키 유도함수를 결합한 ID 기반 3자 복수키 동의 프로토콜☆

ID-based Tripartite Multiple Key Agreement Protocol Combined with Key Derivation Function

이 상 곤* 이 훈 재**
Sang-gon Lee Hoon-Jae Lee

요 약

복수키 동의 프로토콜의 목적은 단일키 동의 프로토콜의 거듭 실행에 비하여 계산량과 통신량 면에서 효율성을 얻기 위함이다. 최근에 ID 기반의 3자 복수키 동의 프로토콜들이 제안되었지만, unknown key-share 공격 혹은 impersonation 공격 등에 대한 취약함이 발견되어 모든 종류의 공격에 대하여 안전하면서 효율적인 ID 기반 인증된 3자 키 동의 프로토콜의 설계는 아직 미지의 문제로 남아있다. 본 논문에서는 단일키 동의 프로토콜과 키 유도함수를 결합한 복수키 동의 기법을 제안한다. 기존의 복수키 동의 프로토콜에 비하여 계산적 효율성을 증가시킬 수 있을 뿐 아니라, 안전성이 증명된 단일키 동의 프로토콜과 키 유도함수를 사용함으로써 안전성을 보장받을 수 있다.

Abstract

The purpose of the multiple key agreement protocol is to get efficiency in computational and communicational aspects compared to multiple executions of single key agreement protocol. However ID based tripartite multiple key agreement protocols have been proposed, it is reported that they can not resist unknown key-share attack or impersonation attack. How to design a secure and efficient ID-based authenticated tripartite multiple key agreement scheme to prevent all kinds of attacks remains an open problem. This paper proposes a multiple key agreement scheme combing the existing single key agreement protocol with a key derivation function. The proposed scheme can not only increase computational efficiency compared to the existing multiple key agreement protocol, but can ensure security of the proposed schemes by using a security proofed single key agreement protocol and key derivation function.

☞ Keyword : key agreement(키 동의), ID based(신원기반), Security(안전성), Unknown key-share attack(알려지지 않은 키 공유 공격), inpersonation attack(위장공격).

1. 서 론

키 동의 프로토콜은 온라인에서 안전한 거래를 위하여 둘 이상의 개체들 간에 공유하는 세션 키를 설정하는 프로토콜이다. 셋 이상이 키를 공유하는 상황을 conference keying 이라 한다. 셋이 참가하는 회의는 가장 흔한 전자회의의 규모 일 뿐 아니라, 두 통신 당사자를 위한 다양한 서비스를 제공해 줄 수 있다. 예를 들어 제3자는 회의의 의장이 되거나 심판자로서 참여할 수 있다. 동일한 그룹 멤버들이 여러 번의 세션을 실행하는 경우 각 세션마다 서로 다른 세션 키가 필요하게 된다. Joux[1]는 pairing을 이용한 3자간 키 동의 프로토콜을 제안하였다. 3자간 키 동의에 전형적인 Diffie-Hellman(DH) 키 교환방식을 이용할 경우 참여 개체 당 2회의 브

* 정 회 원: 동서대학교 컴퓨터정보공학부 부교수
nok60@dongseo.ac.kr
** 정 회 원: 동서대학교 컴퓨터정보공학부 부교수
hjlee@dongseo.ac.kr
[2006/01/20 투고 - 2006/02/17 심사 - 2006/05/15 심사완료]
☆ 본 연구는 2005년 동서대학교 교내특별연구 지원에 의하여 연구되었음.

로드 캐스팅 메시지 전송이 요구되지만, Joux의 프로토콜은 개체 당 1회의 메시지 전송으로 키 동의를 이룬다는 점에서 획기적인 프로토콜로 인식되고 있다. 하지만 Joux 기법 역시 DH 기법과 마찬가지로 인증 기능을 가지고 있지 않으므로 man-in-the-middle 공격에 취약하다. Joux의 기법에 인증을 제공하는 방법으로는 공개키 인증서를 이용하는 기법[2], ID를 이용하는 기법[3], 그리고 패스워드를 이용하는 기법[4]으로 분류된다. ID를 이용한 인증 기법은 인증서 관리와 관련된 작업을 면제받는 장점이 있다.

