中图分类号:TP309　　文献标识码: A　　文章编号: 1009- 2552(2006) 12- 0121- 04

# 基于 Mignotte's 列的( $t$, $n$) 门限群签名方案

邹时华, 曾吉文, 全俊杰

(厦门大学数学科学学院, 厦门 361005)

摘　要: 电子学报中曾提出了一个在不改变其他有效群成员的签名密钥的情况下, 可安全快速地加入或删除群成员的群签名方案。但此方案没有涉及到实际运用中需要设置门限的情况, 现对原方案做了改进和推广, 保留了原方案中可安全快速地加入或删除群成员的优点, 推广到了每次参与签名人数变动较大, 并需要设置门限的情况, 利用中国剩余定理提出了一个新的基于 Mignotte's 门限秘密共享方案的(t, n) 门限群签名方案。

关键词: Mignotte's 门限秘密共享方案; 中国剩余定理; 门限群签名方案

# A( t , n) threshold group signature scheme based on Mignotte's sequence

ZOU Shi- hua, ZENG Ji- wen, QUAN Jun- jie

(**School of Mathematics, Xiamen University, Xiamen** 361005, **China**)

**Abstract**: A group signature scheme that can enlist in and delete a group member safely and quickly without changing the secret keys of other available group members has been proposed in Electronic Journal. The scheme did not deal with the situation of setting threshold. This paper presents a new threshold group signature scheme by extending the Chen Ze- wen's construction . The scheme based on Mignotte's sequence makes use of the Chinese remainder theorem.

**Key words**: Mignotte's threshold secret sharing scheme; Chinese remainder theorem; threshold group signature

## 0　引言

在一个较大的群体如一个大公司中, 对某份文件 $m$, 有一部分人同意此文件并合作对 $m$ 签名, 对另一份文件 $m'$, 则是另外一部分人同意, 构成一个新的签名群体, 与前一部分人变动可能较大。如果规定在签署文件时, 必须超过特定人数才算签名有效, 则需要一个门限群签名方案。

群签名是 D. chaun 和 E. van. Heyst 于 1991 年提出的, 群签名方案中, 群中任意成员可以代表整个群体对消息 $m$ 签名, 在有争议时, 可通过群管理者确定真实签名者的身份。Desmedt 和 Frankel 首次提出 $(t, n)$ 门限群签名, $(t, n)$ 门限群签名是任意 $t$ 个或更多个成员合作生成代表群的有效签名体制。自从提出后, 门限群签名得到广泛研究, 提出了各种各样的门限群签名方案, 这些门限群签名方案大多基于秘密共享方案, 如 Shamir 的基于多项式插值的门限秘密共享方案, Blakley 的基于几何的门限秘密共享方案等。这些门限群签名方案中, 群密钥是群中任意 $t$ 个或更多个成员共享的秘密, 这样群中 $t$ 个或更多个成员合谋就可以恢复秘密多项式, 从而获得群密钥。借助公开的身份, 他们可以恢复群中所有成员的密钥, 从而任何一组人可以合谋假冒另一组成员产生代表群的签名和伪造身份追查方程。本文提出一个即使其他成员合谋得到门限秘密也无法冒充有效签名者签名的方案, 因为恢复的门限秘密首先必须得到中心的验证, 而验证时需要用到有效签名者的私钥, 其他无效签名者无法得到实际签名者的私钥从而不能得到中心的验证。而且每一轮的群公钥随着得到验证的签名者不同而变化, 避免了

收稿日期: 2006- 07- 10

作者简介: 邹时华(1976- ), 男, 厦门大学硕士研究生, 研究方向为代数在密码学中的应用。

未参加验证的无效签名者利用群公钥伪造签名。本文提出的基于中国剩余定理的$(t, n)$门限群签名方案利用了Mignotte's列构造门限秘密共享方案, $n$ 个人的群体中超过 $t$ 个人可以恢复门限秘密, 利用所有参与者的私钥对门限秘密加密后发送到中心接受验证, 中心验证后利用原来保存的参与者的私钥所对应的秘密信息进行计算, 得到群公钥, 将其发布。有效签名群体利用私钥对消息签名。由于群公钥的计算只与有效签名者有关, 所以其他无效签名者的伪造签名无法得到验证者的验证。文中所提出的方案的安全性基于大数分解的困难性。

