키 전개 방식 전방보안 서명기법

최철준, 김광조

한국정보통신대학원대학교

Key Evolving Forward Secure Signature Scheme

Chul-Joon Choi, Kwangjo Kim

Information & Communications Univ.

요 약

공개키 기반 구조에서 개인키의 노출은 심각한 문제를 일으킬 수 있다. 현재의 전자서명 기법에서는 서명자가 자신의 개인키 도난에 대한 사실을 인지하지 못하면, 공격자는 원하는 만큼의 전자서명을 생성 할 수 있다. 따라서 서명자가 개인키를 노출하였을 때 서명자를 보호 할 수 있는 전자서명 기법이 필요하다. 본 논문에서는 Pairing을 이용한 키 전개방식 전방보안 전자서명기법을 제시하고, 이를 Schnorr 서명기법에 적용한다. 제안하는 서명기법은 기존의 전자서명기법과 호환됨으로써 매우 실용적이다.

1. 서론

본 논문에서는 전방보안의 개념이 적용된 전자서명기법을 제안한다. 이 기법을 적용하면 서명자가 키를 노출하였을 때 발생할 수 있는 피해를 최소화 할 수 있다.

1.1 키 노출에 의한 문제점

공개키 시스템을 사용하는데 있어서, 사람들은 자신의 키를 잃어버릴 수 있다. 특히, 키의 노출이나 도난과 같은 사건이 일어난 후 사용자가 인지하지 못하는 경우가 많다. 현재의 전자서명 기법에서, 사용자가 개인키를 노출하고 나서 노출한 사실을 인지하지 못한다면, 공격자는 어떠한 제약도 받지 않고 원하는 만큼의 위조된 서명을 생성할 수 있다. 따라서 서명자가 개인키를 노출하였을 때 서명자를 보호 할 수 있는 강력한 전자서명 기법이 필요하다. 그러나 키 노출이나 도난에 강한 실용적인 전자서명 기법을 구현하는 것은 어려운 문제이다. 키 노출에 따른 문제를 해결하기 위한 가장 잘 알려진 방법이 비밀 분산기법이다. 이 방법은 서명자가 보관해야 하는 비밀 정보를 분산하여 보관함으로써 비밀정보 노출 위험을 줄이는 것이다. 그러나 이러한 방법은 비밀정보의 분산이나 사용에 있어서 복잡한 프로세스가 요구된다.

1.2 관련연구

키 노출에 따른 피해를 최소화하기 위해 전방보안(Forward Secrecy)의 개념이 Anderson[1]에 의해 처음 제안 되었다. 이후 Bellare와 Miner[2]는 처음으로 실용적인 전방보안 전자서명기법을 제안하였다. Bellare와 Miner는 [2]에서 전체 기간을 T주기로 나누고, 각 주기에 새로운 개인키를 사용하는 키 전개 서명기법을 사용하였다. 각 주기의 개인키는 현재 주기에서 서명을 생성하는데 사용되고, 다음주기의 개인키를 생성하는데 사용된다. 이때 공개키는 일반적인 서명기법과 같이 전체 기간동안 변하지 않고 유지된다. 검증 알고리즘은 서명의 유효성뿐만 아니라 해당주기의 유효성도 확인한다. 이 서명기법은 공격자가 현 주기의 개인키를 획득하더라도 과거의 서명을 위조할 수 없게 함으로써 전방보안을 만족한다. [2]가 제안된 이후 많은 연구결과가 보고 되었다[3][4][5][6][7][8].

Krawczyk[3]는 어떤 서명 기법에도 적용 가능한 전방보안 서명기법을 제안하였다. 이 방식은 공개키와 개인키 쌍들을 미리 생성하고 인증서를 추가하여 공개키를 인증서에 추가하는 방식을 사용하였다. Itkis 와 Reyzin[4]은 Guillou-Quisquater 서명기법을 기반으로 한 전방보안 서명기법을 제안하였다. 이 방식은 서명과 검증알고리즘의 효율성을 높였으나 키 생성과 갱신알고리즘의 계산량이 매우 많다. 전방보안 문턱서명기법을 Abdalla, Miner와 Mimprempre가 [9]에서 제안하였다. 또한 전방보안 공개키 암호화기법이 Canetti, Halevi, 그리고 Katz에 의해 [10]에서 제안되었다. 쌍일차함수를 이용한 전방보안 서명 기법은 [7]에서 제안되었다. 이 방식은 [10]의 방식을 응용하여, 이진 트리 구조를 이용한 키 갱신 방법을 사용하였다.