1.1 관련된 연구들

최근 들어 Joux 기법을 근간으로 한 번의 키 동의 프로토콜 수행으로 복수의 키를 설정할 수 있는 ID 기반 복수키 동의 프로토콜들이 제안되었다[3,5-7]. 일반적인 키 동의 프로토콜은 한번의 프로토콜 수행으로 단일 세션키를 동의하게 되지만 Liu 등[3]이 제안한 기법(LZC 프로토콜)은 3자간에 한번의 프로토콜 수행으로 8개의 서로 다른 세션키를 동의 할 수 있으며, Kim 등[5]은 2 개체 간에 다수의 세션 키들을 동의하는 프로토콜을 제안하였다. 그리고 박 등[6]은 다중 KGC(key generator center) 환경에서 실행가능하도록 Liu 등의 프로토콜을 확장하였다.

하지만 Shim 등[7]은 LZC 프로토콜에서 unknown key-share 공격에 취약함을 발견하였으며 수정된 프로토콜을 제시하였다. 하지만 J. Chou 등[8]은 Shim 등의 프로토콜이 impersonation 공격에 취약함을 보였지만 대안을 제시하지는 않았다. 따라서 모든 종류의 공격에 안전한 ID 기반 인증된 3자 복수키 동의 프로토콜의 설계는 아직 미지의 문제로 남아있다.

1.2 연구 동기 및 연구내용

타원곡선 pairing과 ID를 기반으로 한 키 동의 프로토콜은 PKI 기반이 요구되지 않으며 1라운드에 키 동의를 이룬다는 점에서 매력적이다. 지금까지 제안된 1라운드 ID 기반 인증된 3자 복수키 동의 프로토콜은 참여 개체가 다수의 키 동의용 메시지를 보내도록 함으로써 이들 메시지의 조합에 따라 복수 세션 키 설정이 가능하도록 하였다. 하지만 [7, 8]에서 보는 바와 같이 다중 메시지 전송으로 인하여 어려가지 공격에 취약함을 알 수 있다. 복수키 동의 프로토콜의 목적은 단일키 동의 프로토콜의 거듭 실행과 비교하여 계산적인 면과 통신비용적인 면에서 효율성을 얻기 위함이다. 그러나 단일키 동의 프로토콜의 1회 실행에 비하여 지수연산, 스칼라 곱셈 및 pairing 등의 무거운 연산과 보다 많은 메시지 전송이 요구된다.

3자간 복수키를 동의함에 있어 기존의 다중 메시지 전송에 의한 복수키 동의 프로토콜 이외에 쉬운 여러 가지 다양한 방법이 있다. 안전성이 증명된 단일키 동의 프로토콜을 실행한 후 공유된 키를 마스터키로 하여 파생 키들을 생성하면 될 것이다. 파생 키들을 만드는 방법에는 대칭 키 암호 알고리듬의 라운드 키 생성 알고리듬을 사용하거나 키 유도함수(key derivation function : KDF)[9]를 사용할 수 있다.

본 논문에서는 ID 기반 3자 단일키 동의 프로토콜과 키 유도함수를 사용한 복수키 동의 프로토콜을 제시한다. 그리고 제시된 프로토콜의 안전성을 고찰하고, 계산량과 통신량 면에서 기존에 발표된 프로토콜들과 비교·검토한다. 물론 제시된 방법은 누구나 일반적으로 생각할 수 있는 다중 세션 키 생성 기법이지만, 본 논문의 의도는 "기존에 제안된 복수키 생성 기법 보다는 본 논문에서 제안하는 기법이 훨씬 효율적이며 안전성 면에서도 저하되지 않는다."는 것이며, "아직 이런 점에 관하여 논의를 제기한 바가 없다."는 것이다. 실제적인 응용 시스템에서는 암호 프로토콜의 암호학적 강인성과 구현의 복잡성 등을 협상할 필요가 있다.

복수키 동의 프로토콜 선택의 문제를 직면하는 경우, 본 논문에서 제기하는 기법은 한 가지 대안이 될 수 있다.

본 논문의 순서는 다음과 같다. 2장에서는 기존의 ID 기반 3자 키 동의 프로토콜들을 소개한다. 그리고 3장에서는 본 논문에서 제시하는 ID 기반 3자 키 복수키 동의 프로토콜에 관하여 서술하고, 안전성을 검토하고, 그리고 기존의 프로토콜들과 계산량 및 통신량 면에서 비교·검토한다. 마지막으로 4장에서 결론을 맺는다.

2. 기존의 ID 기반 3자 복수키 동의 프로토콜

본 장에서는, 비록 안전하지 않은 것으로 보고되었지만, 기존에 발표된 ID 기반 복수키 동의 프로토콜들[3,7]을 살펴본다. Liu 등[3]의 프로토콜을 LZC 프로토콜, Shim 등[7]의 프로토콜을 SW 프로토콜로 표기하기로 한다. 프로토콜을 설명하기에 필요한 기초 사항을 먼저 설명한다.

