文章编号:1001 - 9081(2006)07 - 1810 - 03

面向多服务的可灵活撤销的非对称公钥叛逆者追踪方案

张学军[1,2],王育民[2]
(1. 西北师范大学 教育技术与传播学院, 甘肃 兰州 730070;
2. 西安电子科技大学 计算机网络与信息安全教育部重点实验室, 陕西 西安 710071)
(xjzhang99@163.com)

摘　要: Matsushita 等人提出了一个可灵活撤销用户的公钥叛逆者追踪方案,但其方案是对称方案,并且没有提供多服务的功能。利用不经意多项式估值协议(OPE)和服务参数提出了一个改进的 Matsushita 方案。改进方案在保持了原 Matsushita 方案可灵活撤销用户、黑盒子追踪、安全性不变的基础上,增加了提供多种服务、防止叛逆者抵赖(非对称)等优点,整体性能好于 Matsushita 方案。

关键词:叛逆者追踪;多服务;非对称;可灵活撤销用户;黑盒子追踪
中图分类号: TP309.2　　**文献标识码:**A

Multi-service-oriented asymmetric public-key traitor tracing scheme with flexible revocation

ZHANG Xue-jun[1,2], WANG Yu-min[2]
(1. *School of Education Technology and Communication, Northwest Normal University, Lanzhou Gansu* 730070, *China*;
2. *Key Laboratory of Computer Network and Information Security, Ministry of Education, Xidian University, Xi'an Shaanxi* 710071, *China*)

Abstract: Matsushita et al. proposed a traitor tracing scheme with flexible user revocation, which was a symmetric scheme, and did not provide multiple-service capability. An improved scheme was proposed by using Oblivious Polynomial Evaluation(OPE) and service parameter. In this improved scheme, such features of Matsushita's as flexible user revocation, black-box tracing and security remained, and other advantages such as providing multi-service and preventing traitors from repudiation(asymmetry) were added. The whole capabilities of the improved scheme were better than that of Matsushita's.

Key words: traitor tracing; multi-service; asymmetry; flexible user revocation; black-box tracing

0　引言

加密数据(如付费电视、广播加密、在线数据库)的安全分发是现代信息社会的一个重要问题。数据提供商(Data Supplier, DS)通过广播信道向授权用户提供加密信息。授权用户用自己的密钥先将加密数据包头中的会话密钥解密,然后再用得到的会话密钥解密加密数据。如果恶意的授权用户将自己的密钥泄露给别的非法用户使用或者某些授权用户先共谋制造出密钥然后让非法用户使用,那么这些恶意的授权用户就称为叛逆者,非法用户称为盗版者,通过盗版者的解码器分析出叛逆者的工作则称为叛逆者追踪。

叛逆者追踪的概念[1]自提出以后,各种叛逆者追踪方案相继被提出来[2~8]。其中文献[1~4]中的方案都是对称方案,其缺陷是系统管理员(或者数据提供商就是系统管理员)无法获得叛逆者无法否认的证据,即不具有不可否认性。因为这些方案中系统管理员掌握所有合法授权用户的密钥信息,如果系统管理员不可信,可以构造非法解密密钥诬陷合法用户。文献[5]首先提出的非对称叛逆者追踪方案解决了对称方案中存在的问题,即只有授权用户自己知道自己的密钥,数据提供商无法陷害因而叛逆者无法抵赖,但该方案用户量大时开销也大,效率不高,无法实际应用。文献[4]提出了一个可灵活撤销用户的公钥叛逆者追踪方案(本文简称 Matsushita 方案),但该方案是对称方案,并且没有提供多服务(如多个电视频道服务、多个数据库服务等)的功能。为此笔者利用不经意多项式估值协议(Oblivious Polynomial Evaluation, OPE)[9]和服务参数提出了一个改进的 Matsushita 方案。改进方案在保持了原 Matsushita 方案可灵活撤销用户、黑盒子追踪、安全性不变的基础上,增加了提供多种服务、防止叛逆者抵赖(即非对称)等优点,整体性能好于 Matsushita 方案。