## 1 预备知识

### 1.1 群签名与门限群签名的概念及安全性要求

群签名方案是一个包含以下过程: 建立系统, 加入群成员, 撤消群成员, 签名, 验证, 打开等算法的一个数字签名方案。其安全性要求: ①匿名性: 给定群签名, 除群管理者外, 任何人确定真实签名者身份在计算上是困难的。 ②防伪造性: 只有合法群成员才能生成2有效群签名。 ③可跟踪性: 在出现争议时, 群管理者可打开签名确定签名者身份, 且签名者无法阻止。 ④防陷害攻击: 不能冒充其他成员签名。 ⑤抗合谋攻击: 一些成员串通在一起也不能产生一个合法且不能被追踪的签名。 ⑥不可关联性: 除群管理者外, 任何人想判断两个签名是否由同一个人签署是困难的。$(t, n)$门限群签名方案是指任意 $t$ 个或更多个成员合作才能生成代表群的有效签名体制, 要求具有群签名的特性和门限特性。一个性能良好的门限群签名还应该具有的特点有: 签名和验证的简单性与匿名性, 身份的可追查性, 系统的强壮性与稳定性。

### 1.2 中国剩余定理

设 $m_1, m_2, \cdots, m_k$ 是两两互素的 $k$ 个正整数, $k \geq 2$, 则同余方程组

$$\begin{cases} x = a_1 \bmod m_1 \\ x = a_2 \bmod m_2 \\ \cdots\cdots \\ x = a_k \bmod m_k \end{cases}$$

有模 $M = m_1 m_1 \cdots m_1$ 的惟一解 $x = \sum_{i=1}^{k} a_i M_i (M_i^{-1} \bmod m_i) \pmod M$ 其中 $M_i = M/m_i$。

### 1.3 Mignotte's-$(t, n)$门限秘密共享方案

①设 $n$ 是一个正整数, $n \geq 2, 2 \leq t \leq n$, 若 $m_1 < m_2 < \cdots < m_n$, 且$(m_i, m_j) = 1$对所有 $1 \leq i < j \leq n$ 成立, 当 $m_{n-t+2} m_{n-t+1} \cdots m_n < m_1 m_2 \cdots m_t$ 时, 整数列 $m_1, m_2, \cdots, m_n$ 称为$(t, n)$-Mignotte列。

②给定一个$(t, n)$-Mignotte列, 秘密共享方案运行如下: 任意选择秘密 $S$, 使得 $\beta < S < \alpha$, 其中 $\beta = m_{n-t+2} m_{n-t+1} \cdots m_n$; $\alpha = m_1 m_2 \cdots m_t$。计算分享 $I_i$:

$$I_i = S \bmod m_i \quad 1 \leq i \leq n$$

将 $n$ 个分享$I_i$ 秘密发送给$n$ 个用户。若 $n$ 个用户中有超过 $t$ 个用户想重构秘密$S$, 不妨设 $k$ 个, $k \geq t$, 设 $k$ 个用户为 $U_{i1}, U_{i2}, \cdots, U_{ik}$, 将他们的分享 $I_{i1}, I_{i2}, \cdots, I_{ik}$ 利用同余方程组

$$\begin{cases} x = I_{i1} \bmod m_{i1} \\ x = I_{i2} \bmod m_{i2} \\ \cdots\cdots \\ x = I_{ik} \bmod m_{ik} \end{cases}$$

可以重构秘密 $S$, 而少于 $t$ 个人无法重构秘密$S$, 甚至无法得到关于 $S$ 的任何信息。也可考虑用同样基于中国剩余定理的Asmuth-Bloom秘密共享方案。