2.1 수정된 Weil pairing과 ID 기반 공개 키 시스템

p를 $p=2 \pmod 3$이고 어떤 소수 $q>3$에 대하여 $p=6q-1$인 소수, E를 F_p상의 $y^2=x^3+1$에서 정의된 초 특이 타원곡선(supersingular elliptic curve)이라 둔다. 그리고 P를 위수가 $q=(p+1)/6$인 그룹의 생성원(generator), μ_q를 위수 q의 모든 원소를 포함하는 부 그룹 $F_{p^2}^*$이라 둔다. H_1과 H를 $H_1,\ H:\{0,1\}^* \rightarrow F_p$인 충돌 저항 해쉬함수라 둔다. 타원곡선 상에서의 Weil pairing은 사상(mapping) $e=: G_q \times G_q \rightarrow \mu_q$이다. 수정된 Weil pairing은 다음과 같이 정의된다.

$$\hat{e}=: G_q \times G_q \rightarrow \mu_q,\ \hat{e}(P,Q)=e(P,\phi(Q))$$

여기서 $\phi(x,y)=(\zeta x,y), 1 \neq \zeta \in F_{p^2}^*$가 $x^3-1=0 \pmod p$의 해이고, G_q는 위수가 q인 점들의 그룹이다.

KGC는 자신의 공개 키 $P_{KGC}=s \cdot P$를 계산한다. 여기서 $s \in Z_q^*$는 KGC의 비밀 키이다. KGC는 시스템 파라미터 $params=\{p,q,E,P,P_{KGC},\hat{e},H_1,H\}$를 공개한다. 각 사용자는 자신의 식별자 정보 ID를 KGC에 제출한다. 그러면 KGC는 사용자의 공개 키를 $Q_{ID}=H(ID)$로 두고, 사용자의 비밀키 $S_{ID}=s \cdot Q_{ID}$를 계산하여 사용자에게 반환한다.

2.2 기존의 3자 복수키 동의 프로토콜

기존의 3자 복수키 동의 프로토콜은 다음과 같이 3단계 1라운드에 완성된다.

단계 1: 복수 메시지 전송(broadcasting)
단계 2: 메시지 서명 검증
단계 3: 복수 메시지를 이용한 복수키 생성

LZC 프로토콜을 먼저 설명한다.

ID 기반 공개키 암호시스템에서 프로토콜 참여자 A, B, C는 각각 자신의 식별자 ID_A, ID_B, ID_C로부터 공개키 $Q_A=H_1(ID_A), Q_B=H_1(ID_B), Q_C=H_1(ID_C)$를 만들고, KGC로부터 각각 자신의 개인 키 $S_A=sQ_A, S_B=sQ_B, S_C=sQ_c$를 발급 받는다. 세션 비밀 값으로 프로토콜 참여자 A, B, C는 각각 두 개의 난수 $(a,a'),(b,b'),(c,c') \in Z_q^*$를 선택한 후, 단계 1 과정으로 다음과 같이 메시지를 교환한다.

$$A \rightarrow B,C:\ P_A=aP,\ P_A'=a'P,\ T_A=H(P_A,P_A')S_A+aP_A',$$

$B \rightarrow A, C$: $P_B = bP, P_B' = b'P, T_B = H(P_B, P_B')S_B + bP_B'$,

$C \rightarrow A, B$: $P_C = cP, P_C' = c'P, T_C = H(P_C, P_C')S_C + cP_C'$.

단계 2 과정으로 참여자 A는 아래와 같이 수신한 메시지의 서명을 검증한다.

$$\hat{e}(T_B + T_C, P) = \hat{e}(H(P_B, P_B')Q_B + H(P_C, P_C')Q_C, P_{KGC})\hat{e}(P_B, P_B')\hat{e}(P_C, P_C').$$

만일 위의 등식이 성립하면 단계 3 과정으로 A는 B, C로부터 수신한 메시지들을 이용하여 다음과 같이 8개의 세션 키를 생성한다.

$$K_A^{(1)} = \hat{e}(P_B, P_C)^a,\ K_A^{(2)} = \hat{e}(P_B, P_C')^a,$$
$$K_A^{(3)} = \hat{e}(P_B', P_C)^a,\ K_A^{(4)} = \hat{e}(P_B', P_C')^a,$$
$$K_A^{(5)} = \hat{e}(P_B, P_C)^{a'},\ K_A^{(6)} = \hat{e}(P_B, P_C')^{a'},$$
$$K_A^{(7)} = \hat{e}(P_B', P_C)^{a'},\ K_A^{(8)} = \hat{e}(P_B', P_C')^{a'}.$$