1　不经意多项式估值协议 OPE

在 OPE 协议中, Bob 知道一个多项式 P, Alice 知道一个值 a。协议执行结束后, Alice 获得 $P(a)$,但是无法获得多项式 P 除了 $P(a)$ 以外其他任何信息,同时 Bob 也不能得到有关 a 的信息。协议过程如下:Bob 随机选取一个二元多项式 $Q(x, y)$,满足 $Q(0, y) = P(y)$,用来隐藏 $P(y)$;Alice 随机选取一个一元多项式 $S(x)$,满足 $S(0) = a$,用来隐藏 a。Alice 想在不透露 $S(x)$ 的情况下插值构造一个一元多项式 $R(x) = Q(x, S(x))$,显然有 $R(0) = Q(0, S(0)) = P(S(0)) = P(a)$。设 $R(x)$ 为 z 次多项式,当 Alice 获得 $R(x)$ 上 $z+1$ 个点的值 $\{(x_i, R(x_i)), i = 1, 2, \cdots, z+1\}$ 后,就能利用 Lagrange 插值构造出 $R(x)$,最终获得 $R(0) = P(a)$。$R(x)$ 上 $z+1$ 个点的值可按下面的方法获取:Alice 构造一个随机的序列对 $(x_{i,j}, y_{i,j})$

收稿日期:2006 - 01 - 22　　**基金项目:**国家自然科学基金资助项目(60372046);华为基金资助项目(YSCB2005037NP)
作者简介:张学军(1968 -),男,甘肃兰州人,副教授,博士研究生,主要研究方向:信息安全、叛逆者追踪;　王育民(1936 -),男,北京人,教授,博士生导师,主要研究方向:编码理论、密码学、信息安全.

$(i=0,\cdots,z;\ j=1,2,\cdots,n)$,其中对于每个 j,都有一对 $(x_i,\ S(x_i))$。将所有 $(x_{i,j},\ y_{i,j})$ 发送给 Bob,Bob 计算所有的 $Q(x_{i,j},\ y_{i,j})$ 并将结果发送给 Alice。Alice 应用不经意传输协议[9] OT_n^1 提取需要的值。在获得 $z+1$ 个点的值 $(x_i,\ R(x_i))$ 后,Alice 就能利用 Lagrange 插值构造出 $R(x)$,从而获得 $R(0)=P(a)$。整个 OPE 协议的构造是基于噪声多项式插值问题,而破解噪声多项式重构问题被认为是非常困难的,对此文献[9]中已给出了详细说明。

2　改进方案

设 n 为所有用户数,k 为在一次合谋中最大的叛逆者人数(即合谋门限)。$p,\ q$ 为大素数且 $q\mid p-1,\ q\geqslant n+k+1$。$g$ 为 Z_p^* 上的 q 次单位原根,G_q 为 Z_p^* 的 q 阶子群。设 U 为所有用户集($U\subseteq Z_q/\{0\}$),X 为被撤销用户的集合。所有参与者都了解 $p,\ q,\ g$,除非特别声明,所有运算都在 Z_p^* 上进行。

2.1　系统初始化

数据提供商 DS 秘密随机选择 $a_0,\ a_1,\ \cdots,\ a_k,\ b_1,\ \cdots,\ b_k\in Z_q$,计算系统公钥:

$$e=(g,\ y_{0,0},\ \cdots,\ y_{0,k},\ y_{1,1},\ \cdots,\ y_{1,k})$$
$$=(g,\ g^{a_0},\ \cdots,\ g^{a_k},\ g^{b_1},\ \cdots,\ g^{b_k})$$

用户集 U 被分解为 k 个不同子集 $U_1,\ U_2,\ \cdots,\ U_k$,$U=\bigcup_{i=1}^{k}U_k$,当 $i\neq j$ 时,$U_i\cap U_j=\varnothing$,即 $U_1,\ U_2,\ \cdots,\ U_k$ 构成 U 的划分。

令函数 $f_i(x)=\sum_{j=0}^{k}a_{i,j}x^j \bmod q$,其中 $a_{i,j}=\begin{cases}a_j & i\neq j\\ b_j & i=j\end{cases}$

再令函数 $g_i(x,\ y)=f_i(x)+t_cy$,其中 t_c 为服务 c 对应的服务参数。

2.2　用户注册

当用户 u 注册时,秘密选取随机数 $\alpha_u\in Z_q^*$,发送 $u\parallel pk_u\parallel sign_{sk_u}(g^{\alpha_u})$ 给 DS($pk_u,\ sk_u$ 为 PKI 中用户 u 的公钥和密钥,$sign$ 为可恢复消息的签字)。不失一般性,假定用户 u 被分配给集合 U_i,即 $u\in U_i$,DS 记录注册信息 $i\parallel u\parallel pk_u\parallel sign_{sk_u}(g^{\alpha_u})$,令 $R_u=g^{\alpha_u}$,DS 将 i 发送给用户 u。用户 u 的密钥为 $(i,\ u,\ \alpha_u)$。

