基于 XTR 体制的盲签名方案 [1]

陈晓峰 高虎明 王育民

(西安电子科技大学 ISN 国家重点实验室 西安 710071)

摘 要 XTR 是一种新的基于有限域的乘法群的子群中元素迹的紧致表示的公钥密码体制. 与 RSA 和 ECC 相比较, 同等安全程度下 XTR 密钥长度远远小于 RSA , 最多只是 ECC 密钥长度的 2 倍, 但 XTR 参数和密钥选取的速度远远快于 ECC . 利用基于离散对数问题的盲签名方案以及有限域中元素迹的快速算法, 该文给出了两种基于 XTR 体制的盲签名方案, 其安全性等价于解 XTR-DL 困难问题, 但是传输的数据量只有原来方案的 1/3 .

关键词 XTR 公钥体制, 盲签名, 迹表示

中图号 TN918.1

1 引 言

XTR 公钥体制, 即有效的紧致的子群迹表示, 由 Lenstra 等人在 Crypto 2000[1] 提出, 它是一种传统的基于子群离散对数问题的密码体系. 它使用 $\mathrm{GF}(p^2)$ 的算术来达到 $\mathrm{GF}(p^6)$ 上的安全性, 这样 XTR 群中的离散对数问题 (XTR-DL) 就比分解 $6\cdot\log_2 p$-bit RSA 模更为困难. 如果 p,q 取 170-bit 的素数, 则 XTR 就比 1020-bit RSA 更为安全. 而且, XTR 的密钥和参数选取要比椭圆曲线密码体制 (ECC) 简单, 其指数计算比 ECC 标量乘计算快. 所以, 在同等的安全程度下, XTR 就大大减少了数据的存储量, 计算量和通讯量. 目前, 许多新的改进快速算法纷纷提出, XTR 已成为一种非常有吸引力的公钥密码体制.

XTR 并不是最先使用迹表示和计算有限域的子群元素的方幂的体制, 文献 [2] 首先提出使用有限域的扩域及其子群来降低所传输的数据量. 然而, 这种方法非常烦琐, 效率不高. XTR 使用 $\mathrm{GF}(p^2)$ 的迹来表示 $\mathrm{GF}(p^6)^*$ 的阶为 p^2-p+1 子群的元素, 从而使数据量大约降低到原来的 1/3 .

盲签名由 D. Chaum[3] 提出, 即签名者在未知消息 m 的情况下对 m 签名, 任何人都可验证签名的正确性, 然而签名者无法将消息-签名对与某一特定的签名协议联系起来. 盲签名协议在电子现金方案的设计中有着重要的作用 [3,4] . D.Chaum 给出了一种基于 RSA 体制的盲签名协议, 后来 J.Camenisch[5], D.Pointcheval[6] 提出了基于离散对数问题的盲签名方案. 基于 XTR 体制的盲签名协议尚未有人给出, 也许是由于有限域中元素的计算无法平行推广到元素迹的运算, 如 $g^a\cdot g^b=g^{a+b}$, 然而一般情形下 $\mathrm{Tr}(g^a)\cdot\mathrm{Tr}(g^b)\neq\mathrm{Tr}(g^{a+b})$.

利用基于离散对数问题的盲签名方案以及有限域中元素迹的快速算法, 本文给出了两种基于 XTR 体制的盲签名方案, 其安全性等价于解 XTR-DL 困难问题. 由于 XTR 体制的优点, 签名所交换的数据量约为原来的 1/3 , 这样就大大提高了盲签名协议的效率, 从而也就大大提高了现有的基于离散对数体制的电子现金方案的效率.

本文剩余的部分如下安排: 在下一节我们将简单地介绍 XTR 体制并给出有限域中元素迹的快速算法, 然后给出基于 XTR 的盲 Nyberg-Rueppel 签名和盲 Schnorr 签名, 最后我们分析了签名方案的安全性和效率并给出了结论.

