情報保護學會論文誌
第13券 第5號, 2003. 10

Gap Diffie-Hellman 군에 기반한 전방향 안전성을 갖는 서명 기법*

강 보 경**, 박 제 홍**, 한 상 근**

A New Forward-Secure Signature Scheme based on GDH groups

Bo-Gyeong Kang**, Je Hong Park**, Sang-Geun Hahn**

요 약

보통의 공개 열쇠 암호 시스템에서 비밀 열쇠가 노출되면 그 비밀 열쇠로 생성된 암호문이나 서명 등의 해독, 위조가 가능하다. 이런 비밀 열쇠 노출의 위험성은 상대적으로 물리적인 보안이 이루어지지 않은 핸드폰, 스마트카드 등과 같은 열쇠 저장 장치에 가장 큰 위협이 되고 있다. 따라서 현재 비밀 정보가 노출되어도 과거의 비밀 열쇠에 의해 생성된 정보는 안전하게 유지되어야 한다는 전방향 안전성 개념(forward security)은 열쇠교환 프로토콜 및 여러 암호 요소(primitive)들이 필수적으로 만족해야 할 조건이다. 본 논문에서는 Gap Diffie-Hellman 군을 사용하여 전방향 안전성을 만족하는 서명 기법을 제안한다. 제안된 서명 기법은 계산적 Diffie-Hellman 문제의 어려움을 기반으로 선택 메시지 공격 모델에 대한 증명가능한 안전성을 가진다.

ABSTRACT

We often use cryptographic systems on small devices such as mobile phones, smart cards and so on. But such devices are delicate against the threat of key exposure of secret keys. To reduce the damage caused by exposure of secret keys stored on such devices, the concept of forward security is introduced. In this paper, we present a new forward secure signature scheme based on Gap Diffie-Hellman groups. Our scheme achieves security against chosen-message attacks under the computational Diffie-Hellman assumption in the random oracle model.

keyword : *forward secure, signature scheme, binary tree, provable security*

Ⅰ. 서 론

암호 시스템에서 계산의 상당부분은 물리적인 보안이 미흡한 장치(스마트카드, 모바일 폰, PC 등)에서 이루어진다. 이와 같은 장치를 이용할 경우 발생할 수 있는 열쇠 노출의 피해를 줄이기 위해서 암호 시스템은 기본적으로 현재 사용중인 비밀 열쇠가 노출되어도 이전 비밀 열쇠를 이용하여 얻어진 정보들의 안전성은 보장되는 전방향 안전성(forward security)을 만족시켜야 한다. 열쇠교환 프로토콜에서 필수적인 안전성 조건으로 처음 소개된 전방향 안전성은 쌍방간의 상호 작용(interactive protocol)에 의해 열쇠를 공유하여 사용하고 일정 시간이 지나면 다시 상호 열쇠교환 프로토콜을 수행하여 공유 열쇠를 갱

* 본 연구는 한국과학재단 특정기초연구사업(R01-2002-000-00151-0) 지원으로 수행하였습니다.
** 한국과학기술원 수학과 암호론 연구실({snubogus, Jehong.Park, sghahn}@kaist.ac.kr)

신하는 방법에 의하여 보장되었다. 이러한 전방향 안전성을 만족하는 열쇠교환 프로토콜은 자연스럽게 수신자와 송신자가 공유된 열쇠를 이용하여 안전하게 메시지를 주고받은 후에 그것을 즉각적으로 물리적인 장치에서 지움으로써 전방향 안전성을 만족시키는 쌍방간의 상호작용이 필요한 암호 시스템(interactive encryption scheme)으로 확장될 수 있다. 이후에 Anderson은 열쇠교환 프로토콜의 전방향 안전성을 쌍방간의 상호작용이 필요 없는 암호 요소의 안전성 개념[2]으로 다음과 같이 확장하였다.

시스템의 총 유효한 시간 N 을 $0,1,\cdots,N-1$ 로 나눈다. 수신자는 처음시간의 비밀 열쇠를 초기화 하여 자신의 저장장치에 저장한 후, 시간의 흐름에 따라 비밀 열쇠를 새롭게 진화시킨다(key evolving). 즉, 수신자는 $i-1$ 시간의 비밀 열쇠에 일방향 함수를 이용하여 i 시간의 열쇠를 생성하고 즉시 $i-1$ 시간의 비밀 열쇠를 저장장치에서 삭제한다. 따라서 i 시간의 비밀 열쇠가 노출이 되더라도 그 전 시간 동안의 암호화된 정보의 안전성을 보장할 수 있다. 또한 비밀 열쇠에 대응되는 공개 열쇠는 유효한 전체 시간 동안 고정되므로 수신자와 송신자사이의 열쇠 교환을 위한 상호작용이 필요 없다.

이러한 개념이 소개된 이후 쌍방간의 상호작용 없이 전방향 안전성을 만족하는 서명 기법, 자기 인증 기법(identification scheme), 비밀 열쇠 암호 시스템 등의 여러 암호 요소들이 제안되었다.[1,3,4,8,11,13,15,16] 또한 최근 Katz[7]에 의해 처음으로 이를 만족하는 공개 열쇠 암호 시스템이 제안되었다. 이 시스템은 랜덤 오라클 모델이 아닌 표준 모델에서의 결정적 겹선형 Diffie-Hellman문제(decision bilinear Diffie-Hellman problem)가 어렵다는 가정 하에서 선택 암호문 공격에 대한 전방향 안전성을 제공한다. 또한 랜덤 오라클 모델에서는 계산적 Diffie-Hellman 문제가 어렵다는 가정 하에서 선택 암호문 공격에 대한 전방향 안전성을 제공한다. 이 시스템은 계층 사용자 ID기반 암호 시스템[9,12]에서 사용된 트리 구조를 변형한 이항 트리 구조(binary tree structure)를 사용하고, 초특이 곡선(supersingular curve)과 같은 특수한 군 구조에서 효율적인 사용이 가능한 겹선형 함수를 이용한 Boneh의 사용자 ID기반 암호 시스템[5]이 변형 적용되었다.

앞에서 언급한 바와 같이 암호 시스템과는 다르게 쌍방간의 상호 작용은 없는 동시에 전방향 안전성을 만족하는 서명 기법은 여러 차례 제안되었다. 그러나 랜덤 오라클 모델에서 계산적 Diffie-Hellman 문제의 어려움을 안전성의 기반으로 하는 서명 기법은 제안되지 않았다. 본 논문에서는 Boneh 등이 제안한 BLS 서명 기법[6]을 변형하여 Katz등의 암호 시스템[7]에 대응되는 서명 기법을 제안하고 랜덤 오라클 모델에서 선택 메세지 공격에 대한 전방향 안전성을 증명한다. 이는 최초의 계산적 Diffie-Hellman 문제를 기반으로 전방향 안전성을 가지는 서명 기법이다.