2.3　用户订阅服务

1) 当用户 u 订阅服务 c 时,DS 秘密选取一个随机数 $v_{cu}\in Z_q^*$。使用 OPE[9] 协议,用户 u 得到 $s=v_{cu}(f_i(u)+t_c\alpha_u)$。

2) 用户 u 发送 $i\parallel u\parallel pk_u\parallel sign_{sk_u}(g^s\parallel g^{\alpha_u})$ 给 DS。

3) DS 首先恢复出 $g^s\parallel g^{\alpha_u}$,然后恢复出用户 u 注册信息中的 R_u。首先验证 $R_u\overset{?}{=}g^{\alpha_u}$,若不相等则退出订阅服务;若相等则进一步验证 $g^s\overset{?}{=}g^{v_{cu}f_i(u)}(g^{\alpha_u})^{v_{cu}t_c}$,若相等将 v_{cu} 发送给用户 u。

4) DS 记录订购单 $text=i\parallel u\parallel pk_u\parallel sign_{sk_u}(g^s\parallel g^{\alpha_u})$。

5) 用户 u 获得对服务 c 的服务密钥为 $s_{cu}=s/v_{cu}=f_i(u)+t_c\alpha_u=g_i(u,\ \alpha_u)$。

2.4　加密

DS 首先检查 $Y\triangleq X\backslash\{\cup_{j\in\{z\mid U_z\subseteq X\}}U_j\}$ 是否为空集。如果 $Y=\{x_1,\ x_2,\ \cdots,\ x_\omega\}$ 非空,求整数 d,满足 $d(k+1)\leqslant\omega\leqslant d(k+1)+k$,令 $m=d(k+1)+k$。否则($Y=\varnothing$ 或者 $X=\varnothing$),令 $m=k,\ \omega=0$。

DS 随机选择 $c_0,\ c_1,\ \cdots,\ c_m\in Z_q$,$x_{\omega+1},\ c_{\omega+2},\ \cdots,\ x_m\in Z_q\backslash\{U\cup\{0\}\}$(如果 $\omega<m$),$r\in Z_q$,$r_j\in Z_q(j\in\{z\mid 1\leqslant z\leqslant m,\ z\neq 0\ (\bmod\ (k+1))$,$U_{z\bmod(k+1)}\subseteq X$,建立如下分组头:

$$h(r,X)=(h,h',h_{0,0},\cdots,h_{0,m},h_{1,1},\cdots,h_{1,m},H_1,\cdots,H_m)$$

其中:

$$h=g^r\qquad h'=g^{rt_c}\qquad h_{0,j}=y_{0z_j}^r g^{c_j}$$

$$z_j=j \bmod (k+1)$$

$$h_{1,j}=\begin{cases}g^{r_j} & U_{z_j}\subseteq X,\ z_j\neq 0\\ y_{1,z_j}^r g^{c_j} & U_{z_j}\nsubseteq X,\ z_j\neq 0\end{cases}$$

$$H_j=(x_j,\ g^{F(x_j)})\qquad F(x)=\sum_{j=0}^{m}c_jx^j \bmod q$$

$g^{F(0)}=g^{c_0}$,作为数据提供商 DS 与合法用户之间的会话密钥。

2.5　解密

设有未被撤销的合法用户 $u\in U_i(u\notin X)$,用户 u 利用密钥 $(i,\ u,\ \alpha_u)$ 和服务密钥 $g_i(u,\ \alpha_u)$ 执行以下解密步骤:

1) $$h^{g_i(u,\alpha_u)}\mid h'^{\alpha_u}=(g^r)^{g_i(u,\alpha_u)}/(g^{rt_c})^{\alpha_u}$$
$$=g^{rf_i(u)}g^{rt_c\alpha_u}/g^{rt_c\alpha_u}$$
$$=g^{rf_i(u)}$$
$$=h^{f_i(u)}$$

2) $$D_i(u)=\prod_{j=0}^{m}B_{i,j}^{u^j}$$
$$=\prod_{j=0}^{d}(h_{0,j(k+1)}\times h_{0,j(k+1)+1}^{u}\times\cdots\times h_{1,j(k+1)+i}^{u^i}\times\cdots\times h_{0,j(k+1)+k}^{u^k})^{u^{j(k+1)}}$$
$$=\prod_{l=0}^{d}(g_{j=0}^{r\sum_{j=0}^{k}a_{i,j}u^j})^{u^{l(k+1)}}\times g_{j=0}^{\sum_{j=0}^{m}c_ju^j}$$
$$=h^{f_i(u)\sum_{l=0}^{d}u^{l(k+1)}}\times g^{F(u)}$$

