게 된다. B에 의해 만들어진 실행-쿼리와 전송-쿼리의 합을 $q_T=q_{ex}+q_s$라 하고 $\alpha\in[1,q_T]$는 B가 테스트-쿼리를 요청하기 위한 전송/실행-쿼리를 A가 추측하기 위해 A에 의해서 선택된 값이라 하자. A는 B의 공격환경을 시뮬레이션해 주며 다음과 같은 쿼리에 대해 적절한 응답을 되돌려 준다. 그리고 일관성을 유지하기 위해서 암/복호화-,전송-, 해쉬-쿼리에 대해 목록을 만들어 유지한다.

암호화-쿼리. B가 암호화-쿼리(p,M)를 요청하는 경우 A는 암호화 목록인 E_{list}에 (p,M,C)가 존재하면 C를 B에게 전송하고, 존재하지 않으면 길이가 $|M|$인 랜덤한 암호문 C를 전송하고 E_{list}에 (p,M,C)를 저장한다.

복호화-쿼리. B가 복호화-쿼리(pw',C)를 요청하는 경우 A는 암호화 목록인 E_{list}에 (pw',M,C)가 존재하면 M를 B에게 전송하고, 존재하지 않으면 길이가 $|C|$인 랜덤한 평문 M을 전송하고 E_{list}에 (pw',M,C)를 저장한다.

실행-쿼리. 만약 실행-쿼리가 B의 α번째 전송/실행-쿼리가 아니면 A는 알고 있는 패스워드를 이용하여 PW-GKE의 전달 메시지를 생성하여 B에게 보낸다. 만약 실행-쿼리가 α번째 전송/실행-쿼리이면 A는 같은 쿼리를 실행 오라클에게 전송하여 받은

L-GKE에 대한 전달 메시지를 패스워드를 이용하여 PW-GKE에 대한 전달 메시지로 변경하여 B에게 전송한다.

전송-쿼리. $U_U^s = U \mid U_1 \mid \cdots \mid U_n$이라 하자. 만약 전송-쿼리(Send($\prod_U^s$, *))가 B의 α번째 전송/실행-쿼리가 아니면 A는 전송-쿼리가 포함된 형태를 L_{list}목록(형태가 (U_U^s, c)인)에서 찾아본 후 다음과 같이 대응한다.

- 만약 목록에 존재하고 c=1 이면 A는 이미 그것을 실행 오라클에게 실행-쿼리로 전송한 것이므로 오라클에게 받은 전달 메시지를 패스워드를 이용하여 PW-GKE형태로 바꾼 전달 메시지를 B에게 전송한다.
- 만약 목록에 존재하지 않으면 A는 목록에 $(U_U^s, 0)$을 추가한다. 이와 같은 경우나 또는 목록에 존재하는데 $c=0$일 때에는 A는 자신이 알고 있는 패스워드를 이용하여 시뮬레이션하여 나온 전달 메시지를 B에게 전송한다.

만약 전송-쿼리가 B의 α번째 전송/실행-쿼리이면 A는 다음과 같이 대응한다.

- 만약 전송-쿼리가 이미 B에 의해 손상-쿼리로 전송되어 졌다면 A는 추측한 α가 틀리게 된 것이므로 실패하게 된다.
- A는 전송-쿼리의 형태를 L_{list}목록에서 찾는다. 만약 목록에 존재하면 A는 추측한 α가 틀리게 된 것이므로 실패하게 된다. 그 밖의 경우에 A는 목록에 $(U_U^s, 1)$을 추가한고 B의 전송-쿼리를 그대로 실행 오라클에 전송하여 얻은 전달 메시지를 패스워드를 이용하여 PW-GKE의 전달 메시지로 변환한 후 이를 B에게 전송한다.

손상-쿼리. B가 손상-쿼리를 요청하면 A는 알고 있는 패스워드를 B에게 전송한다.

유출-쿼리. B가 임의의 세션에 대한 유출-쿼리(Reveal($\prod_U^s$))를 요청하는 경우에 이 유출-쿼리의 형태는 L_{list}목록에 있어야 한다. 그러므로 A는 목록에서 (U_U^s, z)를 찾을 수 있다. 만약 z=1 이면 A는 추측한 α가 틀리게 된 것이므로 실패하게 된다. 만약 z=0이면 A는 시뮬레이션을 통해 해당하는 세션에 대한 그룹 키를 계산 할 수 있으며, 계산된 그룹 키(K_U)를 B에게 전송한다.