본 논문의 구성은 다음과 같다. 2장에서는 새로운 서명 기법에서 사용되는 겹선형 함수와 Gap Diffie Hellman 군의 정의들을 소개하고 관련 안전성 기반 문제들을 정리한다. 또한 열쇠 진화(key evolving) 서명 기법의 정의를 살펴보고 열쇠 진화 서명 기법이 전방향 안전성을 만족하는 조건을 알아본다. 그리고 3장에서는 본 논문에서 제안하는 전방향 안전성을 가지는 새로운 서명 기법을 소개하고 그 안전성을 증명한다. 마지막 4장에서 결론과 향후 연구방향에 대해 서술한다.

Ⅱ. 정의 및 기본개념

2.1 기본 개념

이 절에서는 앞으로 제안할 서명 기법의 안전성 기반 문제를 소개하고 겹선형 함수 및 Gap Diffie Hellman 군을 정의한다. 우선 다음과 같이 표시하자.

- G_1 : 위수가 소수 q인 순환 덧셈군
- G_2 : 위수가 소수 q인 순환 곱셈군
- $P \in G_1$: 군 G_1 의 생성원

이때 군 G_1에서 정의될 수 있는 일반적인 문제들은 다음과 같다.

(1) 계산적 Diffie-Hellman(CDH) 문제

군 G_1의 원소 P, aP, bP가 주어졌을 때 abP를 계산하는 것으로 이 문제를 해결하는 알고리즘 A의 이점(advantage)을 다음과 같이 정의한다.

$$AdvCDH_A = \Pr[A(P, aP, bP) = abP : a \leftarrow Z_p, h \leftarrow G_1].$$

여기서 a, h는 각각 Z_p, G_1에서 임의로 선택한다. 최대 t시간 실행 후 $AdvCDH_A \geq \varepsilon$이면 알고리즘 A가 (t, ε)으로 군 G_1의 CDH 문제를 해결한다고 정의

한다.

(2) 결정적 Diffie-Hellman(DDH) 문제

군 G_1의 원소 P, aP, bP, cP가 주어졌을 때 법 q로 $c \equiv ab$를 만족 하는지 여부를 판단하는 문제로 성립하는 경우의 (aP, bP, cP)을 Diffie-Hellman tuple이라 한다.

[정의 1] 함수 $e: G_1 \times G_1 \rightarrow G_2$는 계산이 효율적이며 군 G_1의 임의의 원소 P_1, P_2와 Q_1, Q_2에 대하여 다음 조건을 만족하면 겹선형(bilinear) 함수라 한다.

(1) [Bilinear]

$e(P_1 + P_2, Q_1) = e(P_1, Q_1) \cdot e(P_2, Q_1)$

$e(P_1, Q_1 + Q_2) = e(P_1, Q_1) \cdot e(P_1, Q_2)$

(2) [Non-degenerate]

$e(P, Q) \neq 1$인 $P, Q \in G_1$가 존재

잘 알려진 겹선형 함수의 예로 초특이 타원곡선에서의 Weil pairing 및 Tate pairing이 있으며 이를 이용하면 DDH 문제는 쉽고 CDH 문제는 해결하기 어려운 성질을 갖는 Gap Diffie-Hellman (GDH)군을 유도할 수 있다.

[정의 2] GDH 군 G에서 (t, ε)으로 군 G의 CDH 문제를 해결하는 알고리즘이 없으면 군 G를 (t, ε)-GDH 군이라 한다.

2.2 열쇠진화 서명 기법(Key Evolving Signature Scheme)

열쇠 진화 서명 기법은 일반적인 서명 기법에 주기적인 열쇠 갱신 알고리즘을 적용한 것이다. 그러므로 서명 기법의 전방향 안전성의 개념은 열쇠 진화 서명 기법의 구조를 기반으로 정의될 수 있다.

[정의 3] 열쇠 진화 서명 기법($KE.PKS$)은 다음과 같은 ($KE.GEN, KE.UPD, KE.SIGN, KE.VRFY$) 4개의 다항식 시간 알고리즘이다.

(1) 열쇠 생성 알고리즘 $KE.GEN$: 확률적 다항식 시간 알고리즘으로 안전성 인수 K와 전체 시간 N을 입력으로 하여 N시간 동안 사용되는 공개 열쇠인 PK와 그에 대응되는 최초의 비밀 열쇠 SK_0를 출력한다.

(2) 열쇠 진화 알고리즘 $KE.UPD$: 확률적 다항식 시간 알고리즘으로 입력값 SK_{i-1}에 대해 i시간의 비밀 열쇠인 SK_i를 출력한다.

(3) 서명 알고리즘 $KE.SIGN$: 확률적 다항식 시간 알고리즘으로 현재 i시간 비밀 열쇠 SK_i를 사용하여 메시지 M에 대한 i시간의 서명 $\langle i, \sigma \rangle$를 출력한다.

(4) 인증 알고리즘 $KE.VRFY$: 결정적 다항식 시간 알고리즘으로 공개 열쇠 PK를 입력 값으로 서명 $\langle i, \sigma \rangle$가 메시지 M에 대한 유효한 서명이면 1, 그렇지 않으면 0을 출력한다. 즉, $KE.VRFY(PK, M, \langle i, \sigma \rangle) = 1$을 만족하면 $\langle i, \sigma \rangle$는 메시지 M에 대한 i시간의 유효한 서명이다.

열쇠 진화 서명 기법이 전방향 안전성을 가진다는 것은 공격자가 현재의 비밀 열쇠를 얻어도 그 이전 시간의 비밀 열쇠로 생성된 서명을 위조하는 것이 계산적으로 불가능하다는 것을 의미한다. 이를 구체적으로 설명하면 다음과 같다. 랜덤 오라클 모델에서 공격자 A는 서명자의 공개 열쇠인 PK와 전체 시간 N, 현재 시간을 입력 값으로 다음과 같은 3단계에 걸친 공격을 수행한다. 기본적으로 랜덤 오라클 H_1, H_2의 출력 값을 요구할 수 있다. 첫째, 공격자 스스로 선택한 메시지 M에 대한 서명 값을 요구한다. 이를 선택 메시지 공격(chosen message attack) 단계라 한다. 이 단계에서 공격자는 보통의 서명 기법의 공격모델과 마찬가지로 현재 시간의 서명을 생성해주는 오라클에 접근할 수 있다. 각 시간에서 공격자는 계속해서 다음 시간에도 선택 메시지 공격을 수행할 것인지 아니면 Break-In 공격을 수행할 것인지를 결정한다. Break-In 공격 단계는 공격자가 자신이 선택한 i 시간의 비밀 열쇠를 요구하는 것을 의미한다. 위 두 가지 공격단계를 성공적으로 마친 후에 공격자가 $0 \leq j < i < N$ 인 j시간의 유효한 서명 $\langle j, \sigma \rangle$을 출력하면 공격자가 승리한 것이다. 이를 형식화한 Bellare 와 Miner[3]의 안전성 모델은 다음과 같다. 즉 아래와 같은 실험(experiment)에서 최종적으로 $d=1$ 값이 출력되는 확률이 공격자 A의 승리확률로 정의된다.

