비밀분산법을 이용한 익명성 보장 핑거프린팅 기법

용승림○ 이상호
이화여자대학교
{dragon○, shlee}@ewha.ac.kr

Anonymous Fingerprinting Using Secret Sharing Scheme

Seung-Lim Yong○ Sang-Ho Lee
Dept. of Computer Science and Engineering, Ewha Womans University

요 약

핑거프린팅 기법은 암호학적인 기법들을 이용하여 디지털 데이터를 불법적으로 재배포한 사용자를 찾아냄으로써 디지털 데이터의 저작권을 보호한다. 익명성이 보장되는 핑거프린팅 기법은 대칭적인 기법과 달리 사용자만이 핑거프린트가 삽입된 데이터를 알 수 있고 비대칭 기법과 달리 데이터가 재배포되기 전에는 사용자의 익명성이 보장되는 기법이다. 본 논문에서는 사용자의 익명성이 보장되는 새로운 핑거프린팅 기법을 제안한다. 비밀분산법을 이용하여 사용자의 신원정보를 숨겨두었다가 재배포 발견시 조정자에게 누가 재배포하였는지 증거를 제출함으로써 재배포자의 배포 사실을 증명한다. 또한 등록시에 Schnorr 서명을 이용하여 고발된 사용자가 범행을 부인할 수 없도록 한다. 제안된 방법은 이산대수문제와 양자간 안전한 계산과정의 안전성에 근거하여 사용자의 익명성이 보장된다.

1. 서 론

인터넷과 같은 컴퓨터 망과 컴퓨터 이용의 급격한 발달로 전자상거래가 활발해지고 디지털 데이터의 확산 및 보급이 일반화되고 있다. 그러나, 이러한 데이터들은 디지털이라는 속성으로 인하여 누구나 손쉽게 불법적인 복제를 통해서 이들을 획득할 수 있게 되고, 이 때문에 저작권 문제가 야기되고 있다. 따라서 정보기반 전자 상거래에서 디지털 데이터의 저작권 보호는 아주 중요한 문제가 되었다.

핑거프린팅 기법은 디지털 데이터의 저작권을 보호하기 위해 데이터의 복사 자체를 막는 암호학적인 기법이 아니라 암호학적인 기법들을 이용하여 디지털 데이터를 불법적으로 재분배한 사람을 찾아내는 기법이다[4]. 저작권 보호에 관한 규칙을 어기고 데이터를 불법적으로 분배하는 사람을 재배포자(traitor)라 한다. 핑거프린팅 기법은 데이터가 불법적으로 재배포 되었을 때 상점이 그 데이터를 구매한 재배포자를 식별할 수 있게끔 함으로써 디지털 저작권을 보호한다.

핑거프린팅 기법은 대칭적인 기법과 비대칭적인 기법, 익명성이 보장되는 기법의 세 가지 부류로 나뉜다. 대칭적인 기법은 상점이 서로 다른 데이터의 복사본들을 각 사용자에게 나누어주고 데이터가 재배포되었을 경우 그 복사본이 어떤 사람에게 나누어준 것인지를 찾아내어 재배포자를 찾아내는 방법이다[1,2]. 반면 비대칭적인 기법은 사용자만이 핑거프린트가 삽입된 데이터를 알 수 있고 상점은 핑거프린트된 데이터를 알 수 없는 기법이다. 그러나 상점이 정직한 사용자의 신원도 알아낼 수 있다는 단점이 있다[5]. 익명성이 보장되는 핑거프린팅 기법은 데이터가 불법적으로 재배포되기 전까지 상점은 핑거프린트된 데이터를 모를 뿐 아니라 사용자의 익명성도 철저히 보장되는 기법이다. 만약 재분배된 데이터가 발견되면 상점은 등록센터의 도움을 받고, 핑거프린트된 데이터에서 사용자 정보를 추출하여 조정자(arbiter)에게 재배포자의 유죄를 입증할 증거로 제시하여 재배포자의 신원을 확인할 수 있다[6].

본 논문에서는 비밀분산법과 Schnorr의 전자서명 기법을 이용하여 지수계산량을 줄이면서 재배포자의 신원을 확인할 수 있는 효율적인 익명의 핑거프린팅 기법을 제안한다.

2. 관련 연구

2.1 비밀분산법

비밀분산법이란 비밀정보를 다수의 조각으로 분할하여 다수에게 공유시킴으로써 원 정보를 보다 안전하게 유지·관리하는 암호 프로토콜 중의 하나이다. 비밀정보 D는 비밀정보 복원 권한을 가진 각 참가자들의 비밀조각으로부터 다시 복원될 수 있다. 비밀정보 D에 대해서 분할된 다수의 조각을 비밀조각이라고 하고, 비밀정보를 생성하고 분배하는 사람을 분배자라고 하며, 생성된 비밀조각을 가지고 비밀정보 복원에 참여하는 사람을 참가자라고 한다. 대표적인 비밀분산법으로 Shamir에 의해 제안된 (t,n)-임계치 비밀분산법이 있다[8]. Shamir의 (t,n)-임계치 비밀분산법은 다음과 같다.