테스트-쿼리. B가 임의의 세션에 대한 테스트-쿼리(Test($\prod_U^s$))를 요청하는 경우에 이 유출-쿼리의 형태는 L_{list}목록에 있어야 한다. 그러므로 A는 목록에서 (U_U^s, z)를 찾을 수 있다. 만약 z=0 이면 A는 추측한 α가 틀리게 된 것이므로 실패하게 된다. 만약 z=1이면 A는 B의 테스트-쿼리를 그대로 테스트 오라클에 요청한 후 그에 대한 결과 값을 B에게 전송한다.

Correct를 A가 추측한 α가 맞는 사건 이라하자. 만약 Correct와 $Succ_{pw}$가 일어나면 위의 시뮬레이션은 실패한다. 위의 시뮬레이션이 성공할 이점을 계산하면 다음과 같다.

$$
\begin{aligned}
&|2\Pr_A[Succ] - 1| \\
&= |2\Pr_B[Succ \wedge Correct \wedge \neg Succ_{pw}] + \\
&\Pr_B[\neg Correct \vee Succ_{pw}] - 1| \\
&= |2/q_t \cdot \Pr_B[Succ \wedge \neg Succ_{pw}] - \Pr_B[Succ_{pw}] + \\
&\Pr_B[\neg Correct | Succ_{pw}] + \Pr_B[Succ_{pw}] - 1| \\
&= |2/q_t \Pr_B[Succ] - 2/q_t \Pr_B[Succ \wedge Succ_{pw}] + \\
&\Pr_B[Succ_{pw}] + (q_t - 1)/q_t (1 - \Pr_B[Succ_{pw}]) - 1| \\
&\geq |2\Pr_B[Succ] - 1| - \\
&1/q_t |2\Pr_B[Succ \wedge Succ_{pw}] - \Pr_B[Succ_{pw}]
\end{aligned}
$$

가정에 의하여

$$|2\Pr_A[>] - 1| \leq Adv_{L-GKA}^{GKE-fs}(t, 1)$$

이므로

$$
\begin{aligned}
&Adv_{PW-GKE}^{PGKE-fs}(t, q_{ex}, q_s) \\
&\leq q_T \cdot Adv_{L-GKE}^{GKE-fs}(t, 1) + \Pr[Succ_{pw}(t)]
\end{aligned}
$$

[보조정리1] h를 랜덤 오라클이라 가정하고 N을 가능한 패스워드의 총 수, n을 사용자의 총 수라 하자. 해쉬-쿼리의 수가 q_h라 할때 프로토콜 PW-GKE의 인증부분인 패스워드를 공격하여 성공하는 확률은

다음과 같다.

$$\Pr[Succ_{pw}(t)] \le \frac{n(q_h-1)}{2} Adv_G^{CDH}(t) + \frac{q_s}{N}.$$

(증명) 패스워드 공격자는 전수조사와 같이 패스워드를 추측하여 이를 이용하여 그룹의 다른 사용자와 통신할 수 있다. 이때 추측한 패스워드가 실제 패스워드와 일치할 확률은 1/N 이다. 만약 공격자가 $k(k \le N)$번의 서로 다른 패스워드를 추측하여 공격을 시도할 때의 성공확률은 k/N 이다. 공격자는 이 공격을 전송-쿼리를 통하여 시도할 수 있기 때문에 결국 공격자가 패스워드 추측 공격을 통한 성공확률은 q_s/N이다.

패스워드를 공격하는 또 다른 방법은 전달 메시지(transcript)나 전송-쿼리를 이용하여 패스워드를 추측하는 것이다. 이 경우 우리는 패스워드를 공격하는 공격자 A를 이용하여 CDH를 푸는 공격자 B를 다음과 같이 구성할 수 있다.

최초에 CDH 문제 (g, g^x, g^y)가 공격자 B에게 주어진다고 가정하자. 그러면 공격자 B는 다음과 같이 A의 공격환경을 시뮬레이션 한다.

- B는 랜덤한 값 $(r_1, r_2, \ldots, r_n)$과 $i \in [1, n]$를 선택하여 $g^{r_1}, \ldots, g^{r_i}=g^x, g^{r_{i+1}}=g^y, \ldots, g^{r_n}$을 계산한 후 이를 B가 선택한 임의의 패스워드로 암호화한 $E_{pw}(g^{r_1}), \ldots, E_{pw}(g^{r_n})$값을 A에게 전달한다. A는 추측한 패스워드를 이용하여 이를 복호화 한다.
- A가 전송-쿼리를 요청하였을 때, B는 $g^{r_1}, \ldots, g^{r_n}$을 이용하여

$$g^{r_1 r_2}, \ldots, g^{r_{i-1} r_i}, g^{r_{i+1} r_{i+2}}, \ldots, g^{r_n r_1}$$