[실험 1] $\mathrm{EXP}^{FS.PKS}_{A}(KE.PKS(K, N), A)$

임의의 랜덤 오라클 H_1, H_2를 선택한다.

입력 : $(PK, SK_0) \leftarrow KE.GEN(K, N)$

$j \leftarrow 0$;

$Decision$ = Break-In 이거나 $j = N$ 일때까지

$Decision \leftarrow A^{KE.SIGN(SK_j, \cdot), H_2(\cdot)}(PK, j)$

$SK_{j+1} \leftarrow KE.UPD(SK_j, j+1)$;

$j \leftarrow j+1$;

를 반복한다.

$(M, (b, \sigma)) \leftarrow A^{H_2(\cdot)}(SK_j)$;

M이 서명 오라클 $KE.SIGN(SK_b, \cdot)$에 쿼리 되지 않았고 $\langle b, \sigma \rangle$가 M의 유효한 서명인 동시에 $0 \le b < j \le N$ 이면 $d=1$ 그렇지 않으면 $d=0$을 출력한다.

위 실험에서 $d=1$이 출력되는 확률을 다음과 같이 정의한다.

$$AdvKE.PKS_A = \Pr[\mathrm{EXP}\ FS.PKS_A = 1].$$

[정의 4] 위의 [실험 1]에서 최대 t시간 동안 최대 q_s개의 서명 쿼리와 q_{H_2}개의 해쉬 쿼리를 실행한 후의 알고리즘 A의 이점이 $AdvKE.PKS_A \ge \varepsilon$이면 A가 $(t, q_s, q_{H_2}, \varepsilon)$으로 열쇠 진화 서명 기법을 공격한다 라고 하고, 이와 같은 알고리즘이 존재하지 않을 때 열쇠 진화 서명 기법 $KE.PKS$이 랜덤 오라클 모델에서 선택 메세지 공격에 대하여 $(t, q_s, q_{H_2}, \varepsilon)$-전방향 안전하다고 정의하며 $KE.PKS(t, q_s, q_{H_2}, \varepsilon)$으로 표현한다.

당연한 결과로 위와 같이 정의된 $KE.PKS(t, q_s, q_{H_2}, \varepsilon)$ 서명 기법을 고정된 비밀 열쇠를 사용하는 일정한 시간 i 동안 선택 메세지 공격하는 알고리즘이 i 시간의 서명을 위조하는 것은 어렵다.

Ⅲ. 전방향 안전성을 가지는 새로운 서명 기법

3.1 기본 용어 및 구조

논문에서 제안되는 전방향 안전성을 가지는 새로운 서명 기법은 계층 ID 기반 서명 기법(hierarchical identity-based signature)[9,12]에서 사용된 트리 구조를 변형한 이항 트리 구조를 이용한다.

전체 깊이(depth)가 l인 이항 트리를 사용하는 경우 전체 시간은 $N = 2^{l+1} - 1$이며 각 시간은 0에서 $N-1$ 사이의 정수 값이다. 이 이항 트리에 잘 알려진 우선순위이동(pre-order traversal)을 이용하여 다음과 같은 방법으로 각 노드를 하나의 유효한 시간으로 대응시키며 각 노드는 고유한 비밀 열쇠를 갖는다. 시간 i에 대응되는 노드의 비트열을 w^i로 나타내고 대응되는 비밀 열쇠를 S_{w^i}라 하자. 우선순위이동에서 $w^0 = \varepsilon$는 뿌리 노드(root node)이고, w^i가 내부 노드(internal node) 일때 $w^{i+1} = w^i0$ 이다. w^i가 나뭇잎 노드(leaf node)이고 $i < N-1$ 일때 $w'0$가 w^i의 prefix 중에서 비트열의 길이가 가장 긴 노드라 하면 $w^{i+1} = w'1$로 놓는다. 그 외의 노드에 관련된 용어는 다음과 같다.

- $w^i = w_1 w_2 \cdots w_n$: i 시간에 대응되는 노드의 비트열
- w^i0는 w^i의 왼쪽 자식 노드(left child node)
- w^i1는 w^i의 오른쪽 자식 노드(right child node)
- $w^i|k$: 비트열 w^i의 w_1부터 w_k 비트까지 w^i의 k-prefix

(예) $w^i = 01001$이면 $w^i|2 = 01$, $w^i|5 = w^i = 01001$

- $w^i|k^*$: $w^i|k$ 와 마지막 비트만 다름 $w^i|k$ 의 형제 노드(sibling node)

(예) $w^i|2^* = 00$, $w^i|5^* = 01000$

본 논문에서 제안되는 서명 기법은 전체 N시간 동안 공개 열쇠가 PK로 고정된다. 그리고 i시간의 비밀 열쇠 SK_i는 노드 w^i에 대응된 비밀 열쇠인 S_{w^i}와 함께 뿌리 노드에서 노드 w^i까지 연결된 경로에 있는 모든 노드들의 오른쪽 형제 노드에 대응되는 비밀 열쇠들을 포함한다. 즉, i시간의 비밀 열쇠 SK_i는 마지막 비트가 0인 w^i의 prefix를 $w^i|k$라 할때 오른쪽 형제 노드인 $w^i|k^*$에 대응되는 비밀 열쇠 $S_{w^i|k^*}$를 포함한다. 간략한 표현을 위하여 $w^i|k$의 마지막 비트가 0이 아닌 경우의 $S_{w^i|k^*}$값을 0으로 놓자. SK_i는 $(S_{w^i|1^*}, S_{w^i|2^*}, \cdots, S_{w^i|n^*}, S_{w^i})$으로 나타낼 수 있다.

시간 i 동안 서명자(signer)는 노드 w^i에 대응된 비밀 열쇠 S_{w^i}를 사용하여 매세지에 대한 서명을 생성하게 되며 검증자(verifier)는 공개 열쇠 PK와 시간 i를 이용하여 주어진 서명을 검증한다. 한편, 열쇠 진화 알고리즘은 i 시간이 지나면 아래와 같은 방법으로 새로운 비밀 열쇠를 생성한다.

(1) w^i가 내부 노드이면

w^i의 왼쪽 자식 노드는 다음 시간에 사용될 w^{i+1}

$=w^i0$로 놓고 대응되는 비밀 열쇠 $S_{w^{i+1}}$를 생성한다. 또한 오른쪽 자식 노드 w^i1에 대응되는 비밀 열쇠 S_{w^i1}를 생성한다.