2 XTR 体制

2.1 *系统参数* 令 $p\equiv 2\bmod 3$ 是一个素数, 6 次分圆多项式在 p 的值 $\phi_6(p)=p^2-p+1$ 有一个素因子 q . $g\in\mathrm{GF}(p^6)^*$ 的阶为 q, $\mathrm{Tr}(g)\in\mathrm{GF}(p^2)$ 是 g 的迹, 这里 $\mathrm{GF}(p^6)^*$ 表示有限域 $\mathrm{GF}(p^6)$ 的乘法群. 给定 $\mathrm{Tr}(g)$, 由 g 生成的 q 阶子群就称为 XTR 群.

[1] 2002-01-28 收到, 2002-07-03 改回
973 国家重大项目 (批准号: G19990358-04) 资助

令 $c=\mathrm{Tr}(g)$, 多项式 $F(c,X)=X^3-cX^2+c^pX-1\in \mathrm{GF}(p^2)[X]$. 对整数 $n\in Z$ 我们定义 c_n 为 $F(c,X)$ 的根的 n 次幂之和, 即如果 $F(c,h_j)=0, j=0,1,2$, 则 $c_n=h_0^n+h_1^n+h_2^n=\mathrm{Tr}(g^n)$. 显见 $c_1=c$.

给定 c 和任意整数 n, Lenstra 等给出了一个快速算法计算 $S_n(c)=(c_{n-1},c_n,c_{n+1})$. 这样, XTR 的参数就为 p,q,c. 用户选择秘密密钥 n 并通过计算 $S_n(c)$ 得到对应的公钥. 而且 Lenstra 指出 c_{n-1}(或 c_{n+1}) 可以通过 c,c_n 和 c_{n+1}(或 c_{n-1}) 表示出来, 这样 c_{n-1},c_{n+1} 就不必包括在 XTR 公钥信息中.

2.2 $F(c,X)$ 的基本性质 下面我们将给出 $F(c,X)$ 的一些基本性质, 详细的证明见文献 [1,7].

引理 1 多项式 $F(c,X)=X^3-cX^2+c^pX-1$, $c_n=h_0^n+h_1^n+h_2^n=\mathrm{Tr}(g^n)$, 则

(1) $F(c,h_j^{-p})=0,\quad j=0,1,2$.

(2) $c_{-n}=c_{np}=c_n^p,\quad n\in Z$.

(3) $c_{u+v}=c_u\cdot c_v-c_v^p\cdot c_{u-v}+c_{u-2v},\quad u,v\in Z$.

(4) $F(c_n,h_j^n)=0,\quad j=0,1,2,\quad n\in Z$.

引理 2 给定 c,c_{n-1},c_n 和 c_{n+1}, 则

(1) $c_{2n}=c_n^2-2c_n^p$.

(2) $c_{n+2}=c\cdot c_{n+1}-c^p\cdot c_n+c_{n-1}$.

(3) $c_{2n-1}=c_{n-1}\cdot c_n-c^p\cdot c_n^p+c_{n+1}^p$.

(4) $c_{2n+1}=c_{n+1}\cdot c_n-c\cdot c_n^p+c_{n-1}^p$.

定义 1 令 $C(V)$ 表示 3×3 矩阵 V 的中间列, 而且

$$A(c)=\begin{pmatrix}0&0&1\\1&0&-c^p\\0&1&c\end{pmatrix},\quad M_n(c)=\begin{pmatrix}c_{n-2}&c_{n-1}&c_n\\c_{n-1}&c_n&c_{n+1}\\c_n&c_{n+1}&c_{n+2}\end{pmatrix}$$

引理 3 $M_0(c)$ 的行列式的值为 $D=c^{2p+2}+18c^{p+1}-4(c^{3p}+c^3)-27$. 若 $D\neq 0$, 则

$$M_0(c)^{-1}=\frac{1}{D}\begin{pmatrix}2c^2-6c^p&2c^{2p}+3c-c^{p+2}&c^{p+1}-9\\2c^{2p}+3c-c^{p+2}&(c^2-2c^p)^{p+1}-9&(2c^p+3c-c^{p+2})^p\\c^{p+1}-9&(2c^{2p}+3c-c^{p+2})^p&(2c^2-6c^p)^p\end{pmatrix}$$

引理 4 给定 c 和 $S_n(c)$ 则有 $C(A(c)^n)=M_0(c)^{-1}\cdot(S_n(c))^T$.

