블라인드 XTR-DSA 서명을 이용해 블랙메일링을 막는 효율적인 방법

박 혜 영*, 한 동 국*, 이 동 훈*, 이 상 진*

*고려대학교 정보보호기술연구센터

An Efficient Method Defeating Blackmailing Using Blind XTR-DSA Scheme

Hye-Young Park*, Dong-Guk Han*, Dong Hoon Lee*, Sangjin Lee*

*Center for Information Security Technologies(CIST), Korea University

요 약

본 논문에서는 XTR에 기반한 블라인드 서명인 블라인드 XTR-DSA 스킴을 이용하여 전자화폐 시스템에서 블랙메일링 공격을 막는 방법을 제안한다. 인출과정에서 은행이 블라인드 XTR-DSA 스킴을 이용하여 서명을 하면 효과적으로 블랙메일리에게 표시된 화폐를 발행할 수 있다. 기존의 방법에서는 블랙메일리에게 표시된 화폐를 발행하기 위해 사용자의 개인키를 은행에 전달해야했으나 본 논문에서 제안하는 블라인드 XTR-DSA 스킴을 이용할 경우 사용자의 개인키 전달이 필요 없게 되어 기존의 방법보다 더욱 효율적이다.

I. 서론

전자화폐시스템에서 사용자의 익명성을 보장하는 것은 전자화폐가 실질 화폐와 같은 역할을 할 수 있게 하기 위해 반드시 필요한 요소이다. 그러나 이러한 익명성의 보장은 개인의 프라이버시를 보장하기 위한 원래의 목적을 악용한 범죄에 이용될 수 있다. 이에 사용자의 익명성을 보장하면서 블랙메일링과 같은 범죄가 발생할 경우 익명성을 취소할 수 있는 전자화폐 시스템들이 제안되었다. 이러한 지불시스템에서 TTP는 불법적인 거래의 경우 사용자의 익명성을 취소할 수 있다. 그러나 TTP가 자신의 능력을 남용한다면 정당한 사용자의 프라이버시는 침해받을 수 있다. 이에 Kugler 와 Vogt는 TTP 없이 사용자의 익명성은 보장하면서 블랙메일된 돈에 대한 익명성을 취소할 수 있는 온라인 전자 지불시스템을 제안하였다[10]. 일반적으로 블랙메일링은 블랙메일리의 능력에 따라 다음과 같이 분류할 수 있다.

완벽한 범죄(Perfect crime) 이것은 블랙메일리가 피해자에게 익명채널을 통해 접근하여 자신에 의해 선택되이지고 블라인딩된 화폐를 인출하도록 협박하는 것이다. 여기서 블랙메일리는 피해자와만 통신한다.

신분위장(Impersonation) 이것은 블랙메일리가 피해자의 은행 계좌에 대한 정보-신분확인에 쓰이는 개인키-를 얻어서 그 자신이 돈을 인출하는 방법이다. 여기서는 블랙메일리가 자신이 은행 계좌의 주인인 것처럼 직접 은행과 통신을 한다. 그러나 블랙메일리는 피해자와 은행의 통신 내용을 알 수 없다.

피해자 납치(Kidnapping) 이것은 블랙메일리가 피해자를 육체적으로 제압하여 신분위장과 유사한 방법으로 화폐를 인출하는 방법이다. 신분위장과 마찬가지로 블랙메일리가 직접 은행과 통신을 한다.

Kugler와 Vogt에 의해 제안된 전자지불 시스템에서 블랙메일링을 막기 위한 주요 아이디어는 블랙메일리에게 표시된 화폐를 인출하여 주는 것이다[10]. 은행은 피해자로부터 블랙메일링의 정보를 얻은 후 일반적인 경우와 다른 마킹키(marking key)를 이용하여 서명한 화폐를 전달한다. 그러나 은행이 고의로 정당한 사용자에게 표시된 돈을 발행하여 사용자의 프라이버시를 침해할 수 있으므로 확인 프로토콜(confirmation protocol)을 통해 정당한 사용자에게 화폐가 표시되지 않았다는 것을 확신시켜 준다. 블랙메일링의 경우 블랙메일리 역시 확인 프로토콜을 통해 화폐의 표시유무를 검증할 수 있게 되므로 은행은 확인 프로토콜을 조작하여 블랙메일리가 표시된 화폐를 받도록 해야 한다. 조작된 확인 프로토콜을 생성하기 위해서 사용자의 개인키가 필요하고 이것은 블랙메일링의 정보와 함께 인출 전에 은행에 전달되어야 한다. 그러나 가장 강력한 공격인 납치(kidnapping)의 경우 어떠한 방법으로도 피해자는 은행에게 블랙메일링을 당하고 있다는 정보를 줄 수 없다. 이 경우 인증을 위해 두 개의 PIN을 사용하는 안전한 하드웨어(secure hardware)를 구성하여 정상적인 거래에서 사용하는 PIN이 사용되지 않을 경우 확인 프로토콜에서 사용되는 개인키가 함께 전달되도록 하였다. 따라서 완벽한 범죄나 신분위장의 경우 1의 확률로, 납치의 경우 1/2의 확률로 표시된 화폐를 블랙메일리에게 전달할 수 있다.