참여자 B도 아래와 같이 수신한 메시지의 서명을 검증한다.

$$\hat{e}(T_A + T_C, P) = \hat{e}(H(P_A, P_A')Q_A + H(P_C, P_C')Q_C, P_{KGC})\hat{e}(P_A, P_A')\hat{e}(P_C, P_C').$$

만일 위의 등식이 성립하면 단계 3 과정으로 B는 다음과 같이 8개의 세션 키를 생성한다.

$$K_B^{(1)} = \hat{e}(P_A, P_C)^b,\ K_B^{(2)} = \hat{e}(P_A, P_C')^b,$$
$$K_B^{(3)} = \hat{e}(P_A, P_C)^{b'},\ K_B^{(4)} = \hat{e}(P_A, P_C')^{b'},$$
$$K_B^{(5)} = \hat{e}(P_A', P_C)^b,\ K_B^{(6)} = \hat{e}(P_A', P_C')^b,$$
$$K_B^{(7)} = \hat{e}(P_A', P_C)^{b'},\ K_B^{(8)} = \hat{e}(P_A', P_C')^{b'}.$$

참여자 C도 아래와 같이 수신한 메시지의 서명을 검증한다.

$$\hat{e}(T_A + T_B, P) = \hat{e}(H(P_B, P_B')Q_B + H(P_A, P_A')Q_A, P_{KGC})\hat{e}(P_B, P_B')\hat{e}(P_A, P_A').$$

만일 위의 등식이 성립하면 단계 3 과정으로 C는 다음과 같이 8개의 세션 키를 생성한다.

$$K_C^{(1)} = \hat{e}(P_A, P_B)^c,\ K_C^{(2)} = \hat{e}(P_A, P_B)^{c'},$$
$$K_C^{(3)} = \hat{e}(P_A, P_B')^c,\ K_C^{(4)} = \hat{e}(P_A, P_B')^{c'},$$
$$K_C^{(5)} = \hat{e}(P_A', P_B)^c,\ K_C^{(6)} = \hat{e}(P_A', P_B)^{c'},$$
$$K_C^{(7)} = \hat{e}(P_A', P_B')^c,\ K_C^{(8)} = \hat{e}(P_A', P_B')^{c'}.$$

이상과 같이 하여 참여자 A, B, C는 8개의 동일한 세션 키를 공유하는데 성공하였다.

다음으로 SW 프로토콜을 살펴본다.

프로토콜 참여자 A, B, C의 개인키 셋업은 LZC 프로토콜의 개인키 셋업 절차와 동일하다. 세션 비밀 값으로 프로토콜 참여자 A, B, C는 각각 두 개의 난수 $(a,a'),(b,b'),(c,c') \in Z_q^*$ 를 선택한 후, 단계 1과정으로 다음과 같이 메시지를 교환한다.

$A \rightarrow B, C$: $P_A = aP, P_A' = a'P, T_A = S_A + a^2P + a'P_{KGC}$,

$B \rightarrow A, C$: $P_B = bP, P_B' = b'P, T_B = S_B + b^2P + b'P_{KGC}$,

$C \rightarrow A, B$: $P_C = cP, P_C' = c'P, T_C = S_C + c^2P + c'P_{KGC}$.

단계 2 과정으로 참여자 A는 아래와 같이 수신한 메시지의 서명을 검증한다.

$$\hat{e}(T_B + T_C, P) = \hat{e}(Q_B + Q_C + P_B' + P_C', P_{KGC})\hat{e}(P_B, P_B)\hat{e}(P_C, P_C).$$

만일 위의 등식이 성립하면 단계 3 과정으로 A는 B, C로부터 수신한 메시지들을 이용

하여 LZC와 동일한 방법으로 8개의 세션 키를 계산한다.

참여자 B도 아래와 같이 수신한 메시지의 서명을 검증한다.

$$\hat{e}(T_A + T_C, P) = \hat{e}(Q_A + Q_C + P_A' + P_C', P_{KGC})\hat{e}(P_A, P_A)\hat{e}(P_C, P_C).$$

만일 위의 등식이 성립하면 단계 3 과정으로 B는 A, C로부터 수신한 메시지들을 이용하여 LZC와 동일한 방법으로 8개의 세션 키를 계산한다.

참여자 C도 아래와 같이 수신한 메시지의 서명을 검증한다.

$$\hat{e}(T_B + T_A, P) = \hat{e}(Q_B + Q_A + P_B' + P_A', P_{KGC})\hat{e}(P_B, P_B)\hat{e}(P_A, P_A).$$

만일 위의 등식이 성립하면 단계 3 과정으로 C는 A, B로부터 수신한 메시지들을 이용하여 LZC와 동일한 방법으로 8개의 세션 키를 계산한다.