1.3 암호학적 가정

제안하는 키 전개방식 전자서명기법은 Boneh와 Franklin[11]에 의해 정형화된 BDH(Bilinear Diffie-Hellman)문제를 기반으로 한다. 특히 GDH(Gap Diffie-Hellman) 문제를 이용하여 서명자의 현 주기 공개키를 인증한다. BDH 및 GDH에 대한 자세한 내용은 각각 [11]와 [12]을 참조 하도록 권장한다. 또한 본 논문에서는 제안 알고리즘이 랜덤 오라클 모델에서 동작하는 것으로 가정한다. 이때 제안 알고리즘은 안전성 분석에서 랜덤하다고 알려진 공개된 해쉬함수 H_1과 H_2를 오라클로 사용하게 된다.

1.4 논문에서 제시한 결과

많은 연구자들에 의해 전방보안을 만족하는 전자서명기법이 제시되었다. 그러나 이러한 전자서명기법은 기존의 서명기법과 호환되지 않는다는 문제점을 안고 있다. 본 논문에서는 Schnorr 서명기법과 같은 이산로그의 어려운 문제에 기반 한 서명기법과 호환될 수 있는 전방보안 전자서명기법을 제안한다. 이 기법은 쌍일차 Diffie-Hellman문제(BDH)를 기반하며, Gap Diffie-Hellman(GDH) 문제를 이용한다. 본 논문에서는 키 사용기간을 T주기로 나누고, 각 주기마다 개인키를 갱신하여 전방보안을 만족할 수 있도록 한다. 전방보안을 보장하기 위한 방법으로 키 생성 단계에서 마스터키 x를 생성한 후, x를 제곱하여 첫 번째 주기의 키를 생성하고(x^2P), x^2을 제곱하여 두 번째 주기의 키를 생성(x^4P)하는 방식으로 T주기의 개인키와 공개키 쌍을 미리 생성한다. 키 쌍이 모두 생성되면 마스터키 x는 메모리에서 지우고, 생성된 키 쌍을 보관한다. 각 주기에 사용되는 공개키 정보를 서명값과 함께 검증자에게 전달하고, 검증자는 서명검증 전에 해당주기의 공개키를 GDH 문제를 이용하여 검증한 후 서명값를 검증하게 한다.

2. 전방보안 전자서명기법

키 전개방식 전자서명기법과, 전방보안(Forward Security)의 개념에 대하여 정의한다. 또한 전방보안을 공격하는 공격자 모델을 정의한다.

2.1 표기

본 논문에서 사용하는 T는 키 갱신 주기의 총 횟수를 의미한다. M은 메시지를 나타내며, SK_i는 i 번째 주기의 키 쌍이다. SK_i는 i 번째 주기 개인키 S_i와 i 번째 주기의 공개키 $\mathcal{F}_i$로 구성된다. S_i는 서명 시에 사용되는 개인키이다. $\mathcal{F}_i$는 i 번째 주기에 사용되는 공개키로, Q_1부터 Q_i 까지 모든 Q들로 구성된다. 또 이 값은 서명자가 서명값과 함께 검증자에게 전달한다. x는 마스터키에 해당되는 값으로 1과 q사이에서 임으로 선택한 수이다. 또한 본 논문에서는 해쉬함수 $H_1 : \{0, 1\}^* \rightarrow G^*_1$과 $H_2 : G_2 \rightarrow \{0, 1\}^n$를 정의하여 사용한다.

2.2 정의

이 장에서는 전방보안을 보장하는 키 전개방식 전자서명기법을 정의한다. 키 전개방식 전자서명 알고리즘을 제안하고, 이 기법이 갖는 전방보안의 의미를 설명한다.

