无可信中心的可公开验证多秘密共享

于 佳[1),2),3)] 陈养奎[1)] 郝 蓉[1)] 孔凡玉[4),5)] 程相国[1)] 潘振宽[1)]

[1)](青岛大学信息工程学院 山东 青岛 266071)

[2)](山东省科学院山东省计算机网络重点实验室 济南 250014)

[3)](中国科学院信息工程研究所信息安全国家重点实验室 北京 100093)

[4)](山东大学网络信息安全研究所 济南 250100)

[5)](山东大学密码技术与信息安全教育部重点实验室 济南 250100)

摘 要 多秘密共享是通过一次计算过程就可以实现同时对多个秘密进行共享的密码体制,在一般的多秘密共享中,都需要可信中心的参与,由可信中心进行秘密份额的分发.然而,在很多情况下,无法保证可信中心的存在,即使存在可信中心,它也很容易遭受敌手的攻击,成为系统的盲点.该文提出了一个无可信中心的可公开验证多个秘密共享方案,共享的多个随机秘密是由参与成员共同产生的,密钥份额的有效性不仅可以被份额持有者自己验证,而且可以被其他任何成员验证,这使方案具有更广的应用背景,可用于设计电子投票协议、密钥托管协议等.为了适用于无线自组网等新的网络环境,该文也讨论了无可信中心的条件下动态撤出和增加成员的问题.

关键词 秘密共享;多秘密共享;可信中心;可公开验证方案;安全性;网络安全;信息安全

中图法分类号 TP309 **DOI 号** 10.3724/SP.J.1016.2014.01030

Publicly Verifiable Multi-Secret Sharing Without Trusted Centers

YU Jia[1),2),3)] CHEN Yang-Kui[1)] HAO Rong[1)] KONG Fan-Yu[4),5)]
CHENG Xiang-Guo[1)] PAN Zhen-Kuan[1)]

[1)](*College of Information Engineering, Qingdao University, Qingdao, Shandong* 266071)

[2)](*Shandong provincial Key Laboratory of Computer Network, Shandong Academy of Sciences, Jinan* 250014)

[3)](*State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093)

[4)](*Institute of Network Security, Shandong University, Jinan* 250100)

[5)](*Key Laboratory of Cryptographic Technology and Information Security, Ministry of Education, Shandong University, Jinan* 250100)

Abstract A multi-secret sharing scheme is a cryptographic scheme in which multiple secrets can be simultaneously shared during one computational process. However, in a normal multi-secret sharing scheme, we need the participation of a trusted center, which completes the distribution of secret shares. Sometimes, however, we cannot guarantee the existence of a trusted center. Even if there is a trusted center, it is easily targeted for an adversary and become a blind spot of the system. A publicly verifiable multi-secret sharing scheme without trusted centers is proposed in this paper. Shared multiple secrets are jointly generated by the participations. The validity of shares can be verified not only by shareholders themselves but also by any other members. Therefore, this scheme has wider application prospects, such as the designs of electronic voting protocol, key escrow protocol etc. In order to make the scheme adapt to new network circumstance

收稿日期:2012-10-25;最终修改稿收到日期:2014-03-02.本课题得到国家自然科学基金(61272425,61303197,61202475)、青岛市科技计划项目(12-1-4-2-(16)-jch,12-1-4-2-(14)-jch,13-1-4-151-jch)、华为科技基金(YB2013120027)、信息安全国家重点实验室开放课题项目、山东省计算机网络重点实验室开放课题项目(SDKLCN-2013-03)资助.**于 佳**,男,1976 年生,博士,教授,中国计算机学会(CCF)高级会员,主要研究领域为信息安全、密码学.E-mail:yujia@qdu.edu.cn;qduyujia@gmail.com.**陈养奎**,男,1984 年生,硕士研究生,主要研究方向为信息安全.**郝 蓉**,女,1976 年生,硕士,实验师,主要研究方向为信息安全、密码学.**孔凡玉**,男,1978 年生,博士,副教授,主要研究方向为信息安全、密码学.**程相国**,男,1969 年生,博士,副教授,主要研究方向为信息安全、密码学.**潘振宽**,男,1966 年生,博士,教授,主要研究领域为图像处理、信息安全.

such as ad hoc networks, we also discuss the problem of how to dynamically withdraw and add members without trusted centers in this scheme.