3. 제시된 ID 기반 3자 복수키 동의 프로토콜

Nella 등[10]은 Joux의 프로토콜과 ID 기반 암호 기법을 결합하여 1라운드에 실행되는 ID 기반 3자 단일키 동의 프로토콜을 제안하였다. 하지만 Shim 등[11]은 Nella 등의 프로토콜이 여전히 man-in-the-middle 공격에 취약함을 보이고 해결방안을 제시하였다. 본 논문에서 1라운드에 실행되는 ID 기반 3자 단일키 동의 프로토콜과 키 유도함수를 결합한 복수키 동의 프로토콜을 제시하고 안전성을 고찰한다. 그리고 계산량과 통신량 면에서 기존에 발표된 복수키 동의 프로토콜과 비교 · 검토한다. 제시되는 프로토콜의 장점은 기반이 되는 3자 단일키 프로토콜을 선택할 수 있다는 것이며, 선택되는 단일키 동의 프로토콜의 안전성이 증명되는 한 동의되는 복수키의 안전성은 보장된다는 것이다.

3.1 제시된 복수키 동의 프로토콜의 절차

본 절에서는 Shim의 3자 단일키 교환 프로토콜[11]과 PKCS #1의 키 유도함수[9]를 결합한 ID 기반 인증된 3자 복수키 동의 프로토콜을 설명한다. 제시하는 복수키 동의 프로토콜은 아래 같이 4단계 절차로 구분해 볼 수 있다.

단계 1: 단일 메시지 전송
단계 2: 메시지 서명 검증
단계 3: 단일키 생성
단계 4: 키 유도함수를 사용한 복수키 생성

단계 1에서 1라운드 메시지 전송이 이루어진 다음 각 프로토콜 참여자는 독립적으로 연속하여 2, 3, 4 단계를 실행하여 복수키를 동의하게 된다.

프로토콜 참여자 A, B, C는 단계 1과정으로 다음과 같이 메시지를 교환한다.

$$A \rightarrow B, C: P_A = aP,\ T_A = H(P_A)S_A + aP_{KGC},$$
$$B \rightarrow A, C: P_B = bP,\ T_B = H(P_B)S_B + bP_{KGC},$$
$$C \rightarrow A, B: P_C = cP,\ T_C = H(P_C)S_C + cP_{KGC}.$$

단계 2 과정으로 참여자 A는 아래와 같이 수신한 메시지의 서명을 검증한다.

$$\hat{e}(T_B + T_C, P) = \hat{e}(P_{KGC}, H(P_B)Q_B + H(P_C)Q_C + P_B + P_C).$$

만일 위의 등식이 성립하면 A는 단계 3 과정으로 다음과 같이 1개의 세션 키를 생성한다.

$$K_A = \hat{e}(P_B, P_C)^a = \hat{e}(P,P)^{abc}.$$

참여자 B 역시 단계 2 과정으로 아래와 같이 수신한 메시지의 서명을 검증한다.

$$\hat{e}(T_A + T_C, P) = \hat{e}(P_{KGC}, H(P_A)Q_A + H(P_C)Q_C + P_A + P_C).$$

만일 위의 등식이 성립하면 B는 단계 3 과정으로 다음과 같이 1개의 세션 키를 생성한다.

$$K_B = \hat{e}(P_A, P_C)^b = \hat{e}(P,P)^{abc}.$$

참여자 C 역시 단계 2 과정으로 아래와 같이 수신한 메시지의 서명을 검증한다.

$$\hat{e}(T_A + T_B, P) = \hat{e}(P_{KGC}, H(P_A)Q_A + H(P_B)Q_B + P_A + P_B).$$

만일 위의 등식이 성립하면 C는 단계 3 과정으로 다음과 같이 1개의 세션 키를 생성한다.

$$K_C = \hat{e}(P_A, P_B)^c = \hat{e}(P,P)^{abc}.$$

이제 $K = K_A = K_B = K_C$이 되어 3명의 프로토콜 참여자는 동일한 세션 마스터 키 K를 공유하였다.

단계 4과정으로 PKCS #1의 키 유도함수로부터 다중 세션 키를 생성한다. 키 유도 함수의 작동은 주어진 입력 $Z = K$에 대하여 다음과 같이 계산한다.