其中:

$$d=(m-k)/(k+1),$$

$$B_{i,j}=\begin{cases}h_{0,j} & i\neq j \bmod (k+1)\\ h_{1,j} & i=j \bmod (k+1)\end{cases}$$

3) $$g^{F(u)}=D_i(u)/h^{f_i(u)\sum_{l=0}^{d}u^{j(k+1)}}。$$

4) 令 $x_0=u$,用户 x_0 利用计算出的 $(x_0,\ g^{F(x_0)})$ 及分组头中的 $H_1,\ \cdots,\ H_m$,通过 Lagrange 插值计算会话密钥 $g^{F(0)}$:

$$g^{F(0)}=\prod_{j=0}^{m}(g^{F(x_j)})^{L_j}=g_{j=0}^{\sum_{j=0}^{m}L_jF(x_j)}$$

其中:$L_j=\prod_{0\leqslant l\leqslant m,\ l\neq j}\frac{x_l}{x_l-x_j} \bmod q$。

2.6　黑盒子追踪

改进方案的黑盒子追踪采用二分查找法,其算法与文献[4]相同,追踪效率为 $O(\log n)$。

3　安全性分析

对于 Matsushita 方案的安全性而言,有下面两个定理成立:

定理 1[4]　假定集合 X 中的用户获得了一定数量(多项式界)旧的会话密钥、旧的分组头,并且还获得了新分组头、公钥和它们自己的密钥。如果在此之后,这些用户被撤销,则对于任意的集合 X,且对任何一次合谋中最多不超过 k 个被撤销用户而言,相应于新分组头的会话密钥的计算复杂性至少同 G_q 群中的 DDH 问题一样。

定理 2[4] 对于最多不超过 k 个叛逆者构造的盗版解码器,二分查找黑盒子追踪算法能以 $1-\varepsilon$ 的概率识别至少一个叛逆者,其中 ε 是可以忽略的。

对于改进方案的安全性而言,下面定理成立:

定理 3 对于改进方案的安全性而言,至少同文献[4]一样安全。

证明 首先,改进方案的公钥完全同文献[4]一样。其次,改进方案的分组头中只比文献[4]多出一个元素 $h' = g^{rt_c}$,由于离散对数问题的困难性,显然不能由 $h' = g^{rt_c}$ 得到服务参数 t_c。用户 u 从上述解密步骤1)中只能得到 $h^{f_i(u)}$,但却不能得到 $f_i(u)$;而文献[4]中用户 u 可直接拥有 $f_i(u)$。最后,用户 u 拥有对服务 c 的服务密钥 $g_i(u, \alpha_u)$,但是从 $g_i(u, \alpha_u) = f_i(u) + t_c\alpha_u$ 中不能直接得到 $f_i(u)$,因为用户 u 并不知道 t_c;事实上,最多不超过 k 个叛逆者合谋也不能破解 $f_i(u)$(因为由 k 个叛逆者合谋组成的 k 个方程中含有 $k+1$ 个未知数),证毕。

若用户 u 在执行 OPE 协议时使用的 α_u 与发给 DS 的签字中所含的 α_u 不同,则用户 u 不能通过 DS 的验证(2.3 小节3))。用户 u 的密钥 (i, u, α_u) 中的 α_u 和服务密钥 $g_i(u, \alpha_u)$ 只有自己知道,因此当DS追踪到 u 是叛逆者时,用户 u 无法抵赖,DS的利益得到了保障。同样,DS不知道用户 u 密钥中的 α_u 和服务密钥 $g_i(u, \alpha_u)$,因此 DS 无法陷害无辜用户 u,用户 u 的利益也得到了保障。

4 性能比较

表1中 $O(\cdot)$ 表示时间复杂度,n 为所有用户数,k 为在一次合谋中最大的叛逆者人数(即合谋门限),$|M|$ 表示整数 M 的比特位长,p 和 q 为大素数且 $q \mid p-1$,$l = 4(d+1)k + 3d$,其中 d 是一个整数且满足 Y 的长度小于等于 $d(k+1)+k$,Y 指一个被撤销的用户集合,这些用户与一个或多个未被撤销的用户在他们的每一个子集中共存[4]。

从表1可以看出,文献[4]和改进方案均支持灵活撤销用户的功能;文献[4]和改进方案的黑盒追踪效率和公钥长度相同,分别为 $O(\log n)$ 和 $(2k+2)|p|$;文献[4]的密文长度和密钥长度比改进方案稍短;改进方案提供多服务,但文献[4]不提供;改进方案是非对称方案,但文献[4]不是。考虑到多服务、非对称(防止叛逆者抵赖)的重要性,从总体性能上来看,改进方案要好于文献[4]。