[정의 1] *키 전개방식 전자서명기법* **FSS=(Gen, Upd, Sign, Ver)**는 다음과 같은 4가지 형태를 갖는 PPT(Probabilistic Polynomial Time)알고리즘이다.

- 키 생성 알고리즘 **Gen**은 입력으로 1^k(k는 보안상수)와 T(키 갱신 총 주기)을 받아 공개키 PK와 개인키 SK를 출력하는 PPT 알고리즘이다.
- 키 갱신 알고리즘 **Upd**는 이전 주기의 개인키 SK_i와 인덱스 $i+1$를 입력으로 받아 현 주기의 개인키 SK_{i+1}을 출력하는 PPT 알고리즘이다.
- 서명생성알고리즘 **Sign**은 메시지 M과 SK_{i+1}을 입력값으로 받아 서명 σ를 생성하는 PPT 알고리즘이다.
- 서명검증알고리즘 **Ver**은 메시지 M, 서명값 σ와 공개키를 입력받아 1(유효) 또는 0(무효)을 출력하는 DPT(Deterministic Polynomial Time) 알고리즘이다.

제안하는 키 전개방식 전자서명기법의 완전성(Correctness)은 다음과 같이 정의한다.

[정의 2] FSS 서명 기법의 완전성

$\forall(SK, PK)\in Gen$, $\forall SK_{i+1}=(S_{i+1}, \mathcal{F}_{i+1})\in Upd$, $\forall M$, 그리고 $\forall \sigma\in Sign(M, SK_{i+1})$에 대하여 $Ver(M, \sigma, \mathcal{F}_{i+1}, Q_0)=1$을 출력한다면, FSS 서명기법은 완전하다고 한다.

위에서 정의한 알고리즘은 다음과 같이 동작한다. 서명자가 키 생성알고리즘을 이용하여 키 쌍(PK, SK)를 생성한다. 이때 공개키 PK는 일반적인 전자서명 알고리즘과 같이 등록되거나 인증서에 첨부된다. 서명자는 개인키 SK를 저장한다. 첫 주기가 시작되는 시점에서 서명자는 키 갱신알고리즘을 이용하여 개인키를 갱신한다($SK_1 \leftarrow Upd(SK_0, 1)$). 이때 이전 키 SK_0는 메모리에서 삭제한다. 첫 번째 주기 동안 서명자는 SK_1을 이용하여 서명을 하게 된다. 두 번째 주기가 시작되면 SK_2를 생성하고 SK_1은 메모리에서 삭제하다. 이러한 방식으로 T주기가 끝날 때까지 계속 개인키 갱신을 실시한다. 특히 각 주기에 사용되는 공개키는 키 생성단계에서 미리 생성되어 보관되며, 서명 생성 시 서명값과 함께 검증자에게 전달된다. 전달된 각 주기의 공개키는 마스터 공개키 PK를 이용하여 검증된다. 마스터 공개키 PK는 전체 주기 동안 변하지 않고 남아 있다.

제안하는 키 전개방식 전자서명기법의 전방보안(Forward Security)에 대한 정의는 [2]에서 정의한 것을 인용하여 사용한다.

제안 알고리즘의 전방위 보안을 공격하는 공격자는 다음과 같이 분류한다.

- 공격자 A : 공격자 A는 공개된 오라클 H_1과 H_2에 접근하여 쿼리 할 수 있다. 또한 공격자 A는 공개키 PK와 총 주기 T 그리고 현 주기 j를 알고 있다. 또한 공격자 A는 각 주기에 만들어진 공개정보 $\mathcal{F}_i$를 알 수 있다. 이때 공격자 A는 키 전개방식 전자서명기법을 공격하는 공격자라 한다.
- 공격자 B : 공격자 B는 좀더 강력한 능력을 갖고 있다. 공격자 B는 공격자 A가 갖는 모든 능력을 갖고 있으며, 현 주기 j에서 서명자의 개인키 SK_j를 획득할 수 있다. 이때 공격자 B를 키 전개방식 전자서명기법의 전방보안을 공격하는 공격자라고 한다.