(2) w^i가 나뭇잎 노드이면

다음 시간에 사용될 노드 w^{i+1}의 비밀 열쇠 $S_{w^{i+1}}$는 이미 SK_i에 포함되어 있다.

(1)과 (2) 두 경우 모두 열쇠 갱신이 끝나는 즉시 노드 w^i의 비밀 열쇠를 저장 장치에서 삭제한다. 복잡한 것처럼 보이는 위의 열쇠 진화 알고리즘은 잘 알려진 자료구조 스택(stack)을 이용하여 손쉽게 구현될 수 있다. $ST-SK(=SK_i)$는 노드에 대응되는 비밀 열쇠들의 스택으로 i시간 동안 w^i에 대응되는 비밀 열쇠 S_{w^i}를 맨 위(top)에 가진다고 하자. 서명자는 S_{w^i}를 이용하여 i시간 동안의 서명을 생성한다. i시간이 끝나고 $i+1$시간의 비밀 열쇠를 생성할 때 이미 사용된 S_{w^i}는 $ST-SK$ 스택에서 꺼낸다(pop).

(1) w^i가 내부 노드이면

자식 노드인 w^i0와 w^i1에 대응되는 비밀 열쇠 S_{w^i0}, S_{w^i1}을 생성하여 S_{w^i1}과 S_{w^i0}을 순서대로 $ST-SK$ 스택에 넣는다(push). 그리고 S_{w^i0}을 $S_{w^{i+1}}$으로 사용한다.

(2) w^i가 나뭇잎 노드이면

맨 위에 놓인 비밀 열쇠가 $S_{w^{i+1}}$가 된다.

(1)과 (2) 두 경우 모두 스택에서 꺼낸 S_{w^i}를 저장장치에서 완전히 삭제한다.

3.2 구체적인 서명 기법

본 논문에서 제안하는 서명 기법 $FS.PKS$는 다음과 같은 4개의 요소로 이루어져 있으며 전방향 안전성을 만족한다.

3.2.1 $FS.PKS$ 서명 기법

(1) 시스템 파라미터 생성 알고리즘 $FS.GEN(K, l)$

① 인증기관(CA)은 안전성 파라미터 K와 이항 트리 구조의 깊이 l을 입력 값으로 시스템 파라미터인 위수가 소수 q인 두 개의 군 G_1, G_2과 겹선형 함수 $e: G_1\times G_1\rightarrow G_2$을 생성한다. 또한 임의의 생성원 $P\in G_1$과 임의의 $\alpha\in Z/qZ$를 선택하여 $Q=\alpha P$로 놓는다. 해쉬 함수 $H_1: \{0,1\}^*\rightarrow G_1$과 $H_2: \{0,1\}^*\times\{0,1\}^*\rightarrow G_1$을 선택한다.

② 뿌리 노드의 비밀 열쇠는 $SN_\varepsilon=\alpha H_1(\varepsilon)$이고 전체 시간 동안 공개 열쇠는 $PK=Q$이다. 그리고 시스템 파라미터는 $(G_1, G_2, e, P, Q, H_1, H_2)$이다

(2) 열쇠 갱신 알고리즘 $FS.UPD(PK, i, SK_i)$

i 시간에 대응되는 노드를 w라 하고 먼저 스택 $ST-SK$ 에서 S_w를 꺼내자. 트리에서 노드 w의 위치에 따라 다음과 같이 비밀 열쇠를 갱신한다.

① $w=w_1\cdots w_n$이 내부 노드이면 w노드에 대응되는 비밀 열쇠 $S_w=(R_{w1}, R_{w2}, \cdots, R_w, SN_w)$를 입력 값으로 자식 노드 $w0, w1$의 비밀 열쇠를 다음과 같이 출력한다.

㉠ 임의의 $\rho_{w0}, \rho_{w1}\in Z_q$ 선택

$R_{w0}=\rho_{w0}P$, $R_{w1}=\rho_{w1}P$와

$SN_{w0}=SN_w+\rho_{w0}H_1(w0)$, $SN_{w1}=SN_w+\rho_{w1}H_1(w1)$

을 계산하고

㉡ $S_{w1}=(R_{w1}, \cdots, R_w, R_{w1}, SN_{w1})$과

$S_{w0}=(R_{w1}, \cdots, R_w, R_{w0}, SN_{w0})$을 순서대로 스택 $ST-SK$ 에 넣은 후, S_w를 저장장치에서 삭제한다.

② $w=w_1\cdots w_n$이 나뭇잎 노드인 경우에는 S_w를 저장장치에서 삭제한다.

(3) 서명 생성 알고리즘 $FS.SIGN(i, SK_i, M)$

시간 i에 대응되는 노드를 $w=w_1\cdots w_n$라 하자. 서명자는 $ST-SK$의 맨 위에 있는 S_w를 사용하여 다음과 같이 서명한다.

① 임의의 $r\in Z_q$을 선택하여 $U=rP, P_M=H_2(M, i)$와 $FS=SN_w+rP_M$을 계산하고 $\sigma=(U, FS)$라 하자.

② 서명자는 $\langle M, i\rangle$에 대한 서명으로 $\langle i, \sigma=(U, FS)\rangle$과 $R_{w|m}$, $1\le m\le n$을 출력한다.

(4) 서명 검증 알고리즘 $FS.VRFY(PK, M, \langle i, \sigma\rangle)$

시간 i에 대응되는 노드를 $w=w_1\cdots w_n$라 하자. 검증자는 $P_M=H_2(M, i)$일때 다음 등식이 성립하면 $\langle i, \sigma=(U, FS)\rangle$를 $\langle M, i\rangle$에 대한 유효한 서명으로 확인한다.

$$e(P, FS)$$
$$= \prod_{m=1}^{n} e(R_{u|m}, H_1(u|m)) \cdot e(U, P_M) \cdot e(Q, H_1(\varepsilon))$$

다음과 같은 간단한 계산에 의해서 유효한 서명은 위 등식을 만족한다는 것을 알 수 있다.

$$\prod_{m=1}^{n} e(R_{u|m}, H_1(u|m)) \cdot e(U, P_M) \cdot e(Q, H_1(\varepsilon))$$
$$= \prod_{m=1}^{n} e(P, \rho_{u|m} H_1(u|m)) \cdot e(P, rP_M) \cdot e(P, aH_1(\varepsilon))$$
$$= e(P, \sum_{m=1}^{n} \rho_{u|m} H_1(u|m)) + rP_M + aH_1(\varepsilon))$$
$$= e(P, FS).$$

본 서명 기법은 이항 트리의 모든 노드들을 각 시간에 대응시켜 활용하므로 나뭇잎 노드들만을 이용하는 일반적인 트리 구조의 시스템에 비하여 열쇠 생성 및 열쇠 진화 알고리즘의 복잡도가 $O(\log N)$에서 $O(1)$로 향상된다.