[10]에서 은행이 표시된 돈을 인출하여 전달하기 위해 피해자는 화폐 인출 전에 블랙메일링 공격에 대한 정보를 줄 수 있다는 가정을 하였다. [7]에서는 이러한 가정이 실질적이지 않음을 지적하고 XTR 개인식별 프로토콜을 구성하여 인증단계에서 블랙메일링의 정보와 개인키를 줄 수 있는 방법을 제시하였다. XTR을 이용하여 Schnorr 개인식별 프로토콜을 구성하면 하나의 시도값에 대해 확인식을 통과하는 세 가지 응답값을 보낼 수 있다. 사용자와 은행은 사전에 응답값의 크기를 약속해 놓으며 블랙메일러는 정확한 값의 크기를 알 수 없으므로 세 값 중 하나를 선택해야 한다. 이에 따라 은행은 2/3의 확률로 블랙메일링의 정보와 피해자의 개인키를 얻게 되어 블랙메일러에게 표시된 화폐를 발행할 수 있다. 본 논문에서는 블랙메일링을 막기 위해 [7]에 제안된 XTR 개인식별 프로토콜과 새롭게 제안하는 블라인드 XTR-DSA 스킴을 이용하여 블랙메일러에게 표시된 화폐를 전달하는 새로운 방법을 제안한다.

II. 블라인드 XTR-DSA 스킴

1. XTR 공개키 시스템

이번 절에서는 XTR 공개키 시스템의 특성에 대해 살펴보도록 하겠다. XTR 공개키 시스템을 살펴보기에 앞서, 유한체, $GF(p^2)$, $GF(p^6)$에서 몇 가지 용어와 기호에 대한 정의를 알아보도록 한다.

- conjugate : $h \in GF(p^6)$의 $GF(p^2)$ 위에서의 conjugates 은 h, h^{p^2}, h^{p^4} 이다.
- trace : $h \in GF(p^6)$의 $GF(p^2)$ 위에서의 trace $Tr(h)$는 h의 $GF(p^2)$ 위에서의 conjugates 의 합이다.

$$Tr(h) = h + h^{p^2} + h^{p^4} \in GF(p^2).$$

- $S_k(Tr(g)) : (Tr(g^{k-1}), Tr(g^k), Tr(g^{k+1}))$

XTR은 원소를 표현하고 그것의 지수승을 계산하는 데에 trace를 이용하는 방법이다. XTR의 시스템 파라메터에 대해 살펴보자. p는 $p \equiv 2 \pmod 3$을 만족하는 170비트 정도의 소수이며, q는 160비트 정도의 소수로 sixth cyclotomic polynomial $\Phi_6(p) = p^2 - p + 1$의 인수가 되게 잡는다. $g \in GF(p^6)$는 위수가 q인 원소이다. 여기서 XTR 부분군의 생성원으로서 $Tr(g)$를 사용한다. $GF(p^2)$의 원소들의 연산의 효율성을 위해, $GF(p)$상에서 $GF(p^2)$에 대한 최적정규기저를 사용하여 $GF(p^2)$의 원소들을 표현한다. $\{\theta, \theta^2\}$를 $GF(p)$상에서 $GF(p^2)$에 대한 최적정규기저라고 하자. 여기서 θ와 θ^2은 $X^2 + X + 1$의 근이 된다. 또한 $\theta^t = \theta^{t \pmod 3}$이므로 $GF(p^2)$의 원소들은 $x_1\theta + x_2\theta^2$로 표현할 수 있으며 x_1과 x_2는 $GF(p)$에 있는 원소이다. XTR은 $Tr(g)$와 $Tr(g^k)$가 주어졌을 때 k를 찾는 이려움에 기반한 공개키 시스템으로써 부분군의 이산대수 문제에 의존하는 암호시스템에 적용될 수 있다.