### 1.4 新的基于Mignotte's列的门限群签名方案

#### 1.4.1 系统建立

群中心选择两个大的素数$p$ 和$q$, 计算$pq = n$, 选择一个单向Hash函数 $h$, 任意选择 $e \in Z_n$; 计算 $d$, 使得 $ed \equiv 1 \bmod \Phi(n)$。将 $e$ 作为中心公钥公布, $d$ 作为中心私钥保存。随机选择 $x_i, y_i \in Z_n$ 使 $x_i y_i \equiv 1 \bmod \Phi(n)$。选择素数 $m_i > y_i$ 使得$m_1, m_2, \cdots, m_n$ 构成$(t, n)$-Mignotte列。即满足

$$\begin{cases} m_1 < m_2 < \cdots < m_n \\ (m_i, m_j) = 1 (1 \leq i < j \leq n) \\ m_{n-t+2} m_{n-t+1} \cdots m_n < m_1 m_2 \cdots m_t \end{cases}$$

由于 $x_i, y_i \in Z_n$ 的选择是随机的, 必要时可适当地交换 $x_i$ 和 $y_i$ 的顺序使以上条件成立。将$(x_i, m_i, m_i^d)$ 秘密发送给用户 $U_i$, 用户 $U_i$ 验证$m_i = (m_i^d)^e \pmod n$ 是否成立, 若成立, 相信是群中心发送的, 将$(x_i, m_i, m_i^d)$ 作为签名密钥保存。

中心计算 $t$ 个和$t$ 个以上的$m_i$ 的乘积 $\prod_{j \geq t} m_{ij}$, 这样的乘积共 $C_n^t + C_n^{t+1} + \cdots + C_n^n$ 个, 且互不相同, 这是因为 $m_1, m_2, \cdots, m_n$ 两两互素且互不相等。设 $C_n^t + C_n^{t+1} + \cdots + C_n^n = L$, 将这 $L$ 个乘积的集合记为 $\{M_1, M_2, \cdots M_L\}$, 并记录下每个所对应的用户及。

#### 1.4.2 门限秘密分享及其验证

设 $n$ 个人的可能签名群体中, 规定超过 $t$ 个人合作生成的签名为有效签名, 即一个$(t, n)$ 门限群

签名。

(1) 计算并分发分享

假设共进行 $m$ 轮签名, 中心随机选择任意 $S_1$, $S_2$, …, $S_m$, 使得 $\beta < S_j < \alpha, j = 1, 2, \cdots, m$, $S_j$ 是第 $j$ 轮签名时的门限秘密。开始计算每个用户 $U_i$ 的分享 $I_{ij} = S_j \bmod m_i$, $(j = 1, 2, \cdots, m)$ 将 $m$ 元对 $(I_{i1}, I_{i2}, \cdots, I_{im})$ 秘密发送给用户 $U_i$, $(i = 1, 2, \cdots, n)$。

(2) 第一轮签名时, 重构秘密

若 $n$ 个用户中有超过 $t$ 个用户请求签名, 不妨设 $k(k \geq t)$ 个, 这 $k$ 个用户分别为 $U_{i1}$, $U_{i2}$, …, $U_{ik}$, 所有这 $k$ 个用户将自己的第一轮的分享 $I_{i1,1}$, $I_{i2,1}$, …, $I_{ik,1}$, 和对应的 $m_{i1}^d$, $m_{i2}^d$, …, $m_{ik}^d$ 秘密发送给指定的群秘书, 群秘书利用群公钥 $e$ 计算 $(m_{i1}^d)^e \bmod n$, $(m_{i2}^d)^e \bmod n$, …, $(m_{ik}^d)^e \bmod n$, 恢复出 $m_{i1}$, $m_{i2}$, …, $m_{ik}$, 再解同余方程组

$$\begin{cases} x = I_{i1} \bmod m_{i1} \\ x = I_{i2} \bmod m_{i2} \\ \cdots\cdots \\ x = I_{ik} \bmod m_{ik} \end{cases}$$