3.2.2 안전성 증명

[정리 1] 파라미터 생성 알고리즘 $FS.GEN(K, l)$에서 생성된 군 G_1가 (t', ε')-GDH 군 이면 다음을 만족하는 모든 (t, ε)

$$t \le t' - c_{G_1}((q_{H_2} + q_s) \cdot (3q_s + 1)/(q_s + 1) + Tl + 5l + 2T + q_s + 7)$$
$$\varepsilon \ge e \cdot N \cdot (q_s + 1) \cdot \varepsilon'$$

에 대하여 $FS.PKS$는 $(t, q_s, q_{H_2}, \varepsilon)$-전방향 안전성을 가진다.

위 정리는 $FS.PKS$ 서명 기법의 안전성이 G_1에서의 CDH 문제의 어려움을 기반으로 함을 보여준다.

(증명)

(1) 모의 실험(Simulation)

A가 $(t, q_s, q_{H_2}, \varepsilon)$으로 $FS.PKS$를 공격하는 알고리즘이라 하자. 이 알고리즘을 서브루틴으로 사용하여 t'시간동안 ε'이상의 확률로 G_1에서의 CDH 문제를 해결하는 알고리즘 B를 아래와 같이 설계한다. B는 군 G_1의 원소 P, aP, bP을 입력 값으로 abP를 출력하는 것을 목표로 A에게 [실험 1]과 구별되지 않는 모의 환경을 아래와 같이 제공한다. A는 자신의 쿼리에 대한 답이 B에 의해 생성된 것인지 실험의 도전자(challenger)에 의한 것인지 구별할 수 없다. 또한 B는 고정된 시간 $0 \le i \le N-1$ 동안 선택 메시지 공격을 수행하는 알고리즘 A'에게 아래의 [2단계]와 같은 방법으로 모의 환경을 제공할 수 있으며 A'가 시간 i에 대한 서명을 위조하는 경우에 이것을 이용하여 abP 값을 계산할 수 있다. 즉 시간 i의 서명 기법 $FS.SIGN(i, ST-SK, M)$은 시간 i에 대한 선택 메시지 공격에 안전하다.

[1단계] A의 파라미터 생성

B는 유효한 전체 시간 N과 임의의 시간 T, $0 \le T \le N-1$를 선택한다. T시간에 대응되는 노드의 비트열을 w^T라 하자. 군 G_1의 생성원인 P, 공개 열쇠 $Q = aP$ 그리고 N을 A가 공격하는 열쇠 진화 서명 기법의 시스템 파라미터로 제공하고 A 를 실행시킨다.

[2단계] 선택 메시지 공격(Chosen message attack)

B는 A가 선택 메시지 공격 단계에서 요구하는 H_2-쿼리 및 서명 쿼리들에 대하여 다음과 같이 답을 돌려준다.

각 시간마다 A는 다음 시간에도 선택 메시지 공격을 계속하고자 하는 경우엔 *Decision* 값을 0으로 선택하고, 다음 단계의 공격을 수행하고자 하는 경우, *Decision*값을 Break-In으로 선택하여 출력한다.

B는 j를 0으로 초기화하고 $H_1(\varepsilon) = bP = I_\varepsilon$으로 설정한다. 그리고 A가 *Decision* 값을 Break-In으로 출력하거나 j가 N이 될 때까지 아래의 ①-④를 순서대로 진행한다.

$w^j = w_1 \cdots w_t$를 j시간에 대응되는 노드라고 하자.

① H_2-쿼리 : A가 $H_2(\cdot)$ 해쉬 값을 요구할 때 새롭게 설정되는 값들의 리스트(H_2-리스트)를 다음과 같이 생성한다.

㉠ $\langle M_{j_k}, j \rangle$ 가 이미 쿼리 된 경우

저장된 H_2-리스트에서 $\langle M_{j_k}, j, U_{j_k}, h_{j_k}, \lambda_{j_k}, \varphi_{j_k}, c_{j_k} \rangle$ 값을 찾아 $H_2(M_{j_k}, j) = h_{j_k} \in G_1$으로 출력한다.

㉡ $\langle M_{j_k}, j \rangle$ 가 쿼리 되지 않은 경우

- $\Pr[c_{j_k} = 0] = 1/(q_s + 1)$인 임의의 $c_{j_k} \in \{0, 1\}$선택
- 임의의 $\lambda_{j_k}, \varphi_{j_k} \in Z_q$ 선택하고

만약 $c_{j_k} = 0$ 이면

$h_{j_k} \leftarrow \lambda_{j_k} P$; $U_{j_k} \leftarrow *$; $\varphi_{j_k} \leftarrow *$;

그렇지 않으면

$h_{j_k} \leftarrow \lambda_{j_k} P - (1/\varphi_{j_k}) I_\varepsilon$; $U_{j_k} \leftarrow \varphi_{j_k} Q$;

으로 설정한다.

- $\langle M_{j_k}, j, U_{j_k}, h_{j_k}, \lambda_{j_k}, \varphi_{j_k}, c_{j_k} \rangle$을 H_2-리스트에 추가한다.

② 서명 쿼리 : A가 $\langle M_{j_k}, j \rangle$에 대한 서명을 요구할 때 우선 ①에서와 같이 H_2-리스트를 생성한다. 대응되는 원소가 $\langle M_{j_k}, j, U_{j_k}, h_{j_k}, \lambda_{j_k}, \varphi_{j_k}, c_{j_k} \rangle$ 일때

㉠ $c_{j_k} = 0$이면 "실패" 선언하고 중단한다.

㉡ $c_{j_k} = 1$이면 현재 시간 j 이전의 열쇠 갱신 단계 ③에서 생성되어 저장된 $y_{w'|i}$와 $R_{w'|i}$, $1 \le i \le t$을 사용하여 $FS_{j_k} = \sum_{i=1}^{t} y_{w'|i} R_{w'|i} + \varphi_{j_k} \lambda_{j_k} Q$ 값을 계산하고 $\langle M_{j_k}, j \rangle$에 대한 서명으로 $\langle j, \sigma_{j_k} = (U_{j_k}, FS_{j_k}) \rangle$와 $R_{w'|i}$, $1 \le i \le t$을 출력한다.

- 완전성(Completeness) - 유효한 서명을 생성한다.

$$FS_{j_k} = aI_\varepsilon + \sum_{i=1}^{t} y_{w'|i} R_{w'|i} + \varphi_{j_k} a H_2(M_{j_k}, j)$$

$$= aI_\varepsilon + \sum_{i=1}^{t} y_{w'|i} R_{w'|i} + \varphi_{j_k} a(\lambda_{j_k} P - (1/\varphi_{j_k}) I_\varepsilon)$$

$$= \sum_{i=1}^{t} y_{w'|i} R_{w'|i} + \varphi_{j_k} \lambda_{j_k} Q \text{ 이 성립하므로}$$

B는 $aI_\varepsilon = abP$ 값을 계산하지 못하여도 $\langle M_{j_k}, j \rangle$에 대한 유효한 서명을 생성할 수 있다.