2. 블라인드 XTR-DSA 스킴

이번 절에서는 XTR 기반의 블라인드 서명인 블라인드 XTR-DSA 스킴과 trace의 성질에 의해 생겨나는 몇 가지 특성에 대해서 살펴보도록 한다. 블라인드 XTR-DSA에서는 서명 검증식을 통과하는 세 가지 인덱스에 대한 다른 서명이 존재하는데, 사용자와 은행은 사전에 한가지 인덱스를 사용하도록 약속한다. 사용자는 은행에게 자신이 선택한 메시지 m에 대해 약속된 인덱스에 대한 서명을 받고자 한다. 시스템 파라메터는 II.1에서 제시된 것을 따르는 것으로 한다.

■ 사전 작업

사용자와 은행은 인덱스 $i(1 \le i \le 3)$에 대해 일반적인 거래의 경우 사용할 서명의 인덱스에 대해 약속한다.

■ System setup

은행의 서명 생성키 : x ($0 < x < q$)

은행의 서명 확인키 : $S_x(Tr(g))$

■ 블라인드 XTR-DSA 서명 생성

1. (a) 은행은 난수 $k' \in [2, q-3]$를 선택한 후, [11]의 알고리즘 2.3.7을 이용하여 k'와 $Tr(g)$에 기반하여 $Tr(g^{k'})$를 계산한다.

(b) 은행은 $Tr(g^{k'}) = x_1\theta + x_2\theta^2$ ($x_1, x_2 \in GF(p)$)으로 표현한 후, $R' = (x_1 + p \cdot x_2) \bmod q$ 를 계산한다.

(c) R'와 q가 서로소이면 $S_{k'}(Tr(g)) = (Tr(g^{k'-1}), Tr(g^{k'}), Tr(g^{k'+1}))$을 사용자에게 보낸다. 그렇지 않으면 (a)로 돌아간다.

2. (a) 사용자는 $Tr(g^{k'}) = x_1\theta + x_2\theta^2$으로 표현한 후, $R' = (x_1 + p \cdot x_2) \bmod q$를 계산하여 R'와 q가 서로소인지 확인한다.

(b) 사용자는 $\alpha, \beta \in Z_q$를 선택하고 [11]의 Algorithm 2.4.8을 이용하여 α, β, $Tr(g)$ 그리고 $S_{k'}(Tr(g))$에 기반하여 $Tr(g^{k'\alpha+\beta})$를 계산한다.

(c) $Tr(g^{k'\alpha+\beta}) = x_1'\theta + x_2'\theta^2$로 표현한 후 $r = (x_1' + px_2') \bmod q$를 계산한 후 r와 q가 서로소인지 확인한다. 여기서 $x_1', x_2' \in GF(p)$이다. 만약, 서로소가 아니라면 (b)로 돌아간다.

(d) 사용자는 $m' = \alpha m R' r^{-1} \bmod q$를 계산

한 후 은행에 보낸다.

3. (a) 은행은 i $(1\le i\le 3)$ 가운데 약속된 인덱스에 대해 $s'=(k'm'+R'x)p^{2(i-1)} \bmod q$ 를 계산한 후 s'를 사용자에게 보낸다.

4. (a) 사용자는 i $(1\le i\le 3)$ 가운데 약속된 인덱스에 대해 $s=s'rR'^{-1}+\beta mp^{2(i-1)} \bmod q$ 를 계산한다.

(b) 사용자는 메시지 m에 대한 은행의 서명 을 (m, r, s)로 한다.

■ 블라인드 XTR-DSA 서명 확인

(m, r, s)가 정당한 서명인지 확인하기 위해 i $(1\le i\le 3)$에 대해 $Tr(g^s\cdot y^{rp^{2(i-1)}})^{m^{-1}}=z_1\theta+z_2\theta^2$ $(i, 1\le i\le 3)$를 계산한다. 여기서 $z_1, z_2\in GF(p)$이다. $T=(z_1+pz_2) \bmod q$를 계산하여 $r=T$가 성립하면 서명을 받아들인다. $Tr(g^s\cdot y^{rp^{2(i-1)}})^{m^{-1}}$의 계산은 s, r, p, $Tr(g)$와 $S_x(Tr(g))$에 기반하여 [11]의 알고리즘 2.4.8을 이용한다.

정리 3. 변형된 블라인드 XTR-DSA은 블라인드 서명 스킴이다.