공격자 A가 개인키 SK_j에 대한 정보를 알지 않고도 위조된 유효한 서명을 생성할 수 있다면, 공격자 A는 위조에 성공했다고 말한다. 이는 공격자 A가 키 전개방식 전자서명의 위조에 성공한 것으로 간주한다. 또한 공격자 B가 현 주기의 개인키 SK_j를 이용하여 이전 주기의 개인키를 복원하거나, 이전주기의 위조된 유효한 서명을 생성할 수 있다면 공격자 B는 키 전개방식 전자서명기법의 전방보안 특성을 깨뜨리는데 성공한 것으로 한다.

안전성 분석의 정형화를 위해서 공격자 B는 3단계 공격기능을 갖는 것으로 판단한다: 선택적 메시지 공격단계(CMA); Break-in 단계(breakin); 위조단계(froge).

[정의 3] (전방보안) FSS=(Gen, Upd, Sign, Ver)이 키 전개방식 전자서명기법이라고 하고, B를 키 전개방식 전자서명 기법의 전방보안을 공격하는 공격자라고 하자. 그리고 Adv_B^{FSS}를 공격자 B가 FSS를 위조할 확률이라고 하면 다음과 같이 표시 할 수 있다.

$$Adv_B^{FSS} = Pr\left[Ver(M, j, \sigma'_{j+1}, h'_{j+1}, \mathcal{F}_j, PK) = 1 \;\middle|\; \begin{array}{c} (q, n, G_1, G_2, i) \leftarrow \mathcal{G}(1^k) \\ P \in G_1 \\ (\sigma'_{j+1}, h'_{j+1}) \leftarrow \mathcal{A}(\cdot) \\ M \ni \mathcal{O}_{Sign(\cdot)} \\ \langle T, j, Q_0, \mathcal{F}_j, H_1, H_2 \rangle \\ \langle SK_j \rangle \end{array} \right]$$

Adv_B^{FSS}는 공격자 B가 현 주기(j)의 개인키 시스템에 침입하여 SK_j를 획득하여, i 번째 주기의 위조된 서명 σ_i를 생성했을 때 σ_i가 유효한 것으로 판명될 확률로 표시할 수 있다.

3. 제안 서명기법

새로운 키 전개 전자서명 기법을 제안한다. 제안하는 서명 기법은 전방보안의 특성을 만족한다. 제안하는 키 전개 서명 기법을 Schnorr서명기법에 적용함으로써 Schnorr 서명기법이 전방보안을 만족한다.

3.1 제안 알고리즘

제안하는 FSS 알고리즘은 Gen(1^k, T), Upd(SK_i,i+1), Sign(M, SK_{i+1}), Ver(M, σ, $\mathcal{F}_{i+1}$, Q_0)의 4개 알고리즘으로 구성되어 있다. 본 논문에서는 k를 Gen(1^k, T) 알고리즘에 주어지는 보안상수라고 하고, G를 BDH 매개변수 생성기라고 한다.

Gen(1^k, T) : 주어진 보안상수 k에 대하여 키 생성 알고리즘은 다음과 같이 동작한다.

1. 입력값 k를 갖는 G를 실행하여, 큰 소수 q를 생성하고, 위수가 q인 두 그룹 G_1 ,G_2와 쌍일차함수 $e : G_1\times G_1\rightarrow G_2$를 생성한다.

2. 해쉬함수 $H_1 : \{0, 1\}^* \rightarrow G^*_1$과 $H_2 : G_2 \rightarrow \{0, 1\}^n$를 선택한다. 안전성 분석 시에 H_1과 H_2는 랜덤오라클로 간주한다.

3. 임의의 생성자 $P \in G_1$를 선택하고, 난수 $x \in Z^*_q$를 선택하고, $Q_0=xP$, $S_0=xH_1(ID)$를 계산한다. T 주기 동안 사용될 키 쌍들의 집합 $SK=\{SK_i \mid 0 \le i \le T\}$는 다음과 같이 구한다.

   - $\alpha_i = x^{2^i} \bmod q$, $Q_i = \alpha_i P$, $S_i = \alpha_i H_1(ID)$
   $SK_i=(S_i, \mathcal{F}_i)$로 한다. 이때 $\mathcal{F}_i=(Q_1, Q_2, \ldots, Q_i)$이다.