• Parameter definition:

- maskLen : KDF에서 출력될 전체 키 길이(in Octet)
- hLan : 하쉬함수의 길이(in Octet)
- Z : seed 데이터

• 키 유도함수 계산절차

단계 1. $T = empty\ octet\ string$

단계 2. for c = 0 to $\lceil maskLan/hLen \rceil$ - 1
$T = T \| Hash(Z \mid c)$

단계 3. T의 선두 maskLen(octets)를 키 스트링으로 출력

위의 절차와 같이 키 유도함수는 마스터 키(Z)를 다수 개의 해쉬함수에 적용하여 그 출력을 연접하여 출력한다. 키 유도함수는 해쉬함수 출력길이의 정수배 단위로 필요한 만큼의 길이로 출력을 낼 수 있다. 세션키는 정해진 길이단위로 잘라서 사용한다. 참여자 A, B, C는 단계 3을 마치면 동일한 세션 마스트 키를 공유하게 되고, 각 참여자는 키 유도함수를 적용하여 동일한 세션 키들을 계산한다.

3.2 제시된 복수키 동의 프로토콜의 안전성 검토.

안전한 키 동의 프로토콜 설계를 위하여 Wilson과 Menezes[12]는 여러 가지 바람직한 안전성 속성을 정의하였다. 바람직한 안전성 속성으로는 known session key security, forward secrecy, no key-compromise impersonation, unknown key-share, no key control 등이다.

제시된 ID 기반 복수키 동의 프로토콜은 단일키 동의 프로토콜을 1회 실행한 후, 그 출력에 키 유도함수를 적용한다. 따라서 전체 프로토콜의 안전성은 단일키 동의 프로토콜의 안전성을 상속하므로, 단일키 동의 프로토콜은 안전하다는 가정 하에, 키 유도함수 적용 이전까지의 출력에 대해서는 안전성 검토에서 제외한다. 마스터 세션 키 생성 후, 파생키를 생성하는 키 유도함수 적용 부분의 안전성을 살펴보자.

1) 알려진 세션 키 안전성(known session key security)

공격자(adversary)가 이전의 세션 키를 알았더라도 현재 수행 중인 프로토콜의 세션 키를 얻을 수 없어야 한다. 안전한 해쉬함수의 사용을 가정하면, 동일한 마스터 키 입력에 대하여 서로 다른 해쉬 값을 얻게 되고, 이들 출력 중 한 개가 노출된다고 하더라도 해쉬함수의 일방향성 때문에 세션 마스터 키 값을 알 수 없으므로 다른 해쉬의 출력값을 알 수 없게 되어 알려진 세션 키 안전성을 제공한다.

2) 전방향 안전성(forward secrecy)

프로토콜에 참여한 한 개체 이상의 장기 개인 키(long-term key)가 노출되더라도 이전의 세션 키의 기밀성에 영향을 미치지 않아야 한다. 단일키 동의 프로토콜이 전방향 안전성을 만족하면, 해쉬함수의 일방향 특성에 의해서 키 유도함수 적용부분에서도 전방향 안전성이 제공된다.

3) 알려지지 않는 키 공유 불가성(no unknown key-share)

개체 A가 개체 B와 키를 공유하고 있을 때 공격자 C가 A를 강요하여 키를 공유할 수 있으면 프로토콜은 *알려지지 않는 키 공유 공격*을 당하고 있다고 한다. 이 공격은 단일키 동의 프로토콜에서 먼저 만족되어야 할 특성이며 키 유도함수의 적용이후에는 적용되지 않는다.

4) 키 제어 불가성(No key control)

프로토콜 참여자 중에서 특정 참여자 또는 공격자가 미리 선택된 값이나 예측할 수 있는 값으로 세션 키를 생성하도록 강요할 수 없어야 한다. 이 공격 역시 단일키 동의 프로토콜에서 먼저 만족되어야 할 특성이며 키 유도함수의 적용이후에는 적용되지 않는다.

5) 키 탈취 위장 불가성(No key-compromise impersonation)

프로토콜 한 참여자 A의 장기 개인키를 탈취였더라도 공격자가 다른 참여자에 대하여 참여자 A로 위장 할 수 없어야 한다. 이 공격 역시 단일키 동의 프로토콜에서 먼저 만족되어야 할 특성이며 키 유도함수의 적용이후에는 적용되지 않는다.