값을 계산한다. B는 이 계산된 값 각각에 대하여 대응되는 랜덤한 해쉬(hash) 값 $h_1, \ldots, h_{i-1}, h_{i+1}, \ldots, h_n$을 생성하고 계산할 수 없는 값인 $g^{r_i r_{i+1}}$에 대해서도 대응되는 랜덤한 해쉬 값 h_i를 생성하여 모든 해쉬 순서쌍 $(g^{r_i r_{i+1}}, h_j)$을 해쉬 테이블 h_{list}에 저장한다. 그리고 이 해쉬 값을 이용하여 $z_1, z_2, \ldots, z_n$을 계산하여 A에게 전송한다. 이때 z_i와 z_{i+1}의 값은 아래와 같다.

$$z_i = h_{i-1} \oplus h_i = h(g^{r_{i-1} r_i}) \oplus h(g^{r_i r_{i+1}}) = h(g^{r_{i-1} r_i}) \oplus h(g^{xy}),$$

$$z_{i+1} = h_i \oplus h_{i+1} = h(g^{r_i r_{i+1}}) \oplus h(g^{r_{i+1} r_{i+2}}) = h(g^{xy}) \oplus h(g^{r_{i+1} r_{i+2}})$$

- A가 해쉬-쿼리$(c_i, i=1, \ldots q_h)$를 요청하였을 때, B는 쿼리가 h_{list}에 있는지 확인하고 목록에 있으면 그에 대응하는 해쉬 값을 A에게 전송한다. 만약 요청한 쿼리(c_k)가 처음으로 목록에 없을 경우 B는 g^{xy}값에 대응하는 해쉬 값인 h_i값을 A에게 전송하고 c_k값을 저장한다. 그 외의 경우는 랜덤 값을 돌려준다.
- A가 마침내 패스워드를 출력하였을 때, B는 A가 출력한 패스워드가 자신이 알고 있는 패스워드와 동일한지 확인하고, 만약 동일한 패스워드라면 저장한 c_k값을 결과 값으로 출력한다.

B가 출력한 c_k값이 g^{xy}과 동일한 값이 되기 위해서는 A가 g^{xy}인 해쉬-쿼리를 요청해야 한다. A는 자신이 추측한 패스워드를 확인하기 위해서 전송-쿼리를 요청하여 받은 $z_1, z_2, \ldots, z_n$값들 중 적어도 하나의 값과 해쉬-쿼리를 이용하여 받은 z값과 비교하여 패스워드를 확인하는 방법을 반복하여 올바른 패스워드를 출력하게 된다.

만약, A가 패스워드 공격에 성공하여 올바른 패스워드를 출력하게 된다면 A는 비교의 대상으로 $z_1, z_2, \ldots, z_n$중 적어도 하나를 선택하게 되는데, 이때 z_i 또는 z_{i+1}을 선택할 확률은 $2/n$이다. 그리고 선택한 z_i(또는 z_{i+1})와 비교할 값을 얻기 위하여 요청하는 해쉬-쿼리에 반드시 $g^{r_{i-1} r_i}$와 $g^{r_i r_{i+1}} = g^{xy}$가 포함되어야 한다. 이 중 $g^{r_{i-1} r_i}$는 B의 해쉬 목록인 h_{list}에 포함되어 있고, $g^{r_i r_{i+1}}$는 목록에 없으며 B의 관점에서 $g^{r_i r_{i+1}}$이 목록에 없는 첫 해쉬-쿼리일 확률은 적어도 $1/(q_h-1)$이다. 그러므로 A가 패스워드 공격에 ε의 확률로 성공한다면 B는 적어도 $\frac{2}{n(q_h-1)}\varepsilon$의 확률로 CDH 문제를 풀게 된다.

V. 비교와 결론

본 논문에서는 전방향 안전성을 제공하는 패스워드 기반 인증된 그룹 키 교환 프로토콜을 제안 하였고 DDH 문제와 CDH 문제의 어려움에 근거하여

랜덤 오라클 모델과 이상적 암호화 모델에서 안전성을 증명하였다.

여기서 우리는 '지수승 연산'을 프로토콜의 모든 참여가가 지수승하는 총 수라 하고, '라운드 수'는 프로토콜 수행에 필요한 총 라운드 수 그리고 '메시지 길이'는 프로토콜의 각 사용자가 전송하는 총 메시지 길이라고 하자. 우리가 제시한 프로토콜은 상수 번의 통신 라운드를 요구하며 키 공유를 위해 각 사용자는 단지 1 라운드에서만 3번의 지수승 연산을 하고 2 라운드에서는 지수승 연산이 없다. 따라서 모든 사용자는 $O(n)$번의 모듈라 지수승이 요구되며, 1 라운드의 메시지를 브로트 캐스트하게 되면 각 사용자는 총 $2|g|$ 길이의 메시지를 전송하게 된다. 아래의 표에서 알 수 있듯이 Breeson이 제시한 패스워드 기반 그룹 키 교환 프로토콜과 비교하여 제시한 프로토콜은 매우 효율적임을 알 수 있다.