群秘书利用中国剩余定理及 Mignotte' s 门限秘密共享方案知超过 $t$ 个人就可以重构秘密 $S_1$, 这是由于同余方程组有模 $m_{i1}, m_{i2}, \cdots, m_{ik}$ 的惟一解, 且由 $(t, n)$ - mignotte 列的性质可知此解就是 $S_1$, 而少于 $t$ 个人无法重构秘密。群秘书将乘积 $m_{i1}$, $m_{i2}$, …, $m_{ik}$ 发送给群中心, 群中心利用保存的信息恢复 $m_{i1}, m_{i2}, \cdots, m_{ik}$, 找出对应的 $y_{i1}, y_{i2}, \cdots, y_{ik}$ 计算

$$c_1 = \sum_{j=1}^{k} y_{ij} M_{ij} (M_{ij}^{-1} \bmod m_{ij}) (\bmod M)$$

其中 $M = m_{i1} m_{i2} \cdots m_{ik}$, $M_{ij} = M / m_{ij}, j = 1, 2, \cdots, k$, 群中心将 $c_1$ 发送给群秘书, 群秘书将 $(S_1, c_1)$ 秘密发送给群中的每位成员。

(3) 门限秘密验证

某位用户不妨设 $U_{i1}$ 利用自己的私钥 $x_{i1}$ 对秘密 $S_1$ 加密, 计算出 $S_1^1 = S_1^{x_{i1}} \bmod n$, 再将 $(S_1^1, S_1, m_{i1}^d)$ 发送给用户 $U_{i2}$, $U_{i2}$ 先利用公钥 $n$, $e$, 和 $c_1$ 进行验证, 计算 $(m_{i1}^d)^e \bmod n = m_{i1}$ 和 $c_1 \bmod m_{i1} = y_{i1}$, 再验证 $(S_1^1)^{y_{i1}} \bmod n = S_1$ 是否成立, 若成立则确信由 $U_{i1}$ 发送而来, 继续计算: $S_1^2 = (S_1^1)^{x_{i2}} \bmod n$, 将 $(S_1^1, S_1^2, m_{i1}^d, m_{i2}^d)$ 发送给用户 $U_{i3}$。$U_{i3}$ 先验证再计算, 然后再发送 $(S_1^2, S_1^3, m_{i1}^d, m_{i2}^d, m_{i3}^d)$ 给用户 $U_{i4}$, …, 继续此过程直到得到 $S_1^k$, 最后一个用户将 $S_1^k$ 发送给群秘书, 群秘书将 $(S_1^k, m_{i1}, m_{i2}, \cdots, m_{ik})$ 发送给群中心进行验证。群中心利用保存的 $\{M_1, M_1, \cdots, M_L\}$ 与 $\sum_{j=1}^{k} m_{ij}$ 相对照, 判断是否与集合中的某个 $M_i$, $i \in \{1, 2, \cdots, L\}$ 相等, 若不存在, 停止后续步骤。若存在, 利用保存的信息恢复 $m_{i1}, m_{i2}, \cdots, m_{ik}$, 找出对应的 $y_{i1}, y_{i2}, \cdots, y_{ik}$ 再验证

$$(S_1^k)^{y_{i1}, y_{i2}, \cdots, y_{ik}} = S_1$$

是否成立, 若成立, 群中心确认有 $k(k \geq t)$ 个用户, 即有超过门限 $t$ 个用户要合作生成群签名, 群中心将 $c_1$ 公布, 任何公开验证者可利用 $(n, e, c_1)$ 对第一轮的签名进行验证。

(4) 签名过程

对消息 $m$, 上述的 $k$ 元用户集 $\{U_{i1}, U_{i2}, \cdots, U_{ik}\}$ 中的一个或多个合作都可以生成代表这个群体对消息 $m$ 的签名, 不失一般性, 设有 $l(1 \leq l \leq k)$ 个用户且为 $U_{i1}, U_{i2}, \cdots, U_{il}$, 利用单向 Hash 函数 $h$ 计算, 当 $l = 1$ 时, 一个用户不失一般性设为 $U_{i1}$, 计算:

$$\mathrm{sig}(m, x_{i1}) = h(m)^{x_{i1}} \bmod n$$

将签名 $(m, \mathrm{sig}(m, x_{i1}), m_{i1}^d)$ 发送给验证者 verifier。当 $l > 1$ 时, 签名过程的合作类似验证门限秘密时的合作, 计算出:

$$\mathrm{sig}(m, x_{i1}, x_{i2}, \cdots, x_{il}) = h(m)^{x_{i1}, x_{i2}, \cdots, x_{il}} \bmod n$$