6) 기타 안전성을 위협하는 요소에 대한 검토

키 유도함수는 seed가 노출되면 복수 세션 키는 모두 노출된다. 하지만 Seed는 안전한 단일키 동의 프로토콜로부터 생성되므로 프로토콜 메시지의 관측에 의하여 노출되지는 않는다. Seed가 노출되는 은 키 프로토콜이 탑재된 장치의 탈취에 의한 노출이 고려될 수 있다. 이런 상황은 기존의 복수키 동의 프로토콜에서도 마찬가지 상황이다. 즉, 최소한 한 참여자의 장치가 탈취당하여 두 개의 세션 비밀 값(예를 들어 참여자 A의 a와 a')이 노출되면 관측된 프로토콜 메시지로부터 복수 세션 키를 계산할 수 있게 된다. 입력값 seed에 대한 해쉬함수 H의 출력 H(seed)만 주어졌을 때, 입력값 seed를 찾을 확률은 $1/2^n$이다. 여기서 n은 해쉬함수의 출력의 비트길이이다. 일반적으로 256 이상이면 무시할 수 있는 확률로 안전한 것으로 알려져 있다. 따라서 seed 노출에 대한 안전성은 두 프로토콜이 거의 동일하다고 할 수 있다.

키 유도함수에는 해쉬함수가 사용된다. SHA-1의 충돌쌍이 X. Wang 등[13]에 의하여 발견되었다. 따라서 ISO/IEC JTC/SC27에서는 더욱 안전한 해쉬함수(SHA-224, 256, 384, 512, 및 Whrlpool)를 사용할 것을 권고하고 있다[14].

3.3 복잡도 비교

표 1에 본 논문에서 제시한 프로토콜과 기존의 프로토콜인 LZC 및 SW을 계산량과 통신

량 면에서 비교하여 나타내었다. 표에 기록된 숫자는 프로토콜 1회 수행에 객체 별로 요구되는 값이다. 프로토콜에 사용되는 주요 연산으로는 pairing, 스칼라 곱셈, 지수연산, 해쉬연산 등이다. 4가지 연산 가운데 pairing이 가장 복잡하고 그 다음이 지수연산이다. 제시된 프로토콜은 기존의 프로토콜에 비하여 8개의 세션 키 생성을 기준으로 5회의 pairing, 7회의 지수연산을 절약할 수 있으며, 스칼라 곱셈 연산은 1회 많거나 적다. 세션키 크기와 해쉬함수 크기를 256비트로 가정하면 8개의 복수키 생성에 8회의 해쉬연산이 소요된다. 하지만 pairing이나 지수 연산에 비하면 계산량이 매우 적으므로 추가비용은 미미하다고 할 수 있다. 또한 제시된 복수키 동의 프로토콜은 키 유도함수의 출력 길이를 증가시킴(해쉬함수의 추가적인 적용)으로써 공유 세션 키 수를 증가 시킬 수 있는 장점이 있다. 메시지량 면에서도 제시된 프로토콜이 개체 당 1개의 메시지 전송을 줄일 수 있으므로 대역폭 사용 효율성이 높다.

〈표 1〉 프로토콜의 계산량 및 대역폭 비교

Protocol	# 키 요소	Computational overhead per entity			
		# pairing	# 지수연산	# 스칼라 곱셈	# 해쉬연산
ZLC	3	8	8	6	3
SW	3	8	8	4	0
Ours	2	3	1	5	16

4. 결 론

기존에 발표된 복수키 프로토콜들의 의도는 1회의 프로토콜 실행으로 다수 개의 세션 키를 얻음으로써 단일키를 생성 프로토콜을 여러 번 실행하는 것보다 계산과 통신적인 면에서 효율성을 얻기 위함이다. 하지만 지금까지 1라운드에 실행되는 ID 기반 인증된 복수키 프로토콜들이 제안된 바 있으나 모두 안전성에 결함이 있는 것으로 알려져 있어, 안전한 복수키 동의 프로토콜의 설계는 아직 미지의 문제로 남아있다.

본 논문에서는 ID 기반 복수키 동의 프로토콜로서 단일키 동의 프로토콜과 키 유도함수를 결합한 기법을 제시하였다. 제시된 복수키 동의 프로토콜의 안전성은 사용되는 단일키 동의 프로토콜의 안전성에 많이 의존한다. 단일 프로토콜이 안전성 속성들을 만족할 경우 본 논문에서 제시된 프로토콜의 안전성 속성들도 만족됨을 알았다.

제시된 프로토콜은 기존의 프로토콜에 비하여 8개의 세션 키 생성을 기준으로 5회의 pairing, 7회의 지수연산을 절약할 수 있으며, 통신량 면에서도 단말기 당 1개의 메시지를 절약할 수 있다. 따라서 제시된 기법은 안전성 면에서 큰 저하를 초래하지 않으며 계산량 및 통신량 면에서 효율적이다.