[증명.] $i=1$에 대해 얻은 서명에 대하여 이것이 블라인드 서명임을 보이자. 프로토콜의 수행동안 생성되는 서명자의 뷰(view)는 k', $R'=g^{k'}$, $m', s'=k'm'+R'x \bmod q$가 된다. 서명자가 얻은 뷰(view)는 다음과 같은 식을 만족하게 된다.

$$m'=\alpha mR'r^{-1} \bmod q$$
$$s=s'rR'^{-1}+\beta m \bmod q$$
$$r=r'_1+p\cdot r'_2 \bmod q$$

이 때, $Tr(R'^{\alpha}g^{\beta})=r'_1\theta+r'_2\theta^2$ 을 만족하며 r'_1과 r'_2는 $GF(p)$의 원소이다. m, R', r은 q와 서로소 이므로 블라인딩 요소 α, β는 세 식 가운데 위의 두 식에 의해 다음과 같이 유일하게 결정된다.

$$\alpha=m'm^{-1}rR'^{-1} \bmod q$$
$$\beta=(s-s'rR'^{-1})m^{-1} \bmod q$$

β에서 s'대신 $k'm'+R'x \bmod q$를 대입하면,

$k'\alpha+\beta=k'm'm^{-1}+sm^{-1}-sR'^{-1}m^{-1}$

$=(s-rx)m^{-1} \bmod q$를 만족한다. 따라서,

$Tr(R'^{\alpha}g^{\beta})=Tr(g^{k'\alpha+\beta})$

$=Tr(g^{(s-rx)m^{-1}})=Tr((g^sy^{-r})^{m^{-1}})=r'_1\theta+r'_2\theta^2$.

$T=(r'_1+p\cdot r'_2) \bmod q$이므로 $r=T$이다. □

정리 4. 은행이 서명 생성에 사용한 인덱스와 사용자가 서명 검증에 사용하는 인덱스가 같으면 각각의 i $(1\le i\le 3)$에 대해 (m, r, s)는 항상 서명확인식을 통과한다.

III. 블랙메일링을 막는 방법

이번 장에서는 2장에서 제안된 블라인드 XTR-DSA 서명을 이용하여 블랙메일링을 효과적으로 막는 방법을 제시한다.

준비단계 은행과 사용자는 새로운 계좌를 열 때 인증과정에서 사용되는 응답 값의 크기를 약속하며 은행은 항상 응답값의 크기에 해당하는 서명을 발행하는 것으로 한다. 즉, 인증과정에서 가장 작은 응답 값을 사용하기로 하고 은행은 3. 3(a)에서 $i=1$에 대한 서명을 발행하기로 하였다고 하자. 또한 은행이 화폐의 표시 유무를 확인할 수 있도록 하기 위해 화폐에 항상 정당한 인덱스에 대한 정보가 포함되어야 한다. 이를 위해 사용자는 안전한 하드웨어를 사용하며 하드웨어와 은행사이에는 비밀키 E_k가 공유되어 있다고 가정하자. 사용자가 은행으로부터 화폐를 인출 받아 사용할 때 화폐에 반드시 안전한 하드웨어에서 생성된 특정한 값이 포함되어야 한다. 이 값은 인증이 발생했을 때의 시간정보 $Time$과 사용자가 사용하는 정당한 인덱스값 i와 $Time$을 E_k로 암호화 된 $E_K(i\|Time)$의 값으로 구성된다. 여기서 하드웨어 내부에서 지원하는 $Time$은 항상 다른 값이라고 가정한다. 사용자가 사용하는 정당한 인덱스의 값은 고정되어 있으나 항상 다른 $Time$과 함께 암호화되므로 $E_K(i\|Time)$도 항상 다른 값이 된다. 예를 들어 블랙메일러가 세 명의 서로 다른 인덱스를 쓰는 사람과 공모하여 동시에 같은 $Time$에 대응되는 $i(1\le i\le 3)$에 대한 $E_K(i\|Time)$을 생성할 수 없다는 것이다.