引理 5 $c_{a+b}=S_b(c)\cdot C(A(c)^a)$.

定理 1 令 p,q 是素数, $q|p^2-p+1$. 对整数 i, $c_i=\mathrm{Tr}(g^i)$, $c_1=c$. 如果 $k\neq p$, $1-p\bmod q$, 则

$$c_{k-1}=\frac{c_k^p(c^{2p}-3c)-c_{k+1}^p(c^2-3c^p)-c_{k+1}^2c^p+c_k^2(c-c^{2p})+c_kc_{k+1}c^{p+1}}{cc_{k+1}-c^pc_k}$$

$$c_{k+1}=\frac{c_k^p(c^2-3c^p)-c_{k-1}^p(c^{2p}-3c)-c_{k-1}^2c+c_k^2(c^p-c^2)+c_kc_{k-1}c^{p+1}}{c^pc_{k-1}-cc_k}$$

算法 1 输入 c 和 c_n, 输出 "最小" 的 c_{n+1}.

步骤 1 使用文献 [7] 中算法 3.1, 5.2, 5.5 计算 $F(c,X)$ 的一个根 $g\in \mathrm{GF}(p)$.

步骤 2 使用文献 [7] 中算法 3.1 计算 $F(c_n,X)$ 的根 y_1,y_2,y_3.

步骤 3 计算 gy_i 的迹 t_i, $i=1,2,3$.

步骤 4 输出 t_i 中的 "最小者".

所以，由定理 1 和算法 1 我们知：c_{n-1}, c_{n+1} 不必包括在 XTR 公钥信息中.

3 迹的快速计算算法

给定 $c, S_k(c), a, b$, 在 k 未知时 Lenstra 等给出了计算 c_{a+bk} 的快速算法，然而我们无法直接使用该算法构造盲签名方案 (特别是盲 Schnorr 签名方案)．所以，我们需要给出在 A, B 未知时给定 c_a, c_b 如何计算 c_{a+b} 的快速算法.

算法 2　输入 c_a, c_b，输出 c_{a+b}．这里 a, b 是未知的整数.

步骤 1　使用算法 1 计算 c_{a+1}, c_{b+1}.

步骤 2　利用定理 1 计算 c_{a-1}, c_{b-1}，得到 $S_a(c) = (c_{a-1}, c_a, c_{a+1}), S_b(c) = (c_{b-1}, c_b, c_{b+1})$.

步骤 3　使用引理 4 计算 $C(A(c)^a) = M_0(c)^{-1} \cdot (S_a(c))^T$.

步骤 4　利用引理 5 计算 $c_{a+b} = S_b(c) \cdot C(A(c)^a)$，输出 c_{a+b}.

4 基于 XTR 的盲签名方案

4.1 XTR-Blind-Nyberg-Rueppel 签名方案　系统参数同 XTR 的参数．A 的签名私钥为 x，所对应的公钥为 $y = \mathrm{Tr}(g^x)$．B 所要签名的消息为 m．对称加密算法 $E_K(\cdot)$.

步骤 1　A 随机选择 $\tilde{k} \in z_q$，计算 $S_{\tilde{k}}(c)$ 并发送 $\tilde{r} = \mathrm{Tr}(g^{\tilde{k}}) \in \mathrm{GF}(p^2)$ 给 B.

步骤 2　B 随机选择 $\alpha \in z_q, \beta \in z_q^*$，利用算法 2 计算 $\mathrm{Tr}(g^{\alpha+\beta\tilde{k}}) \in \mathrm{GF}(p^2)$.