분배자는 비밀 정보 D를 상수항 $a_0=D$ 로 하는 임의의 $t-1$ 차 다항식 $q(x)=a_0+a_1x+...a_{t-1}x^{t-1}$ 을 생성하고 D_i ($D_1=q(1),...,D_i=q(i),...,D_n=q(n)$)의 비밀조각으로 나눈다. 비밀조각들 중에서 임의의 t개의 값이 주어지면 $q(x)$의 계수의 값들을 보간법을 이용하여 구할 수 있고, 다항식에 0을 대입함으로써 비밀정보 $D=q(0)$ 를 복원할 수 있다. 그러나 비밀조작이 $t-1$ 개만 주어졌을 때는 비밀정보 D를 복원할 수 없다.

2.2 Schnorr의 전자서명 기법

Schnorr의 전자서명 기법의 안전성은 이산대수 문제의 풀기 어려움에 근거한다. Schnorr 서명을 이용하는 사용자는 임의 난수 g와 두 개의 소수 p,q를 나누어 갖는다. 비밀키, 공개키 쌍을 생성하기 위하여 사용자는 하나의 임의난수 $s(0<s<q$를 비밀키로 생성하고, 공개키 $v=g^{-s} \bmod q$ 를 생성한다.

메시지 m에 서명을 하기 위하여 임의의 수 r ($r \in Z_q$)을 선택하고 다음과 같은 계산을 수행한다.

$$x=g^r \bmod p \ , \ c=h(m\|x) \ , \ y=(r+se) \bmod q$$

함수 $h()$는 충돌 회피 일방향 해쉬함수이다. 메시지 m에 대한 서명은 (e,y) 쌍이 된다. 서명을 확인하기 위하여 $x=g^y v^e \bmod p$ 이고 c와 $h(m\|x)$가 같은지 확인한다. 확인이 되면 서명은 정당한 것이다[7].

r 값은 다른 메시지에 서명을 생성하기 위하여 여러 번 사용되어서는 안되며 반드시 한번만 서명에 사용되어야 한다. 만약 하나의 값 r을 두 개의 서로 다른 메시지 m과 m'에 서명을 위하여 사용하면, 두 개의 서명 (c,y) 와 (c',y') 이 생성된다. 그러면 두 개의 값을 이용하여 비밀값 s를 다음과 같은 식을 이용하여 얻어낼 수 있다.

$$s=\left(\frac{y-y'}{c-c'}\right) \equiv \left(\frac{(r+sc)-(r+sc')}{c-c'}\right) \bmod q$$

3. 비밀분산법을 이용한 익명의 핑거프린팅 기법

핑거프린팅 기법에 참여하는 주체는 사용자, 상점, 등록센터, 그리고 조정자(arbiter)이다. 사용자는 상점으로부터 데이터를 구매하기 전에 등록센터에 등록을 한다. 사용자가 이용하는 Schnorr 서명의 공개키는 공개되어 있다고 가정한다. 본 논문에서 제안하는 핑거프린팅 기법은 등록, 핑거프린팅, 신원확인 및 심리 프로토콜로 구성되어 있다.

3.1 등록 프로토콜

1) 사용자는 Schnorr 서명기법을 이용하기 위하여 임의난수 $s(0<s<q$를 비밀키로, $v=g^{-s} \bmod q$ 를 공개키로 생성하여 공개키는 공개한다.
2) 사용자는 임의난수 r,x 를 선택하여 $r'=g^r$, $x'=g^x$ 을 계산하고 아이디 U를 이용하여 $h_1=g^U$ 을 계산한다. 또한 계산한 h_1,x',r' 의 값을 이용하여 $e=h(h(h_1\|x')\|r')$ 를 계산하고 $y=r+se$ 을 계산하여 Shnorr의 전자서명 $<e,y>$을 생성한다. $<e,y>$와 h_1,x'을 등록센터로 보낸다.
3) 등록센터는 사용자로부터 받은 서명값 $<e,y>$와 h_1,x'를 이용하여 서명이 올바른지 확인한다. 사용자의 공개키 v를 이용하여 $z=g^y v^e \bmod p$ 의 값을 계산하고 사용자로부터 받는 서명값에 대입하여 서명값과 $h(h(h_1\|x')\|z)$값이 같으면 서명이 확인되는 것이다.
4) 사용자가 보낸 서명을 확인하고 사용자에게 도전값 a를 보낸다.
5) 사용자는 $b=x-Ua$ 를 계산하고 응답값 b를 등록센터로 보낸다.
6) 등록센터는 a,b , 받아둔 x'을 이용하여 $x'=g^b h_1^a$ 값이 맞는지 확인하고, 값이 맞는 경우 $Cert=Sig(h(h_1\|x'))$ 와 같이 서명을 하여 사용자에게 보낸다.