4. 공개키는 $PK=(q, n, G_1, G_2, \hat{e}, P, Q_0, H_1, H_2)$이다.

5. 개인키는 $SK=\{SK_i \mid 0 \le i \le T\}$이다.

Upd(SK_i, i+1) : 키 갱신 알고리즘($i < T\{1$)

1. 이전 주기의 키 쌍 $SK_i=(S_i, \mathcal{F}_i)$를 삭제하고, 다음 주기의 키 쌍 $SK_{i+1}=(S_{i+1}, \mathcal{F}_{i+1})$을 미리 생성한 개인키 집합 SK에서 선택한다. 이 때 $\mathcal{F}_{i+1}=\mathcal{F}_i \cup \{Q_{i+1}\}$이다.

2. 갱신된 i+1번째 키 $SK_{i+1}=(S_{i+1}, \mathcal{F}_{i+1})$을 출력한다.

Sign(M, SK_{i+1}) : 서명알고리즘 Sign(M SK_{i+1})은 메시지 M과 SK_{i+1}을 입력값으로 받아 $\sigma=Sign(M, SK_{i+1})$를 계산한다. 이때 서명값은 $(\sigma, \mathcal{F}_{i+1})$이다.

Ver(M, σ, $\mathcal{F}_{i+1}$, Q_0) : 검증알고리즘 Ver(M, σ, $\mathcal{F}_{i+1}$, Q_0)는 메시지 M과 서명값 σ 그리고 공개키를 입력으로 받아 유효성을 검사한다. 이때 Ver(M, σ, $\mathcal{F}_{i+1}$, Q_0)=1이면 유효한 서명값이 된다.

3.2 알고리즘의 동작

제안하는 FSS 알고리즘은 키 생성 알고리즘 **Gen(1^k, T)**를 동작하여 키 쌍 (PK, SK)를 생성하면서 시작된다. 이때 개인키에 해당하는 난수 값 x는 마스터키와 같은 역할을 하게 된다. 키 생성과 함께 첫 주기가 시작되면 서명자는 Upd(SK_i, i+1)을 이용하여 SK_1(=Upd(SK_0, 1))을 생성한다. 키 갱신 알고리즘 Upd(SK_0, 1)는 SK를 메모리에서 삭제하고, 키 생성단계에서 미리 만들어진 값 $SK_1=(S_1, \mathcal{F}_1)$을 출력한다. 서명에 사용하는 키 값 S_1은 키 생성 단계에서 마스터 키 값을 제곱하여 만들어 진다($S_1=x^2H_1(ID)$). 또한 현 주기의 공개키로 사용될 값 $\mathcal{F}_1$은 $Q_1(=x^2P)$으로 구성된다. 서명알고리즘 Sign(M, SK_1)를 이용하여 만들어진 서명값은 $(\sigma, \mathcal{F}_1)$가 된다. 검증자는 서명검증 전에 현주기의 공개키 $\mathcal{F}_1$가 유효한 값인지 검사한다. 검증자는 $\mathcal{F}_1$가 유효한 값인지 검사하기 위해서 공개키 Q_0를 획득한 후 쌍일차함수 $\hat{e}$를 이용하여 검사한다. 만약 $\hat{e}(Q_1, P) \neq \hat{e}(Q_0, Q_0)$이면 첫 번째 주기 공개키 Q_1이 위조된 값이므로 서명을 유효하지 않은 것으로 판단한다. 이때 Q_1은 x^2을 알지 못하면 구할 수 없는 값이므로, 위의 등식이 성립하면 서명자의 현 주기 공개키가 인증된다. 즉 x^2을 알지 못하고 Q_0를 알고 있는 공격자는 Q_1을 계산 할 수 없다. 이는 GDH 그룹에서 계산적 Diffie-Hellman 문제는 어렵고, 결정적 Diffie- Hellman 문제는 쉽다는 가정을 사용하고 있기 때문이다. 서명자의 임시 공개키가 인증되면, 검증자는 서명검증 알고리즘을 이용하여 서명을 검증하게 된다.