将签名 $(m, \mathrm{sig}(m, x_{i1}, x_{i2}, \cdots, x_{il}), m_{i1}^d, m_{i2}^d, \cdots, m_{il}^d)$ 发送给验证者 verifier。

(5) 签名验证

验证者 verifier 利用公钥 $(n, e, c_1)$, 计算:

$$(m_{ij}^d)^e \bmod n = m_{ij}, j = 1, 2, \cdots, l$$

$$c_1 \bmod m_{ij} = y_{ij}, j = 1, 2, \cdots, l$$

验证 $(\mathrm{sig}(m, x_{i1}, x_{i2}, \cdots, x_{il}))^{y_{i1}, y_{i2}, \cdots, y_{il}} \bmod n = h(m)$ 是否成立。若成立, 作为有效签名接收。

(6) 签名打开

若出现争议时, 群中心计算:

$$(m_{ij}^d)^e \bmod n = m_{ij}, j = 1, 2, \cdots, l$$

$$c_1 \bmod m_{ij} = y_{ij}, j = 1, 2, \cdots, l$$

利用保存的信息, 将 $y_{ij}$ 所对应的用户 $U_{ij}$ 身份给出。

第二轮签名时, 请求对另外一则消息签名的用户与前一轮签名的用户变动可能较大, 新的达到门限人数的签名群体可以利用第二轮的分享重构第二轮的门限秘密 $S2$, 类似前一轮签名, 在得到中心的验证后, 群中心发布新的群公钥 $c_2$, 新的签名群体中的一个或多个可以生成代表这个群体的签名, 验

证者使用新公钥$(n, e, c_2)$对签名进行验证。以下每轮都类似进行。

### 1.5 签名方案的门限特性与安全性分析

#### 1.5.1 门限特性

首先少于$t$个人的集合由Mignotte's列的性质可得无法重构秘密$S$,从而无法得到中心的确认,中心不会发布与这些人有关的群公钥$c$。即使这些人从其他渠道得到$S$,例如当前轮中一个有效的不少于$t$的用户集重构了秘密$S$,其中的某个用户泄露了秘密$S$,或者与某个有效用户集的指定秘书相勾结得到$S$,设这些用户为$U_{i1}, U_{i2}, \cdots, U_{iw}, 1 \leq w < t$,利用私钥$x_{i1}, x_{i2}, \cdots, x_{iw}, 1 \leq w < t$计算并发送$(S^{x_{i1}, x_{i2}, \cdots, x_{iw}} \bmod n, m_{i1}, m_{i2}, \cdots, m_{iw})$到中心由于中心可判断出乘积$m_{i1}, m_{i2}, \cdots, m_{iw}$在保存的$\{M_1, M_1, \cdots, M_L\}$中无记录,从而拒绝计算群公钥$c$。

若用户$U_{i1}, U_{i2}, \cdots, U_{iw}, 1 \leq w < t$利用非法获得的$S$,试图伪造一个能让中心验证通过的信息,例如构造

$$(S^{x_{i1}, x_{i2}, \cdots, x_{iw} x_i, w+1, \cdots, x_{iv}} \bmod n, m_{i1} m_{i2}, \cdots, m_{iw} m_{i, w+1}, \cdots, m_{iv})$$

其中$t \leq v \leq n$但要获得$x_{i, w+1}, \cdots, x_{iv}$这些私钥在计算上是困难的。所以少于$t$个人的用户集合是无法得到中心的验证的。

#### 1.5.2 签名方案的安全性分析

本文的安全性分析分为以下三种可能攻击:得到用户的签名密钥、伪造群成员的签名、联合攻击。

(1) 若敌手Oscar截获某个有效签名$(m, \mathrm{sig}(m, x_{i1}), m_{i1}^d)$(为简单起见,只考虑一个用户的签名),Oscar试图获得签名密钥,可如下计算:

$$(m_{i1}^d)^e \bmod n = m_{i1}$$

$$c_1 \bmod m_{i1} = y_{i1}$$

要获得签名私钥$x_{i1}$,可试图解方程$x_{i1} y_{i1} \equiv 1 (\bmod \Phi(n))$,解出此方程须先知道$\Phi(n)$的值,而计算出$\Phi(n)$并不比分解$n$容易,由系统假设知分解$n$在计算上是困难的,所以获得签名密钥在计算上是困难的。