표시된 돈의 인출 블랙메일러가 은행과 XTR 개인식별 프로토콜을 수행하는 동안 은행은 블랙메일러가 선택한 응답값에 따라 2/3의 확률로 블랙메일링에 대한 정보를 얻는다. 은행은 블랙메일러가 선택한 응답값에 따라 2/3의 확률로 일반적인 거래의 경우와 다른 인덱스에 대한 서명을 갖는 화폐를 발행한다. XTR 블라인드 서명을 이용하면 표시된 돈, 즉 정당한 경우와는 다른 인덱스로 서명된 돈이 어떠한 조작을 가하지 않아도 i $(1\le i\le 3)$에 대한 서명검증식 중 반드시 하나에서 통과한다. 은행의 전자서명은 블라인드 서명값과 안전한 하드웨어에서 생성된 $Time$, $E_K(i\|Time)$으로 구성된다. 그러나 유효한 서명을 얻은 사람은 서명검증을 통해 은행과 사용자 사이에 약속된 인덱스에 대한 정보를 알 수 있으므로 은행의 서명은 암호화되어 전송되어야 한다.

표시된 돈의 인식 블랙메일러가 얻은 화폐로 상점에서 물건을 구매하면 서명이 은행의 공개키로 암호화되어 있으므로 상점은 이를 검증할 수 없다. 따라서 상점은 이 돈을 바로 은행에 입금시킨다. 돈이 은행에 입금되면 먼저 은행은 자신의

개인키로 복호화 한 후 $i(1 \le i \le 3)$에 대한 검증식을 계산하여 어떠한 인덱스에 대해 통과하는지 확인한다. 그리고 화폐에서 $E_K(i \| Time)$부분을 복호화하여 이것이 사용하는 정당한 인덱스와 위의 검증식에서 통과한 인덱스를 비교한다. 만약 이것이 다르다면 블랙메일된 화폐라는 것을 알 수 있다. 그러나 블랙메일러는 각각 다른 i를 사용하는 사용자의 정상적인 거래에서 모든 i에 해당하는 $E_K(i \| Time)$을 얻어낸 후, 화폐 정보에 해당하는 $(Time, E_K(i \| Time))$을 변경하여 정당한 인덱스 정보를 바꿀 수 있다. 그러나 본 논문에 제안된 시스템에서는 블랙메일러가 이러한 공격을 하는 경우에도 높은 확률로 블랙메일링 공격을 막을 수 있다. 블랙메일러가 위와 같은 공격을 하는 경우 블랙메일러가 실패할 확률, 즉 은행이 블랙메일된 화폐를 표시된 돈으로 인식하여 블랙메일링 공격을 막을 수 있는 확률은 다음과 같다. 블랙메일러가 일반적인 거래에서 사용되는 응답의 크기의 인덱스를 선택하는 사건을 *Correct*라 하고 블랙메일러가 화폐정보 $(Time, E_K(i \| Time))$를 변경하는 사건을 *Modify*라 하자.

$$\Pr[\sim Modify \mid Correct] = \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{6} : \text{성공}$$

$$\Pr[Modify \mid Correct] = \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{6} : \text{실패}$$

또한 블랙메일러가 응답의 크기를 잘못 선택하는 사건을 InCorrect라 하자.

$$\Pr[\sim Modify \mid \in Correct] = \frac{2}{3} \cdot \frac{1}{2} = \frac{2}{6} : \text{실패}$$

$$\Pr[Modify \text{ into agreed index } \mid \in Correct]$$

$$= \frac{2}{3} \cdot \frac{1}{2} \cdot \frac{1}{3} = \frac{1}{9} : \text{성공}$$

$$\Pr[Modify \text{ into disagreed index } \mid \in Correct]$$

$$= \frac{2}{3} \cdot \frac{1}{2} \cdot \frac{2}{3} = \frac{2}{9} : \text{실패}$$

블랙메일러는 자신이 인증과정에서 우연히 정당한 값을 선택하였더라도 화폐 정보를 수정함으로써 은행이 표시된 화폐로 잘못 인식할 수 있으며, 자신이 인증과정에서 정당하지 않은 값을 선택하였더라도 화폐 정보를 수정하여 은행이 표시되지 않은 화폐로 받아들일 수 있다. 그러나 모든 경우에서 블랙메일러에게 인출한 표시된 돈을 은행이 인식할 확률은 1/6 + 2/6 + 2/9 = 13 / 18 이 된다. 이것은 [10]에서 납치의 경우 1/2의 확률로 블랙메일링을 막는 것에 비해서 상당히 큰 확률임을 알 수 있다. 은행은 위와 같이 입금된 돈에 대한 표시유무를 확인한 후 사용자의 별다른 요구가 없을 경우 표시된 돈을 받아들인다. 사용자가 자신의 계좌에서 발행된 표시된 돈에 대해 은행이 거부하도록 요청할 때 [14]에 제안된 self-escrowed cash의 아이디어를 적용하여 블랙메일된 화폐와 사용자 정보에 대한 링크를 찾은 후 사용자의 계좌로 재입금하여 준다.