Keywords　secret sharing; multi-secret sharing; trusted center; publicly verifiable scheme; security; network security; information security

1 引　言

为了确保秘密的安全性和有效性，Shamir[1]和Blakley[2]于1979年分别基于拉格朗日插值多项式和射影几何理论最先提出了门限秘密共享方案. 在一个(t,n)的秘密共享方案中，一个秘密共享在n个服务器中，其中的任意t个服务器都可以合作重构出秘密，而少于t个的服务器均不能正确地重构出秘密. 然而，这两个秘密共享方案并不能保证分发者和份额持有者的诚实性问题. 文献[3-4]提出了两个著名的可验证秘密共享方案，解决了上述问题. 文献[5-6]则提出了两种具有公开验证能力的秘密共享方案，分发者分发的秘密份额的正确性不仅能被份额持有者自己验证，而且可以被其他任何成员验证. Gennaro等人[7]提出了一个无可信中心的可验证秘密共享方案，在没有可信中心的情况下，可以实现秘密共享的目的.

上述的所有方案每次只能共享一个秘密，有时候我们需要同时保护多个秘密，或者把一个比较大的秘密分成多个子秘密，通过保护子秘密来保护秘密，这样就需要一个方案能够实现一次共享多个秘密的目的. 为此，多秘密共享的方法被提出来. 在一个多秘密共享方案中可以同时共享多个秘密，门限个份额持有者合作可以重构出所有秘密. He和Dawson[8]基于单向函数提出了一个实用的多秘密共享方案. Yang等人[9]基于Shamir的秘密共享体制[1]提出了一个多秘密共享方案，方案的运算效率较高，但不具有可验证能力. 文献[10-14]提出了具有可验证功能的多秘密共享方案，可避免因份额分发者和持有者的欺骗而导致不能重构出正确秘密的问题.

观察目前已有的多秘密共享方案，存在下面两个问题需要进一步解决. 首先，多秘密共享方案中都需要一个可信中心负责分发份额，这在很多应用中是不现实并难以实现的，比如在无线传感网和无线自组网中. 因此，提出无可信中心的多秘密共享方案具有重要的应用背景. 其次，目前已有的方案多数不具备可公开验证的属性，即份额持有者所得到份额的正确性只能由自己验证，其他成员无法验证它是否正确. 由于秘密共享的许多应用(比如电子投票、密钥托管、电子支付协议等)都需要可公开验证的性质，因此，构造具有可公开验证性质的多秘密共享方案是一项非常有意义的研究工作.

为了解决上述这两个问题，本文提出了一个无可信中心的可公开验证多秘密共享方案. 方案基于同态结构，与使用单向函数(或双变量单向函数)的多秘密共享不同，具有同态性质的方案可以较容易的实现联合生成秘密的目标. 为了使联合生成的份额能被每个成员验证，利用可公开验证的方法进行秘密分发，任意成员都可以对所有份额进行验证. 为了能够一次性重构出多个秘密，文中给出了一个多秘密求解算法，多于一定数目的成员通过该求解算法，可以计算出联合生成共享的多个秘密. 此方案也可方便地实现成员的撤出和加入.

2 预备知识

2.1 定　义

定义1.　无可信中心的可公开验证多秘密共享由两个阶段组成：秘密产生阶段和秘密恢复阶段.

(1) 秘密产生阶段. 当参与者要生成m个随机秘密时，每个参与者P_i都分别为所有参与者$P_j(j=1,\cdots,n)$产生随机秘密份额s_{ij}，使用加密函数E_i生成加密的$S_{ij}=E_i(s_{ij})$，并将$S_{i1},\cdots,S_{in}$公开. 任何成员都可验证S_{ij}是否是P_i为P_j生成的s_{ij}的正确加密形式，成员P_j可以通过解密函数获取份额s_{ij}. P_j可以通过$s_{ij}(i=1,\cdots,n)$最终计算获得份额s_j.