步骤 3　B 利用 $\mathrm{Tr}(g^{\alpha+\beta\tilde{k}})$ 确定对称加密算法的密钥 K(如选取 $\mathrm{Tr}(g^{\alpha+\beta\tilde{k}})$ 的前 192-bit)，然后计算 $r = E_K(m) \bmod q, \tilde{m} = r\beta^{-1} \bmod q$.

步骤 4　B 检验 $\tilde{m} \in z_q^*$ 并发送 $\tilde{m}$ 给 A，否则返回步骤 2.

步骤 5　A 计算 $\tilde{s} = \tilde{m}x + \tilde{k} \bmod q$ 并发送 $\tilde{s}$ 给 B.

步骤 6　B 计算 $s = \tilde{s}\beta + \alpha \bmod q$，则对消息 m 的盲签名为 (r, s).

验证　验证者首先计算 $S_s(\mathrm{Tr}(g))$ 和 $S_{-r}(y)$，利用算法 2 计算 $\mathrm{Tr}(g^{s-rx})$，取 $\mathrm{Tr}(g^{s-rx})$ 的前 192-bit K^* 作为对称加密算法的密钥．如果 $E_{K^*}(m) = r \bmod q$，则签名正确．否则，拒绝签名．这是因为 $\mathrm{Tr}(g^{s-rx}) = \mathrm{Tr}(g^{\tilde{s}\beta+\alpha-rx}) = \mathrm{Tr}(g^{\tilde{m}x\beta+\tilde{k}\beta+\alpha-rx}) = \mathrm{Tr}(g^{\alpha+\tilde{k}\beta})$，于是 $K = K^*$.

注：如果不使用对称加密算法，方案中的步骤 3 也可由如下步骤 3* 代替：

步骤 3*　不妨设 $\mathrm{Tr}(g^{\alpha+\beta\tilde{k}}) \in \mathrm{GF}(p^2) = (a, b)$，这里 $a, b \in \mathrm{GF}(p)$．B 计算 $r = (ap + b) \cdot m \bmod q, \tilde{m} = r\beta^{-1} \bmod q$.

4.2 XTR-Blind- Schnorr 签名方案　系统参数如上，$H(,)$ 是一个安全的 hash 函数.

步骤 1　A 随机选择 $k \in z_q$，计算 $S_k(\mathrm{Tr}(g))$ 并发送 $r = \mathrm{Tr}(g^k) \in \mathrm{GF}(p^2)$ 给 B.

步骤 2　B 随机选择 $\alpha \in z_q, \beta \in z_q^*$，利用算法 2 计算 $r' = \mathrm{Tr}(g^{k+\alpha+\beta x}) \in \mathrm{GF}(p^2)$.

步骤 3　设 $r' = (a, b)$．B 计算 $e' = H(m, ap + b), e = e' - \beta \bmod q$ 并发送 e 给 A.

步骤 4　A 计算 $s = k - ex \bmod q$ 并发送 s 给 B.

步骤 5　B 计算 $s' = s + \alpha \bmod q$，则对消息 m 的盲签名为 (e', s').

验证　验证者首先计算 $S_{s'}(\mathrm{Tr}(g))$ 和 $S_{e'}(y)$，利用算法 2 计算 $\mathrm{Tr}(g^{s'+e'x}) = (a', b')$．如果 $H(m, a'p + b') = e'$，则签名正确．这是因为 $\mathrm{Tr}(g^{s'+e'x}) = \mathrm{Tr}(g^{s+\alpha+(e+\beta)x}) = \mathrm{Tr}(g^{k+\alpha+\beta x})$.

4.3 签名方案的分析　由于 XTR 体制本身的优点，签名方案中所交换的数据量是原来方案的 1/3，这样就大大提高了盲签名方案的效率，从而提高了电子商务中协议的效率.

方案的安全性基于 XTR 离散对数问题，当 p, q 取 170-bit 的素数，则方案就比 1020-bit RSA 更为安全.

5 结　论

XTR 是一种非常有吸引力的公钥密码体制．与目前实用的 RSA 和 ECC 公钥密码体制相比较 [8]，同等安全程度的 XTR 体制的实现在计算、密钥存储和通信方面的要求与 ECC 基本相同，但 XTR 的密钥生成要比 ECC 快的多，迹的计算也比点的标量乘运算要快．当然，如何进一步改进 XTR 的算法，优化参数选取，使 XTR 走上实用需要进一步的工作 [9–12].