③ 열쇠 갱신

- w^j가 내부 노드인 경우만 아래와 같이 열쇠를 갱신한다.

$w^j \neq w^T$ 일때

$w^j 0$가 w^T의 prefix 이면

임의의 $y_{w'0}, \beta_{w'1}, \gamma_{w'1} \in Z_q$ 선택

$H_1(w^j 0) = y_{w'0} P$

$H_1(w^j 1) = \beta_{w'1} P - (1/\gamma_{w'1}) I_\varepsilon$

$R_{w'0} \in G_1$ 는 임의로 선택

$R_{w'1} = \gamma_{w'1} Q = (\gamma_{w'1} a) P \in G_1$계산

$w^j 0$가 w^T의 prefix 가 아닌 경우

임의의 $y_{w'0}, y_{w'1} \in Z_q$ 선택

$H_1(w^j 0) = y_{w'0} P$, $H_1(w^j 1) = y_{w'1} P$

$R_{w'0}, R_{w'1} \in G_1$ 을 임의로 선택

$w^j = w^T$ 이면

$\beta_{w'0}, \beta_{w'1}, \gamma_{w'0}, \gamma_{w'1} \in Z_q$ 선택

$H_1(w^j 0) = \beta_{w'0} P - (1/\gamma_{w'0}) I_\varepsilon$

$H_1(w^j 1) = \beta_{w'1} P - (1/\gamma_{w'1}) I_\varepsilon$

$R_{w'0} = \gamma_{w'0} Q = (\gamma_{w'0} a) P \in G_1$

$R_{w'1} = \gamma_{w'1} Q = (\gamma_{w'1} a) P \in G_1$

위와 같이 설정된 값들을 저장한다.

④ A 가 *Decision* 값을 출력하였을 때

㉠ $j < T$ 이고 $Decision = 0$ 이면

$j \leftarrow j+1$ 로 놓고 ①로 간다.

㉡ $j = T$ 이고 $Decision =$ Break-In 이면 [3단계]로 진행한다.

㉢ 그 외의 경우에는

"실패" 선언하고 알고리즘 B는 중단한다.

[3단계] 비밀 열쇠 노출 요구 공격(Break-In Attack)

A가 $j+1$ 시간의 비밀 열쇠 SK_{j+1}를 요구한다.

① $w^j|i$ 를 마지막 비트가 0인 w^j의 prefix라 하자. 이때 노드 $w^j|i^*$의 비밀 열쇠 $S_{w^j|i\cdot}$를 [2단계]-③에서 해쉬 $H_1(\cdot)$값을 정의할 때 사용된 $y_{w^j|m}$, $R_{w^j|m}$와 $\beta_{w^j|i\cdot}, \gamma_{w^j|i\cdot}$을 이용하여 다음과 같이 계산한다.

$$SN_{w^j|i\cdot} = \beta_{w^j|i\cdot} \cdot \gamma_{w^j|i\cdot} \, Q + \sum_{m=1}^{i-1} y_{w^j|m} R_{w^j|m} \ .$$

- 완전성(Completeness) - 유효한 비밀 열쇠를 생성한다.

[2단계]-③에서 $R_{w^j|i\cdot} = \gamma_{w^j|i\cdot} Q = (\gamma_{w^j|i\cdot} a) P$으로 정의되었으므로 $1 \le i \le t$ 에 대하여

$$SN_{w^j|i\cdot} = aI_\varepsilon + \sum_{m=1}^{i-1} y_{w^j|m} R_{w^j|m} + \gamma_{w^j|i\cdot} \cdot a H_1(w^j|i^*)$$

$$= aI_\varepsilon + \sum_{m=1}^{i-1} y_{w^j|m} R_{w^j|m} + \gamma_{w^j|i\cdot} \cdot a(\beta_{w^j|i\cdot} P - (1/\gamma_{w^j|i\cdot}) I_\varepsilon)$$

$$= \beta_{w^j|i\cdot} \cdot \gamma_{w^j|i\cdot} Q + \sum_{m=1}^{i-1} y_{w^j|m} R_{w^j|m} \text{ 이 성립한다.}$$

즉, B는 $aI_\varepsilon = abP$ 값을 계산하지 못하여도 유효한 비밀 열쇠 $S_{w^j|i\cdot}$를 생성할 수 있다.

② 비밀 열쇠 $S_{w^j b}$ ($b \in \{0,1\}$)도 마찬가지로 [2단계]-③ 에서 해쉬 $H_1(\cdot)$ 값을 정의할 때 사용된 $y_{w^j|m}$, $R_{w^j|m}$와 $\beta_{w^j b}$, $\gamma_{w^j b}$을 이용하여 다음과 같이 계산한다.

$$SN_{w^j b} = \beta_{w^j b} \cdot \gamma_{w^j b} Q + \sum_{m=1}^{t} y_{w^j|m} R_{w^j|m} \ .$$

- 완전성(Completeness) - 유효한 비밀 열쇠를 생성한다.

[2단계]-③에서 $R_{w^ib}=\gamma_{w^ib}Q=(\gamma_{w^ib}a)P$으로 정의되었으므로

$$SN_{w^ib}=aI_\varepsilon+\sum_{m=1}^{i}y_{w^i\|m}R_{w^i\|m}+\gamma_{w^ib}\cdot aH_1(w^ib)$$

$$=aI_\varepsilon+\sum_{m=1}^{i}y_{w^i\|m}R_{w^i\|m}+\gamma_{w^ib}\cdot a(\beta_{w^ib}P-(1/\gamma_{w^ib})I_\varepsilon)$$

$$=\beta_{w^ib}\cdot\gamma_{w^ib}Q+\sum_{m=1}^{i}y_{w^i\|m}R_{w^i\|m}$$ 이 성립한다.

즉, B는 $aI_\varepsilon=abP$ 값을 계산하지 못하여도 유효한 비밀 열쇠 S_{w^ib}를 생성할 수 있다.

B는 위 과정의 계산을 끝내고 $j+1$ 시간 비밀 열쇠 SK_{j+1}를 A의 비밀 열쇠 노출 공격에 대한 답으로 돌려준다.

[4단계] 서명 위조 단계

[1단계]-[3단계]의 공격과정을 성공적으로 수행한 후 A는 $\langle M,i\rangle$, $0\le i\le T$의 서명 $\langle i,\sigma=(U=xP,FS)\rangle$을 출력한다. i 시간에 대응되는 노드를 $w^i=w_1w_2\cdots w_n$라 하자. A가 출력한 서명이 유효하면

$FS=abP+\sum_{m=1}^{n}y_{w^i\|m}R_{w^i\|m}+xH_2(M,i)$을 만족한다.