3.2 핑거프린팅 프로토콜

1) 상점은 사용자에게 도전값 a'을 보낸다.
2) 사용자는 $b'=x-Ua'$ 값을 계산하고 $b''=g^{b'}$ 을 계산하여 $h(b'')$과 $Cert$값을 상점에게 보낸다.
3) 상점과 사용자는 양자간의 안전한 계산과정(secure two-party computation)을 수행한다[3]. 사용자는 h_1과 b'을 입력값으로, 상점은 사용자에게서 받은 $Cert$ 값과 b'' 값 그리고 사용자가 구매를 원하는 원래의 디지털 데이터 $item$을 입력값으로 입력한다. 그리고 다음의 계산과정을 수행한다.
   (a) $ver_1=Verify_1(b',h(b''))$. b'의 값을 검증하기 위하여 b''의 값을 이용하여 검증을 수행한다. ver_1은 부울 변수로서 상점만이 그 값을 볼 수 있으며 $b''=g^{b'}$ 인 경우에만 참이 된다.
   (b) $ver_2=Verify_2(a',b',h_1,Cert)$. 먼저 $x''=g^{b'}h_1^{a'}$ 값을 계산한다. 계산한 x''값을 서명값에 대입하여 $Cert$ 값과 $h(h_1\|x'')$의 값이 같으면 검증되는 것이다.
   (c) $item'=FING(item,emb)$. $FING()$은 원래의 디지털 데이터에 핑거프린트를 삽입하는 알고리즘이다. 알고리즘의 수행으로 원래의 데이터 $item$값에 $emb=(a',b')$ 가 삽입된 $item'$이 결과값으로 얻어지며 이 값은 사용자만이 볼 수 있는 값이 된다.

위의 양자간의 안전한 계산과정 모두는 상점이 먼저 결과값을 얻어내어 ver_1과 ver_2에 대한 값이 참으로 검증이 된 후에 사용자가 $item'$을 얻을 수 있도록 수행된다.

3.3 신원확인 프로토콜

1) 만약 $item'$이 재배포된 것이 발견되면 상점은 $item'$으로부터 핑거프린트를 추출해 내는 알고리즘을 이용하여 emb를 추출해 낸다.
2) 추출해낸 데이터는 (a',b') 값이 되고 이를 이용하여 거래내역에서 $Cert$값을 추출한다. 등록센터에 $Cert$값을 보낸다.
3) 등록센터는 등록 데이터베이스로부터 $Cert$에 해당하는 값 ($a,b,<e,y>$)의 값을 찾아내어 상점에게 보낸다.
4) 상점은 $Cert$값을 이용하여 Schnorr 서명 $<e,y>$값을 확인한다.

3.4 심리 프로토콜

이제 상점은 조정자에게 디지털 데이터 $item$이 재배포되었음을 확인시켜야 한다. 조정자는 상점과의 다음의 프로토콜을

통하여 고발된 사용자가 디지털 데이터를 재배포 하였음을 확인한다.

1) 상점은 조정자에게 증거가 되는 값을 보낸다.
$proof=((a',b'),(a,b,<e,y>))$

2) 조정자는 $(a,b),(a',b')$ 값을 이용하여 사용자의 아이디 U를 찾아낸다.

$$U=\frac{b-b'}{a'-a}$$

3) U를 이용하여 $h_1=g^U$ 와 $x'=g^{(b+U/a)}$ 을 계산한다. 사용자의 Schnorr 공개키 v와 $proof$ 안의 값 $<e,y>$를 이용하여 $z'=g^y v^e$ 을 계산하고 계산한 값들을 이용하여 $e=h(h(h_1 \| x') \| z)$ 의 값이 맞는지 검증한다. 만약 서명값이 맞는 경우 사용자는 디지털 데이터를 재배포한 것이 되고, 그렇지 않을 경우 사용자는 무죄가 되며 상점이나 등록센터가 책임을 지게 된다.

4. 결과 및 분석

본 논문에서는 상점은 핑거프린트를 삽입하는 삽입 기법이 공모공격에 안전하고 재배포된 데이터로부터 증거값을 찾을 수 있다는 안전성을 가정한다. 사용자에 대한 안전성은 암호학적 기법들의 안전성에 근거한다.