(2) 伪造群成员的签名,分三种情况:①Oscar以前从未参加过签名,现在也不是有效群成员。②Oscar以前参加过某轮签名,现在不是有效群成员。③Oscar现在是有效群成员中的一个,但想冒充另一个有效群成员签名。

情况①Oscar以前从未参加过签名,现在也不是有效群成员。Oscar想伪造签名,不仅要知道本轮的秘密$S$,还要能获得$m_{i1}^d, m_{i2}^d, \cdots, m_{il}^d, x_{i1}, x_{i2}, \cdots, x_{il}$其中得到签名私钥$x_{i1}, x_{i2}, \cdots, x_{il}$在计算上是困难的,由于选取的单向Hash函数$h$的性质,Oscar利用截获的某个有效签名进行存在性伪造也无法成功。

情况②Oscar以前参加过某轮签名,现在不是有效群成员。Oscar利用以前的某轮签名,如$(m, \mathrm{sig}(m, x_{i1}), m_{i1}^d)$,此时有两种可能,第一种,这个签名是Oscar自己签署的,即$x_{i1}$就是Oscar的私钥,由于现在Oscar不是有效群成员,用自己的私钥签名不再能得到验证,所以只能伪造有效群成员的签名。若这个签名是现在某个有效成员以前签署的,Oscar会试图利用$(m, h(m)^{x_{i1}} \bmod n, m_{i1}^d)$去伪造现在的签名,由于无法获得私钥$x_{i1}$,只能保持$h(m)^{x_{i1}} \bmod n$不变,要想获得验证者的验证,只能保持$m$和$h(m)^{x_{i1}} \bmod n$不变。若Oscar利用一个有效群成员以前的两个不同签名,例如$(m_1, h(m_1)^{x_{i1}} \bmod n, m_{i1}^d)$和$(m_2, h(m_2)^{x_{i1}} \bmod n, m_{i1}^d)$,Oscar会试图计算

$$h(m_1)^{x_{i1}} h(m_2)^{x_{i1}} = [h(m_1) h(m_2)]^{x_{i1}}$$

但要由$h(m_1) h(m_2) = h(m_3)$求出$m_3$是不可能的,因为$h$是一个选取的单向即原像稳固的Hash函数。

情况③Oscar现在是有效群成员中的一个,但想冒充另一个有效群成员签名。Oscar现在只知道自己的签名密钥,也可以获得另一个用户$U_j$的签名密钥$(y_j, m_j, m_j^d)$,类似前面的分析,Oscar想获得$U_j$的私钥$x_j$是计算上困难的。

(3) 联合攻击。即使一些成员和群秘书联合在一起,想伪造用户$U_j$的签名,他们必须获得$U_j$的签名密钥$(x_j, y_j, m_j, m_j^d)$,同前两种情况一样,他们可以获得$U_j$的签名密钥$(y_j, m_j, m_j^d)$,但由于不知道$n$的分解,且分解的困难性足以抵抗联合攻击,所以这些人即使联合起来也无法得到私钥$x_j$,从而无法伪造$U_j$的合法签名。

## 2 结束语

本文基于Migonette列,利用中国剩余定理提出了一个新的门限群签名方案,其安全性基于大数分解的困难性。具有以下几个优点:①每一轮签名时的群公钥随实际签名者的变化而变化,避免了以前参加过签名的无效签名者伪造签名。②即使群成员变动较大,群公钥的重新计算简单快速。③在所有成员的私钥不用改变的情况下,可安全 (下转第178页)

数据预处理模块。为了验证问题, 选取了“法轮功”类反动信息作为系统数据的来源, 目的主要是验证系统学习模块对特征字段的统计效果, 及系统能否对不良信息进行准确的智能识别。实验结果由两部分组成: 一是统计结果; 二是分析结果。

4.1 统计结果:

统计文本个数:

正面类文章数: 207 篇

反面类文章数: 228 篇

统计特征字段情况:

正面类特征字段个数: 80 个

反面类特征字段个数: 40 个

特征字段举例:

正面特征字段举例: 反人类、反社会、反科学
精神控制
李洪志

反面特征字段举例: 迫害大法弟子
迫害致死
炼功

4.2 分析结果

对用于统计的正、反共 435 篇文章分析后, 有 9 篇反面文章未判断出来, 其中有两篇是关于法轮功的文章, 其余 7 篇与法轮功无关。

本文又用人民日报的 140 篇文章做测试, 发现结果都正确。

## 5 结束语

实验证明经改进的分词算法, 能够有效的识别出被符号分开的特征字段, 并且在一定程度上消除了分词歧义。在机器学习和识别过程中系统能够将信息分为正反两类, 体现了智能的特点。

参 考 文 献:


[1] 王科, 高常波, 翟雪峰, 等. 汉语分词主要技术及其应用[J]. 通信技术, 2003(6): 12- 15.

[2] 张庭, 等. 贝叶斯统计推断[M]. 科学出版社, 1991.

[3] 陈华辉, 薛春阳. 一种基于贝叶斯网的“垃圾”邮件过滤器[J]. 微机发展, 2000(4).


责任编辑: 么丽萍

(上接第 124 页)完成多次门限群签名。④门限秘密不再是签名密钥, 避免了达到门限人数的群体非法获得他人签名密钥的可能性。还可以考虑采取时间戳协议的方式来验证一个有效群成员以前的签名是否被攻击者实施重放攻击, 即通过时间戳协议, 群中心可以判断一个签名是否是本轮的有效签名。本方案的签名和验证、打开等过程的计算量较小, 但门限秘密的分享与验证过程计算量较大, 而且每一轮签名前都要进行门限验证, 因此本方案适用于对门限特性要求较高, 而且每一轮签名时的实际参与者变动较大的情形。

参 考 文 献:


[1] Sorin Iftene. Compartmented Secret Sharing Based on the Chinese Remainder Theorem[EB/OL]. http://eprint.iacr.org/complete/.

[2] Asmuth C A, Bloom J. A modular approach to key safeguarding[J]. IEEE Transactions on Information Theory, IT- 1983, 29(2): 208- 210.

[3] 陈泽文, 张龙军, 王育民, 等. 一种基于中国剩余定理的群签名方案[J]. 电子学报, 2004, 32(7).

[4] 谢琪, 于秀源. 基于分组秘密共享的(t, n) 门限群签名体制[J]. 计算机学报, 2005(2).

[5] Ding C, Pei D, Salomaa A. Chinese remainder theorem, applications in coding and cryptography[M]. World Scienti- c Publishing Co., Inc., 1996.

[6] Mignotte M. How to share a secret[J]. Cryptography- proceedings of the Workshop on Cryptography, Burg Feuerstein.

[7] 王贵林, 卿斯汉. 几个门限群签名方案的弱点[J]. 软件学报, 2000, 11(10): 1324- 1332.


责任编辑: 么丽萍

(上接第 165 页)

```
SetTextColor(hScrDC, ftext.color);
//设置背景
SetBkMode(hScrDC, TRANSPARENT);
//显示消息
DrawText(hScrDC, ftext.text, ftext.textcount, &rect,
DT_CENTER);
```

## 4 结束语

系统主要实现图像的压缩传输和实时控制, 文件的创建和删除操作, 显示被控制端的主机信息, 可以发送实时消息。应用本系统可以控制对方的计算机, 进行管理和维护。计算机远程控制系统可以方便用户完成诸如排除计算机故障、维护服务器、文件传输、异地办公等功能, 计算机远程控制将会得到更深入的应用。

参 考 文 献:


[1] 周明天, 汪文勇. TCP/IP 网络原理与技术[M]. 北京: 清华大学出版社, 1993.

[2] 黄维通. Visual C++ 面向对象与可视化程序设计[M]. 北京: 清华大学出版社, 2000.

[3] David J, Knuglinski. Visual C++ 内幕[M]. 4 版. 潘爱民, 等译. 北京: 清华大学出版社, 1999.

[4] 陈明. 软件工程学教程[M]. 北京: 科技出版社, 2002.


责任编辑: 么丽萍