참 고 문 헌

[1] A. Joux, " A One-round Protocol for Tripartite Diffie-Hellman," In W. Bosma, editor, Proceedings of Algorithmic Number Theory Symposium - *ANTS IV, LNCS 1838*, pp.385-394, Springer-Verlag, 2000.

[2] S. Al-Riyami and K. Paterson, "Tripartite Authenticated Key Exchange Protocols from Pairings," *IMA Conference on Cryptography and Coding, LNCS 2890*, pp.332-359, Springer-Verlag, 2003.

[3] S. Liu, F. Zhang, and K. Chen, "ID-based Tripartite Key Agreement Protocol with Paring," 2003 IEEE International Symposium of Information Theory, 2003, pp.136-143, or available at *Cryptology ePrint Archive*. Report 2002/122.

[4] S. Lee, Y. Hitchcock, Y. Park, and S. Moon, "Provably Secure Password

Protected Key Exchange Protocol Based on Elliptic Curves," in *the 12th Annual Workshop on Selected Areas in Cryptography(SAC2005), LNCS 3897,* pp.205-220, Springer-Verlag, 2006.

[5] K. Kim, E. Ryu and K. Yoo, "ID-Based Authenticated Multiple-Key Agreement Protocol from Parings," Springer-Verlag, *International Conference on Computational Science and its Applications(ICCSA 2004). LNCS 3046*, pp.627-689, Springer-Verlag, 2004.

[6] 박영호, 이경현, "효율성을 개선한 신원기반 3자간 복수 키 합의 프로토콜", *정보보호학회논문지 제15권 제3호*, pp.77-88. 2005.

[7] K. Shim and S. Woo, "Weakness in ID-based One Round Authenticated Tripartite Multiple-key Agreement Protocol with Pairings, Applied Mathematics and Computation. vol.166, No.3, pp.523-530, Elsevier, 2005.

[8] J. Chou, C. Lin, and C. Chiu, "Weekness of Shim's New ID-based Tripartite Multiple-Key Agreement Porotocol," *Cryptology ePrint Archive*. Report 2005/457, 2003. Available at http://eprint.iacr.org/2003/115.pdf.

[9] PKCS #1 v2.1: RSA Cryptography Standard, RSALaboratories: ftp://ftp.rsasecurity.com/pub/pkcs/pksc-1/pkcs-1v2-1.pdf.

[10] D. Nella and K. C. Reddy, "ID-based tripartite authenticated key agreement protocols from pairings, available at http://eprint.iacr.org/2003/004.

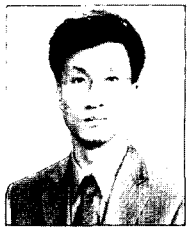
[11] K. Shim, "Cryptanalysis of ID-based Tripartite Authenticated Key Agreement Protocols," *Cryptology ePrint Archive*. Report 2003/115, 2003. Available at http://eprint.iacr.org/2003/115.pdf.

[12] A. Menezes, M. Qu, and S. A. Vanstone, "Some Key Agreement Protocols Providing implicit Authentication," Workshop on Selected Areas in Cryptography(SAC '95), pp. 22-23, 1995.

[13] Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu, "Finding Collisions in the Full SHA-1." Crypo'05, available at http://www.infosec.sdu.edu.cn/paper/sha1-crypto-auth-new-2-yao.pdf

[14] 한국정보보호진흥원, 정보보호기술 표준화 동향, 2005년

◐ 저 자 소 개 ◑



이 상 곤 (Sanggon Lee)

1986년 2월 경북대학교 전자공학과 졸업(학사)
1988년 2월 경북대학교 대학원 전자공학과 졸업(석사)
1993년 2월 경북대학교 대학원 전자공학과 졸업(박사)
1991년 3월 - 1997년 2월 창신대학 정보통신과 조교수
1997년 3월 - 현재 동서대학교 컴퓨터정보공학부 부교수
관심분야 : 암호프로토콜, 네트워크보안, DRM

이 훈 재 (Hoon-Jae Lee)

1985년 2월 경북대학교 전자공학과 졸업(학사)
1987년 2월 경북대학교 대학원 전자공학과 졸업(석사)
1998년 2월 경북대학교 대학원 전자공학과 졸업(박사)
1987년 1월 - 1998년 1월 국방과학연구소 선임연구원
1998년 3월 - 2002년 2월 경운대학교 컴퓨터공학과 조교수
2002년 3월 - 현재 동서대학교 컴퓨터정보공학부 부교수
관심분야 : 정보보안, 네트워크보안, 부채널공격/방어, etc