3.3 Schnorr 서명기법에의 적용

제안하는 키 전개 전자서명기법을 Schnorr 기반 서명기법[13]에 적용함으로써 Schnorr기반 서명기법이 전방보안을 만족하게 한다.

키 생성 알고리즘 : Gen(1^k, T)

1. 입력값 k를 갖는 G를 실행하여, 큰 소수 q를 생성하고, 위수가 q인 두 그룹 G_1, G_2와 쌍일차함수 $\hat{e} : G_1 \times G_1 \rightarrow G_2$를 생성한다.

2. 해쉬함수 $H_1 : \{0, 1\}^* \rightarrow G^*_1$, $H_2 : G_2 \rightarrow \{0, 1\}^n$를 선택한다.

3. 임의의 생성자 $P \in G_1$를 선택하고, 난수 $x \in Z^*_q$를 선택하고, $Q_0=xP$, $S_0=xH_1(ID)$를 계산한다. T 주기 동안 사용될 키 쌍들의 집합 $SK=\{SK_i \mid 0 \le i \le T\}$는 다음과 같이 구한다.

   - $\alpha_i = x^{2^i} \bmod q$, $Q_i = \alpha_i P$, $S_i = \alpha_i H_1(ID)$
   $SK_i=(S_i, \mathcal{F}_i)$로 한다. 이때 $\mathcal{F}_0=(Q_1, Q_2, \ldots, Q_i)$이다.

4. 공개키는 $PK=(q, n, G_1, G_2, \hat{e}, P, Q_0, H_1, H_2)$이다.

5. 개인키는 $SK=\{SK_i \mid 0 \le i \le T\}$이다.

키 갱신 알고리즘 : Upd(SK_i, i+1)

1. 이전 주기의 키 쌍 $SK_i=(S_i, \mathcal{F}_i)$를 삭제하고, 다음 주기의 키 쌍 $SK_{i+1}=(S_{i+1}, \mathcal{F}_{i+1})$을 미리

생성한 개인키 집합 SK에서 선택한다. 이 때 $\mathcal{F}_{i+1}=\mathcal{F}_i \cup \{Q_{i+1}\}$이다.

2. 갱신된 i+1번째 키 쌍 $SK_{i+1}=(S_{i+1}, \mathcal{F}_{i+1})$을 출력한다.

서명생성 알고리즘 : Sign(M, SK_{i+1})

1. 메시지 M에 서명하기 위해서 난수 $r \in Z^*_q$를 선택하고, $u=\hat{e}(rP, P)$를 계산한다.
2. $h=H_2(M, u)$를 계산한다.
3. $\sigma=hS_{i+1}+rP$을 계산한다.
4. 서명 쌍은 $(\sigma, h, \mathcal{F}_{i+1})$이다.

서명검증 알고리즘 : Ver(M, σ, $\mathcal{F}_{i+1}$, Q_0)

1. 주어진 $(M, \sigma, h, \mathcal{F}_{i+1}, Q_0)$에 대하여 다음을 실행한다.

   For I = 1 to i

   if $\hat{e}(Q_{I+1}, P) \neq \hat{e}(Q_I, Q_I)$

   then reject

2. 1의 검증이 유효하면

   $u' = \frac{\hat{e}(\sigma, P)}{(\hat{e}(Q_{i+1}, H_1(ID)))^h}$ 를 계산하고,

   $h'=H_2(M, u')$을 계산한다.

3. h = h'이면 1을 출력하고, 그렇지 않으면 0을 출력한다.

4. 안전성 분석

제안하는 서명기법의 안전성은 랜덤 오라클 H_1, H_2가 존재한다는 가정 하에 앞에서 제시한 모델을 이용해서 평가할 수 있다. 제안하는 서명기법의 전방보안을 공격하는 공격자 B에 대해서, 성공 확률을 측정하는 실험 B-Forge(FSS, B)를 진행하여 공격자 B의 성공확률로 변환한다. 전방보안의 관점에서 공격자 B가 FSS를 깨뜨리는데 성공할 확률 즉 실험 B-Forge(FSS, B)가 1을 출력할 확률은 다음과 같이 나타낼 수 있다.

$$Adv_B^{FSS} = Pr[\ \text{B-Forge(FSS, B)=1}\ ]$$

다음은 키 전개방식 전방보안 서명기법에 대한 공격에서 공격자를 운영하는 실험 B-Forge(FSS, B)를 나타낸다.