盲签名体制在电子商务中起着非常重要的作用，本文利用迹的快速算法，给出了基于 XTR 的两个盲签名方案，使所交换的数据量降低到原来的 1/3，从而大大提高了原有方案的效率。当然，我们的结果也可以推广到对应的公平盲签名方案，这里我们就不再讨论。

参 考 文 献

[1] A. K. Lenstra, E. R. Verheul, The XTR public key system, Crypto'2000, California, USA, LNCS 1880, Springer-Verlag 2000, 1–19.
[2] A. E. Brouwer, R. Pellikaan, E. R. Verheul, Doing more with fewer bits, Asiacrypt'99, Singapore, LNCS 1716, Springer-Verlag, 1999, 321–332.
[3] D. Chaum, Blind signature for untraceable payments, Eurocrypt'82, Burg Feuerstein, Germany, Plenum Press, 1983, 199–203.
[4] M. Stadler, J. M. Piveteau, Jan. Camenisch, Fair blind signatures, Eurocrypt'95, St. Malo, France, LNCS 921, Springer-Verlag, 1995, 209–219.
[5] J. L. Camenisch, J. M. Piveteau, M. A. Stadler, Blind signature based on discrete logarithm problem, Eurocrypt'94, Perugia, Italy, LNCS 950, Springer-Verlag, 1994, 428–432.
[6] D. Pointcheval, J. Stern, Provably secure blind signature schemes, Asiacrypt'96, Kyongju, Korea, LNCS 1163, Springer-Verlag, 1996, 252–265.
[7] A. K. Lenstra, E. R. Verheul, Key improvements to XTR, Asiacrypt'2000, Kyoto, Japan, LNCS 1880, Springer-Verlag, 2000, 1–19.
[8] A. Menezes, Comparing the security of ECC and RSA, manuscript, 2000, available from www.carc.math.uwaterloo.ca/ajmeneze/misc/cryptopramartical.html.
[9] E. R. Verheul, Evidence that XTR is more secure than supersingular elliptic curve cryptosystems, Eurocrypt'2001, Innsbruck, Austria, LNCS 2045, Springer-Verlag, 2001, 195–201.
[10] A. K. Lenstra, E.R. Verheul, Fast irreducibility and subgroup membership testing in XTR, PKC'2001, Cheju Island, Korea, LNCS 1992, Springer-Verlag, 2001, 78–86.
[11] A. Menezes, S. A. Vanstoe, ECSTR(XTR): Elliptic curve singular trace representation, Rump Session of Crypto'2000.
[12] M. Stam, A. K. Lenstra, Speeding up XTR, available from www.ecstr.com/.

BLIND SIGNATURE SCHEMES BASED ON XTR SYSTEM

Chen Xiaofeng Gao Huming Wang Yumin

(*National Key Laboratory on ISN, Xidian University, Xi'an* 710071, *China*)

Abstract XTR is a new public key system based on a method to represent elements of a subgroup of a multiplicative group of a finite field. Compared to RSA and ECC, XTR keys are much smaller than RSA keys of equivalent security, and at most twice as big as ECC keys, but parameter and key selection for XTR are much faster than ECC. Based on XTR system by using traditional blind signature schemes based on discrete logarithm problem and fast method for computing the trace of the elements in the finite field, two blind signature schemes are presented in this paper, the security of which is equivalence to solving XTR-DL problem while the datum is only as 1/3 as that of the previous schemes.

Key words XTR public key system, Blind signature, Trace representation

陈晓峰： 男，1976 年生，博士生，研究方向为电子商务，公钥密码体制及应用.
高虎明： 男，1963 年生，副教授，博士生，研究方向为电子商务，网络安全.
王育民： 男，1936 年生，教授，博士生导师，研究领域为信息理论，编码及密码理论.