Ⅳ. 결론

지금까지 XTR 개인식별 프로토콜과 변형된 XTR 블라인드 서명을 바탕으로 블랙메일링을 막는 방법을 살펴보았다. 기존의 블랙메일링을 막기 위한 시스템에서는 블랙메일리에게 표시가 된 돈을 지급하기 위해 인증, 인출, 확인 프로토콜을 거쳐야 했으나 제안된 방법을 이용하면 인증과 인출프로토콜만으로도 블랙메일리에게 표시가 된 돈, 즉, 정당한 경우와는 다른 서명을 갖는 돈을 지급할 수 있는 장점이 있다. 또한 [10]에서 블랙메일러가 표시된 돈을 받도록 속이기 위해 사용자의 개인키가 필요했으나 본 논문에 제안된 방법에서는 은행이 블랙메일링의 정보만 얻으면 블랙메일리에게 표시된 화폐를 줄 수 있었다. 따라서 블랙메일러가 구분할 수 없는 표시가 된 돈을 더욱 효율적으로 전달할 수 있었다. 또한 [8]에 제안된 최적화된 XTR 유한체를 사용할 경우 본 논문의 아이디어를 더욱 효율적으로 구현할 수 있다.

참 고 문 헌

[1] J. Camenisch, U. Mauer, and M. Stadler. Digital payment systems with passive anonymity-revoking trustees. *In Computer Security-ESORICS '96*, volume 1146 of Lecture Notes in Computer Scienc, pages 31-43. Springer-Verlag, 1996.

[2] D.Chaum. Blind signature for untraceable payments. *In Advances in Cryptology-CRYPTO '82*,pages 199-203, Plenum,1 983.

[3] D. Chaum. Security Without Identification: Transaction Systems to Make Big Brother Obsolete. *Communications of the ACM 28*, 10, October 1985.

[4] D. Chaum. Privacy Protected Payments: Unconditional Payer And/Or Payee Untraceability. *In Smartcard 2000*, pages 69-93, 1989.

[5] G.Davida, Y.Frankel, Y.Tsiounis, M. Yung, "Anonymity Control in E-Cash Systems", *Proceedings of Financial Cryptography Workshop, February, (1997)*, 15 pages.

[6] G. Davide, Y, Tsiounis, and M. Young. Anonymity control in e-cash systems. *In Financial Cryptography '97*, volume 1318 of Lecture Notes in Computer Science, pages 1-16. Springer-Verlag, 1997.

[7] Dong-Guk Han, Hye-Young Park, young-
Ho Park, Sangjin Lee, Dong Hoon Lee, and Hyung-Jin Yang. A Practical Approach Defeating Blackmailing, *Proceedings of ACISP 2002*, LNCS 2384, Springer-Verlag 2002, 464-481.

[8] Dong-Guk Han, Ki Soon Yoon, Young-Ho Park, Chang Han Kim, Jongin Lim, Optimal Extension Fields for XTR, Proceedings of Selected Areas in Cryptography(SAC 2002), accepted.

[9] M. Jakobsson and J. Muller. Improved magic ink signatures using hints. *In Financial Cryptography: Third International Conference, FC '98*, Anguilla, British West Indies, 1999. Springer-Verlag.

[10] D. Kugler and H. Vogt. Marking: A Privacy Protecting Approach Against Blackmailing, *Proceedings PKC 2001*, LNCS 1992, Springer-Verlag, 2001, 137-152.

[11] A.K. Lenstra, E.R. Verheul, The XTR public key system, *Proceedings of Crypto 2000*, LNCS 1880, Springer-Verlag, 2000,1-19; available from www.ecstr.com.

[12] A.K. Lenstra, E.R. Verheul, Key improvements to XTR, *Proceeding of Asiacrypt 2000*, LNCS 1976, Springer-Verlag, 2000,220-233; available from www.ecstr.com.

[13] B. von Solms and D.Naccache. On blind signatures and perfect crimes. *Computers and Security*, 11(6): 581-583,1992.

[14] Birgit Pfitzmann and Ahmad-Reza Sadeghi, Self-Escrowed Cash against User Blackmailing, *Proceedings of Finalcial Cryptography 2000*, LNCS 1962, Springer-Verlag, 2001, pp. 42-52.