실험 B-Forge(FSS, B)

Pick k bit random integer $c \in Z^*_q$.
$d \leftarrow c^2 \bmod q$, $S_1 \leftarrow cP$, $S_2 \leftarrow dP$,
Repeat t times
  $\{c', S'\} \leftarrow B(S_2)$
  If $c = c'$ or $S_1 = S'$ then return 1
  else return 0

본 논문에서는 [11][12]에 의거하여 BDH 및 GDH 문제가 어렵다고 가정하였다. 따라서 알고리즘 B가 1을 출력하는데 걸리는 시간 t가 계산적 Diffie-Hellman 문제를 푸는 가장 잘 알려진 알고리즘의 시간보다 크다고 하면 B-Forge(FSS, B)가 1을 출력할 확률은 매우 작은 값이 된다. 따라서 계산적 Diffie-Hellman 문제가 계산적으로 불가능하다고 가정하면 제안하는 서명기법은 전방보안을 만족한다. 안전성 분석에 대한 자세한 기술은 추후 과제로 남기고 생략한다.

5. 결론

키 전개방식 전방보안 전자서명기법을 제안한다. 제안 기법을 Schnorr 기반 서명기법에 적용함으로써 GDH 그룹 내에서 Schnorr 서명기법이 전방보안을 만족하게 한다. 제안하는 키 전개방식 전방보안 전자서명기법은 이산로그의 어려움을 기반으로 하는 서명기법 및 암호화 기법에 모두 적용할 수 있다. 향후 연구과제로는 제안 서명기법의 안전성을 증명하고, 대리서명기법과 은닉서명기법에 응용하려고 한다.

참고문헌

[1] R. Anderson, "Cryptography and Security Policy," Invited Talk of ACM-CCS'97, 1997.

[2] M. Bellare and S. Miner, "A forward-secure digital signature scheme" In Proc. of CRYPTO'99, LNCS vol. 1666, pp.431-448. 1999.

[3] H. Krawczyk, "Simple forward secure signatures for any signature scheme" In Proc. of ACMCCS'00, pp.108-115. 2000.

[4] G. Itkis and L. Reyzin, "Forward Secure Signatures with optimal signing and verifying" In Proc. of CRYPTO'01, LNCS vol. 2139, pp.332-354. 2001.

[5] M. Abdalla and L. Reyzin, "A new forward-secure digital signature scheme" In Proc. of ASIACRYPT'00, LNCS vol. 1976, pp.116-129. 2000.

[6] T. Maklin, D. Micciancio and S. Miner, "Efficient generic forward-secure signatures with an unbounded number of time periods" In Proc. of Eurocrypt'02, LNCS vol. 2332, pp.400-417. 2002.

[7] F. Hu, C. Wu and J. Irwin, "A new forward-secure signature scheme using bilinear maps" In e-Pring Achieve 188, 2003.

[8] A. Kozlov and L. Reyzin, "Forward-secure signatures with fast key update" In Proc. of CSCN'02, LNCS vol. 2579, pp.247-262. 2002.

[9] M Abdalla, S. Miner, and C. Namprempre, "Forward-secure threshold signature schemes" In Proc. of CT-RSA'01, LNCS vol. 2020, pp.441-456. 2001.

[10] R. Canetti, S. Halevi, and J. Katz, "A forward secure public key encryption scheme" In Proc. of Eurocrypt'03, LNCS vol. 2656, pp.255-271. 2003.

[11] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing" SIAM Journal on Computing, vol. 32, no 3, pp.586-615. 2003.

[12] J. Cha and J Cheon, "An identity-based signature from gap Diffie-Hellman groups" In Proc. of PKC'03, LNCS vol. 2139, pp.18-39. 2003.

[13] F. Hess, "Efficient identity based signature schemes based on pairings" In Proc. of SAC'02, LNCS vol. 2595, pp.310-324. 2002.