[5단계] CDH 문제 해결

B는 A가 출력한 서명 $\langle i,\sigma=(U=xP,FS)\rangle$을 이용하여 자신의 목표인 abP 값을 다음과 같이 계산한다.

H_2-리스트에서 $\langle M,i,*,h,\lambda,\varphi,c\rangle$ 을 찾는다.

$c=0$ 이면

[2단계]-③에서 설정된 $y_{w^i\|m}$, $R_{w^i\|m}$와 λ를 이용하여 $abP=FS-\sum_{m=1}^{n}y_{w^i\|m}R_{w^i\|m}-\lambda U$를 계산한다.

$c=1$ 이면 "실패" 선언, 중단한다.

(2) 분석

위와 같은 모의 실험의 다섯 단계를 성공적으로 마친 후 아래의 4 가지 사건들이 모두 일어나는 경우에만 B는 성공적으로 abP 값을 출력할 수 있다.

<사건 $E1$> [2 단계]-②의 서명 쿼리 단계에서 "실패" 하지 않는다.

<사건 $E2$> [2 단계]-④ 에서 "실패" 하지 않는다. 즉, $Decision=$Break-In이고 $j=T$이다.

<사건 $E3$> [4 단계] 에서 A가 $\langle M,i\rangle$, $0\le i\le T$에 대한 서명 $\langle i,\sigma\rangle$을 위조한다.

<사건 $E4$> $E3$가 일어나고 A가 서명을 위조한 메시지 $\langle M,i\rangle$에 대한 H_2-리스트 c값이 0이다.

B의 성공확률 $\varepsilon'=\Pr[E1\wedge E2\wedge E3\wedge E4]$은 계산에 의해 $\Pr[E1]\cdot\Pr[E2|E1]\cdot\Pr[E3|E1\wedge E2]\cdot\Pr[E4|E1\wedge E2\wedge E3]$와 같으므로 각각의 확률을 다음과 같이 예측함으로써 B 의 성공확률을 계산할 수 있다.

① $\Pr[E1]\ge 1/e$

A가 i번째 메시지에 대한 서명 쿼리를 요구할 때 그전의 $i-1$개의 메시지에 대한 서명 쿼리가 유효했으므로 그 c값들에 대한 정보를 얻을 수도 있다. 그러나 새로운 i번째 메시지에 대한 서명 쿼리를 요구하기 전에는 이 메시지의 H_2-리스트 c값이 B에 의하여 0 혹은 1 둘 중 어느 것으로 정해져 있는지 구별할 수 없다. 따라서 B가 서명 쿼리 단계에서 실패하지 않았다는 것은 A의 서명 쿼리 q_s개의 H_2-리스트 c값이 모두 1로 설정되었다는 것을 의미한다.

$\Pr[c=1]=(1-1/(q_s+1))$이므로

$\Pr[E1]\ge(1-1/(q_s+1))^{q_s}\ge 1/e$ 이다.

② $\Pr[E2|E1]\ge 1/N$

A가 모의 실험 환경을 실제 [실험 1]과 구별할 수 없으므로 B가 정한 시간 T와 A의 Break-In 공격 시간이 일치할 확률은 $1/N$ 이상이다.

③ $\Pr[E3|E1\wedge E2]\ge\varepsilon$

B에 의한 모의 실험이 성공적으로 실행되면 즉, $E1$과 $E2$가 일어나면 A는 자신의 공격 이점인 ε 이상의 확률로 본 논문에서 제안한 서명 기법의 유효한 서명을 위조한다.

④ $\Pr[E4|E1\wedge E2\wedge E3]\ge 1/(q_s+1)$

사건 $E1, E2, E3$가 일어나고 최종적으로 A가 출력한 위조 서명의 H_2-리스트에서 c값이 1이면 B는 실패를 선언한다. A는 이전의 서명 쿼리들의 c값에 대한 정보를 얻을 수 있더라도 $\langle M,i\rangle$에 대한 서명 쿼리는 이루어지지 않았으므로 H_2-리스트에 $\langle M,i\rangle$의 c값이 무엇으로 설정되었는지 구별하지 못한다. 따라서

$\Pr[c=0|E1\wedge E2\wedge E3]\ge 1/(q_s+1)$이 성립한다.

위의 ①-④ 확률들을 종합하면 알고리즘 B가 성공적으로 abP를 계산하는 확률이 $\varepsilon/(eN(q_s+1))$ 이상임을 알 수 있다.

한편, 알고리즘 B의 실행 시간은 A의 실행시간에 다음의 실행 시간들을 모두 더한 값이다. c_{G_1}은 $a \in Z_q$에 대하여 $aP \in G_1$를 계산하는 즉 G_1에서의 상수배 연산에 소요되는 시간이라 하자. 여기서는 Z_q에서 곱셈 및 나눗셈 연산 시간은 고려하지 않을 것이다.

① H_2-쿼리

1개의 해쉬 쿼리에 대하여 H_2-리스트 값을 생성하는데 $1/(q_s+1)+q_s/(q_s+1)\times 3=(3q_s+1)/(q_s+1)$ 번의 상수배 연산이 평균적으로 필요하다. 따라서 기대되는 소요 시간은 $c_{G_1}\cdot(q_{H_2}+q_s)\cdot(3q_s+1)/(q_s+1)$ 이다.

② 서명 쿼리 및 열쇠 갱신

㉠ T시간 동안 각 시간마다 $\sum_{i=1}^{l} y_{w|i} R_{w|i}$ 값 계산을 위하여 최대 l 번의 상수배 연산이 필요하다. 따라서 최대 $c_{G_1}\cdot T\cdot l$ 시간이 소요된다.

㉡ 각 시간에서의 서명은 ㉠에서 계산된 $\sum_{i=1}^{l} y_{w|i} R_{w|i}$ 값에 $\varphi_{j_s}\lambda_{j_k} Q$을 더하여 생성되므로 총 q_s개의 서명 생성을 위하여 $c_{G_1}\cdot q_s$ 시간이 소요된다.

㉢ T시간 동안 열쇠 갱신 단계에서 필요한 상수배 연산의 개수는 최대 $4l+2T+6$ 번이다. 따라서 총 $c_{G_1}\cdot(4l+2T+6)$ 시간이 소요된다.

③ 비밀 열쇠 노출 단계

w^T의 prefix 노드 $w^{\tau}|i$ 들의 오른쪽 형제 노드의 비밀 열쇠 계산을 위하여 $\beta_{w^{\tau}|i}\cdot\ \cdot\gamma_{w^{\tau}|i}\cdot Q$ 값의 계산이 필요하므로 최대 l개의 상수배 연산이 필요하다. 따라서 최대 소요시간은 $c_{G_1}\cdot l$ 이다.

④ CDH 문제 해결단계에서 B는 abP 값 계산을 위하여 λU 값을 계산하므로 c_{G_1} 시간이 소요된다.