表1 改进方案与文献[4]的性能比较

方案	服务性质	方案性质	灵活撤销	黑盒追踪效率	公钥长度	密文长度	密钥长度
Matsushita 方案	单服务	对称	支持	$O(\log n)$	$(2k+2)\|p\|$	$(l+2)\|p\|$	$\|q\|$
改进方案	多服务	非对称	支持	$O(\log n)$	$(2k+2)\|p\|$	$(l+3)\|p\|$	$2\|q\|$

参考文献:

[1] CHOR B, FIAT A, NAOR M. Tracing Traitors[A]. Advances in Cryptology - CRYPT'94[C]. Berlin: Springer-Verlag, 1994, 257 - 270.

[2] TZENG W-G, TZENG Z-J. A Public-Key Traitor Tracing Scheme with Revocation Using Dynamic Shares[A]. PKC 2001[C]. Berlin: Springer-Verlag, 2001. 207 - 224.

[3] 马华, 曹正文. 基于 RSA 加密算法的叛逆者追踪方案[J]. 西安电子科技大学学报, 2004, 31(4): 611 - 613.

[4] MATSUSHITA T. A Flexibly Revocable Key-Distribution Scheme for Efficient Black-Box Tracing[J]. IEICE Transactions on Fundamentals, 2005, E88-A(4): 1055 - 1062.

[5] PFITZMANN B. Trials of Traced Traitors[A]. Information Hiding'96 [C]. Berlin: Springer-Verlag, 1996. 49 - 64.

[6] KUROSAWA K, DESMEDT Y. Optimum Traitor Tracing and Asymmetric Schemes[A]. Proceedings of Eurocrypt98[C]. Berlin: Springer-Verlag, 1998. 145 - 157.

[7] KIAYIAS A, YUNG M. Breaking and Repairing Asymmetric Public-Key Traitor tracing[A]. ACM Workshop on DRM2002[C]. Berlin: Springer-Verlag, 2003. 32 - 50.

[8] 李勇, 杨波. 一种高效非对称的动态公钥叛逆者追踪方案[J]. 西安电子科技大学学报(自然科学版), 2003, 30(3): 394 - 398.

[9] NAOR M, PINKAS B. Oblivious Transfer and Polynomial Evaluation [A]. Proceedings of STOC'99[C]. Atlanta: ACM, 1999. 245 - 254.

(上接第 1809 页)

在解决数字版权保护方面要建立一个信任的机制,这要求系统的基础架构能够允许提供者与用户保持相互的信任,控制策略最终应由双方共同达成,维护双方的权益。

在解决隐私方面提出了 Properly-based Attestation 协议和采用组签名策略,为不暴露平台的根秘密性,提出如零知识证明,组密钥分配等相关的认证技术,使可信计算平台能在不暴露用户隐私的前提下完成认证过程,达到保护用户隐私的目的。

在对用户的使用局限性方面,系统的控制策略应当能够由用户定制,使得用户对系统有更多的控制权利。

未来可信计算平台的研究将主要集中在以下几个方向:1)以工业化方式从计算平台体系结构上解决安全性问题;2)增强计算平台的可信性和为应用层安全创建更可靠的安全根基;3)在网络环境中建立有效的信任关系,并对这种信任关系进行有效的管理。

参考文献:

[1] TPM Main Part1 Design Principles Specification Version 1.2 52 Draft[Z]. 2003.

[2] TPM Main Part2 TPM Structures Specification Version 1.2 57 Draft [Z]. 2003.

[3] TPM Main Part3 Commands Specification Version 1.2 Revision 57 Draft[Z]. 2003.

[4] TPM Specification Part4 TPM Conformance Specification Version 1.2 Draft[Z]. 2003.

[5] TCG Software Stack Specification Version 1.10 RC 10A[Z]. 2003.

[6] TCG Infrastructure Committee Reference Architecture for Integrity Information Interoperability, revision 0.07 Draft[Z]. 2004.

[7] TCG TNC Architecture for Interoperability Specification Ver 1.0 0.16 Draft[Z]. 2004.

[8] IBM Research Report, the role of TPM in enterprise security[Z]. 2004.

[9] TCG Infrastructure Working Group. Use Cases Summary. Draft. Version 0.1[EB/OL]. https://www.Trusted-computinggroup.org/downloads, 2004 - 03 - 07.