본 논문에서 우리는 랜덤 오라클 모델과 이상적 암호화 모델에서 안전성을 증명하였지만, 앞으로는 표준(standard) 모델에서 안전하고 계산 및 통신비용 효율적인 프로토콜에 대한 연구가 필요하다.

표 1 패스워드 기반의 GKE 프로토콜의 비교

	지수승 연산	라운드 수	메시지 길이
Breeson[7]	$O(n^2)$	$O(n)$	$\leq n\|g\|$
PW-GKE	$O(n)$	$O(1)$	$2\|g\|$

참 고 문 헌

[1] E. Bresson, O. Chevassut and D. Pointcheval, "Group Diffie-Hellman Key Exchange Secure Against Dictionary Attacks", Advances in Cryptology Asiacrypt'02, *Lecture Notes in Computer Science* Vol. 2501, *Springer-Verlag*, pp. 497~514, 2002.

[2] E. Bresson, O. Chevassut, D. Pointcheval and J-J Quisquater, "Provably Authenticated Group Diffie-Hellman Key Exchange", *In Proc. of 8th ACM CCCS*, pp. 255~264, November 2002.

[3] S. Bellovin and M. Merritt, "Encrypted key exchange: Password-based protocols secure against dictionary attacks", *In Proceedings of the Symposium on Security and Privacy*, pp. 72~84 IEEE, 1992.

[4] V. Boyko, P. MacKenzie and S. Patal, "Provably secure password-authenticated key exchange using Diffie-Hellman", In B. Prenel, editor, Advances in Cryptology Eurocrypt'00, *Lecture Notes in Computer Science* Vol. 1807, *Springer-Verlag*, pp. 156~171, 2000.

[5] M. Bellare, D. Pointcheaval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks", Advances in Cryptology Eurocrypt'00, *Lecture Notes in Computer Science*, Vol. 1807, *Springer-Verlag*, pp. 139~155, 2000.

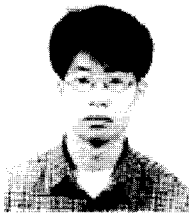
[6] O. Goldreich and Y. Lindell, "Session-key generation using human passwords only", In J. Killian editors, Advances in Cryptology, Crypto'01, *Lecture Notes in Computer Science* Vol. 2139, *Springer-Verlag*, pp. 408~432, 2001.

[7] J. Katz, R. Ostrovsky and M. Yung, "Efficient password-authenticated key exchange using human-memorable passwords", Advances in Cryptology Eurocrypt'01, *Lecture Notes in Computer Science* Vol. 2045, *Springer-Verlag*, pp. 475~494, 2001.

[8] S. M. Lee, H. J. Kim, D. H. Lee, J. I. Lim and C. S. Park "Scalable Gruop Key Management with Minimally Trusted Third Party", *In 4th International Workshop on Information Security Applications*, pp.575~583, Aug 2003.

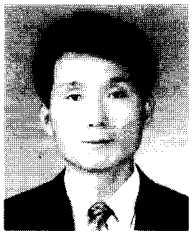
〈著者紹介〉

황 정 연 (Jung Yeon Hwang) 학생회원
1999년 2월: 고려대학교 수학과 학사
2003년 2월: 고려대학교 정보보호대학원 석사
2003년 3월~현재: 고려대학교 정보보호대학원(박사과정)
〈관심분야〉 암호 프로토콜, 암호이론, 네트워크 보안



최 규 영 (Kyu Young Choi) 학생회원
2002년 2월: 고려대학교 수학과 학사
2002년 3월~현재: 고려대학교 정보보호대학원 석사과정
〈관심분야〉 암호 프로토콜, 네트워크보안

이 동 훈 (Dong Hoon Lee) 정회원
1984년 2월: 고려대학교 경제학과 학사
1987년 2월: Oklahoma Univ. 전산학 석사
1992년 5월: Oklahoma Univ. 전산학 박사
1993년 3월~2000년 2월: 고려대학교 전산학과 정교수
2000년 3월~현재: 고려대학교 정보보호대학원 교수
〈관심분야〉 암호이론, 암호 프로토콜, 계산이론, 네트워크 보안



백 종 명 (Jong Myung Baik) 정회원
1983년 2월: 고려대학교 산업공학 학사
1991년~2000년: 한국 전자 통신 연구원 컴퓨터.소프트웨어 기술 연구소
2000년 2월: 고려대학교 전산학 석사
2000년 3월~현재: 고려대학교 정보보호대학원(박사과정)
2004년 1월~현재: (주) 유비넷 대표이사
〈관심분야〉 암호 프로토콜 응용, 시스템 보안