B의 실행 시간은 t와 ①-④ 시간의 총 합인 $t+c_{G_1}((q_{H_2}+q_s)\cdot(3q_s+1)/(q_s+1)+Tl+5l+2T+q_s+7)$ 이하이다.

Ⅳ. 결 론

암호 시스템, 서명 기법 및 암호 요소들이 전방향 안전성을 만족한다는 것은 현재의 비밀 열쇠가 노출된다 할지라도 이전의 비밀 열쇠를 이용하여 암호화된 정보들은 안전하다는 것을 의미한다. 이러한 전방향 안전성의 개념은 최근 활발한 서비스가 제공되고 있는 모바일 폰 및 스마트카드 등 보안이 취약한 장치들에 탑재되는 암호 시스템 및 암호 요소들이 갖추어야 할 필수적인 조건으로 각광받고 있다. 이런 결과로 많은 수의 전방향 안전성을 가지는 시스템들이 제안되었으나 이들은 몇 개의 제한된 수학적 문제들의 어려움을 기반으로 한다.[3,11,15,8] 최근에는 암호 시스템 및 암호 요소들에 있어 전방향 안전성은 기본적인 요건이 되고 있으므로 좀 더 다양한 기반 문제들에 대해 전방향 안전성을 만족하는 시스템들이 개발되어야 할 것이다.

본 논문에서는 기존에 제안된 전방향 안전성을 만족하는 서명 기법들과는 다르게, GDH 군의 성질과 이항 트리[14]를 이용하는 새로운 서명 기법을 소개하고 그 안전성을 증명하였다. 제안된 기법은 깊이가 l 인 트리를 사용하여 지수승에 해당하는 $2^{l+1}-1$ 시간 동안의 서명을 생성한다. 또한 트리의 나뭇잎 노드뿐 아니라 모든 노드들을 각 시간에 대응시켜 활용함으로써 기존의 이항 트리 구조 시스템의 열쇠 생성 및 열쇠 진화 복잡도인 $O(\log N)$을 $O(1)$으로 향상시켰다. 한편 제안된 서명 기법은 Katz 등의 암호 시스템[7]에 대응되는 서명 기법으로 동일한 시스템 파라미터를 공유하므로 하나의 암호 패키지로 구현될 경우 효과적인 활용을 기대할 수 있을 것이다.

참고로, 최근에 본 논문과 독립적으로 Hu 등에 의하여 Katz 등이 제안한 암호 시스템에 대응되는 서명 기법[10]이 소개 되었다. 본 논문에서 제안된 서명 기법은 시간 i 동안에 동일 메시지에 대하여 각각 다른 서명을 생성하는 임의적 서명 기법(randomized signature scheme)인 반면에 Hu 등의 서명 기법은 동일 시간, 동일 메시지에 대하여 고유한 서명을 생성하는 결정적 서명 기법(determinstic signature scheme)이다.

참 고 문 헌

[1] M. Abdalla and L. Reyzin, "A new forward-secure

digital signature scheme", *Advances in Cryptology-ASIACRYPT 2000*, LNCS 1976, pp.116~129, 2000.

[2] R. Anderson, "Two remarks on public key cryptology", *ACM-CCS*, 1997.

[3] M. Bellare and S. K. Miner, "A forward-secure digital signature scheme", *Advances in Cryptology-CRYPTO 1999*, LNCS 1666, pp.431~448, 1999.

[4] M. Bellare and B. Yee, "Forward security in private-key cryptography", *CT-RSA 2003*, LNCS 2612, pp.1~18, 2003.

[5] D. Boneh and M. Franklin, "Identity based encryption from the Weil pairing", *Advances in Cryptology - CRYPTO 2001*, LNCS 2139, pp.213~229, 2001.

[6] D. Boneh, B. Lynn and H. Shacham. "Short signatures from the Weil pairing", *Advances in Cryptology - ASIACRYPT 2001*, LNCS 2248, pp.514~532, 2001.

[7] R. Canetti, S. Halevi and J. Katz, "A forward-secure public-key encryption scheme", *EUROCRYPT 2003*, LNCS 2656, pp.255~271, 2003.

[8] Y. Dodis, J. Katz, S. Xu and M. Yung, "Key-insulated public Key cryptosystems", *Advances in Cryptology - EUROCRYPT 2002*, LNCS 2332, pp.65~82, 2002.

[9] C. Gentry and A. Silverberg, "Hierarchical ID-based cryptography", *Advances in Cryptology-ASIACRYPT 2001*, LNCS 2501, pp.548~566, 2002.

[10] F. Hu, C.-H. Wu and J. D. Irwin, "A new forward secure signature scheme using Bilinear Maps", http://eprint.iacr.org/2003/188, 2003.

[11] G. Itkis and L. Reyzin, "Forward-secure signatures with optimal signing and verifying", *Advances in Cryptology-CRYPTO 2001*, LNCS 2139, pp.332~354, 2001.

[12] T. G. Kim, D. H. Yum and P. J. Lee, "보다 효율적인 Hierarchical ID-based cryptosystem", *정보보호학회 논문지 13권 3호*, pp.127~133, 2003.

[13] H. Krawczyk, "Simple forward-secure signatures from any signature scheme", *ACM-CCS 2000*, pp. 108-115, 2000.

[14] S. W. Lee, J. H. Cheon and Y. D. Kim, "Pairing을 이용한 트리 기반 그룹키 합의 프로토콜", *정보보호학회 논문지 13권 3호*, pp.101~110, 2003.

[15] C.-F. Lu and S.W. Shieh, "Secure key-evolving protocols for discrete logarithm schemes", *CT-RSA 2002*, LNCS 2271, pp.300~309, 2002.

[16] T. Malkin, D. Micciancio and S. K. Miner, "Efficient generic forward-secure signatures with an unbounded number of time periods", *Advances in Cryptology - EUROCRYPT 2002*, LNCS 2332, pp. 400~417, 2002.

〈著者紹介〉

강 보 경 (Bo Gyeong Kang) 학생회원
1999년 8월 : 서울대학교 수학교육학과 졸업
2001년 8월 : 한국과학기술원 수학과 석사 졸업
2001년 9월~현재 : 한국과학기술원 수학과 박사과정
<관심분야> 암호학, 정수론, Complexity Theory

박 제 홍 (Je Hong Park) 학생회원
1998년 2월 : 경북대학교 수학과 졸업
2000년 2월 : 한국과학기술원 수학과 석사 졸업
2000년 3월~현재 : 한국과학기술원 수학과 박사과정
<관심분야> 암호학, 타원곡선

한 상 근 (Sang Geun Hahn) 종신회원
1979년 : 서울대학교 수학과 졸업
1982년 : 뉴멕시코 주립대 석사 졸업
1987년 : 오하이오 주립대 박사 졸업
1987년 ~ 현재 : 한국과학기술원 수학과 교수
<관심분야> 암호학, 타원곡선, 정수론