4.1 안전성 분석

■ 상점에 대한 안전성

핑거프린트를 삽입하는 삽입기법의 특성에 근거하여, 최대 공모공격의 크기를 넘지 않거나 공모하여 재배포된 디지털 데이터가 원래의 데이터와 충분히 비슷한 경우에, 상점은 emb값을 추출해낼 수 있다.

사용자는 상점으로부터 디지털 데이터를 구매하기 전에 등록센터에 자신의 아이디를 Schnorr 서명기법을 이용하여 서명해 놓기 때문에 사용자가 재배포자로 발각됐을 경우 이를 회피할 수 없게 된다.

또한 양자간의 안전한 계산과정을 통하여 상점의 도전값에 대한 사용자의 응답값이 틀리거나, 등록센터로부터 받은 $Cert$ 값이 검증되지 않을 경우 사용자가 디지털 데이터를 얻을 수 없게 된다.

■ 사용자에 대한 안전성

사용자는 공격자가 다른 참여자들과 공모를 하고 사용자가 구매한 다른 구매내역들을 알고 있다 하더라도 특정한 데이터에 대한 정보가 없는 한 정직한 조정자를 심리 프로토콜과정에서 확신시키지 못하기 때문에 안전하다. 또한 상점과 등록센터가 공모를 한다 하더라도 사용자의 Schnorr 서명의 비밀키 값을 알지 못하기 때문에 $proof$의 유효한 Schnorr서명을 만들어 낼 수 없다.

■ 등록센터에 대한 안전성

사용자는 등록시에 자신의 Schnorr 서명의 비밀키 값을 이용하여 h_1,x' 에 서명을 수행하게 된다. 따라서 사용자 자신이 아닌 다른 사람이 사용자를 가장할 경우 등록 과정의 다음 두 계산에 대하여 등록센터가 올바른 답을 얻을 수 없게 된다.

$$z=g^y v^e,\quad e=h(h(h_1 \| x') \| z)$$

위의 안전성은 Schnorr 서명기법의 안전성에 기반한다.

4.2 사용자의 익명성

핑거프린팅 프로토콜을 따르는 정직한 사용자는 이산대수문제와 양자간 안전한 계산과정의 안전성에 근거하여 사용자의 신원이 드러나지 않는다. 상점은 사용자로부터 $h(b'')$과 $Cert$ 값만을 받으며, 양자간 안전한 계산과정의 안전성에 의하여 h_1을 알 수 없고 만약 알아냈다 하더라고 이산대수 문제의 안전성에 근거하여 아이디 U를 알아낼 수 없다.

5. 결론

본 논문에서는 비밀분산법을 이용하여 사용자의 신원정보를 숨겨두었다가 재배포시 드러나도록 구성하고, Schnorr 서명을 이용하여 고발된 사용자가 범행을 부인할 수 없도록 하는 새로운 익명성을 보장하는 핑거프린팅 기법을 제안하였다.

제안한 방법은 이산대수문제와 양자간 안전한 계산과정의 안전성에 근거하여 사용자의 익명성이 보장된다. Schnorr 서명기법을 은닉 Schnorr 서명기법으로 바꿀 경우 등록시 사용자의 익명성도 보장할 수 있다.

참고문헌

[1] G. Blakley, C. Meadow and G. B. Purdy, "Fingerprinting long forgiving messages," In Advances in Cryptology-CRYPTO'85, LNCS 218, pp. 180-189, 1986.

[2] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," In Advances in Cryptology - CRYPTO '95, LNCS 963, pp. 452-465, 1995.

[3] D. Chaum, I. B. Damagaard and J. vad de Graaf, "Multiparty computations ensuring privacy of each party's input and correctness of the result," In Advances in Cryptology -CRYPTO'87, LNCS 293, pp. 87-119, 1988.

[4] B. Pfitzmann and A. R. Sadeghi, "Coin-based anonymous fingerprinting," In Advances in Cryptology - EUROCRYPT '99, LNCS 1592, pp. 150--164, 1999.

[5] B. Pfitzmann and M. Schunter, "Asymmetric fingerprinting," In Advances in Cryptology-EYROCRYPT'96, LNCS 1070, pp. 84-95, 1996.

[6] B. Pfitzmann and M. Waidner, "Anonymous fingerprinting," In Advances in Cryptology-EUROCRYPT'97, LNCS 1233, pp. 88-102, 1997.

[7] C. Schnorr, "Efficient signature generation for smart cards," Journal of Cryptology, 4(3), pp. 161-174, 1991.

[8] A. Shamir, "How to share a secret," CACM, 22(11), pp. 612-613, 1979.