(2) 秘密恢复阶段. 当参与者要重构秘密时，P_j提供份额s_j并证明它是一个正确的解密形式. 运行重构函数计算最终共享的秘密$k_0,k_1,\cdots,k_{m-1}$.

2.2 所需工具

(1) Shamir的(t,n)秘密共享方案[1]

令q为大素数，假定要分发的秘密为$s\in Z_q$，分发者选择随机多项式$f(x)=\sum_{j=0}^{t-1}a_j x^j \pmod q$，这里$a_0=s$，$a_j\in_R Z_q(j=1,\cdots,t-1)$. 计算份额$s_i=f(i)(i=1,\cdots,n)$，并将$s_i$秘密地发送给成员$P_i$.

秘密重构时，任意 t 个成员的集合 B 拿出他们的份额 s_i，计算 $s=\sum_{P_i\in B}C_{Bi}s_i \pmod q$，其中插值系数 $C_{Bi}=\prod_{P_j\in B\setminus\{P_i\}}\frac{j}{j-i}\pmod q$.

(2) 非交互的离散对数相等的零知识证明协议 $DLE(g_1,h_1;g_2,h_2;\alpha)$[15]

设 g_1,g_2 是素数 q 阶乘法群 G_q 的两个生成元，证明者 P 向验证者 V 证明他知道某个秘密 $\alpha\in Z_q^*$，满足 $\alpha=\log_{g_1}h_1=\log_{g_2}h_2$，而不泄露 α 的值.

首先，P 选取 $w\in_R Z_q$，计算 $a_1=g_1^w$，$a_2=g_2^w$，$c=H(a_1\|a_2)$，$r=w-\alpha c \pmod q$，公布 (r,c). 然后，V 验证 $c\equiv H(g_1^rh_1^c\|g_2^rh_2^c)$ 是否成立. 若成立，V 则相信 P 确实知道秘密信息 α；否则，不相信.

3 提出的无可信中心的可公开验证多秘密共享方案

3.1 符号定义

p,q 表示两个大素数，满足 $q|(p-1)$，G_q 为 Z_p^* 的唯一 q 阶子群. g,h 表示 G_q 中两个不同的生成元. $\Omega=\{P_1,\cdots,P_n\}$ 表示所有参与成员构成的集合. 成员 P_i 的私钥表示为 $z_i\in Z_q$，对应的公钥表示为 $y_i=h^{z_i}\pmod p$. H 表示一个密码 hash 函数，t 表示门限值，m 表示共享的多秘密个数，A 表示协议执行中不诚实的成员集合.

3.2 提出的求解多个共享秘密的算法 $CMS(p,q,G_q,g,\tau,S_{x_1},S_{x_2},\cdots,S_{x_\tau})$

p,q,G_q,g 遵循 3.1 节的定义，$\tau-1$ 表示共享多项式 $f(x)$ 的指数，$S_{x_1},S_{x_2},\cdots,S_{x_\tau}$ 表示以 g 为底，以该多项式 τ 个解 $f(x_i)(i=1,\cdots,\tau)$ 为指数所计算的幂. 令一随机多项式为 $f(x)=\sum_{i=0}^{\tau-1}a_ix^i\pmod q$，$\alpha_{x_i}=f(x_i)$，$S_{x_i}=g^{\alpha_{x_i}}(i=1,\cdots,\tau)$，该算法输入：$p,q,G_q,g,\tau,S_{x_1},S_{x_2},\cdots,S_{x_\tau}$，输出：$k_0=g^{a_0}$，$k_1=g^{a_1},\cdots,k_{\tau-1}=g^{a_{\tau-1}}$.

构造的算法可以按照以下过程求解：

(1) 令集合 $B=\{x_1,x_2,\cdots,x_\tau\}$，将 $f(x)$ 写成

$$f(x)=\sum_{i=1}^{\tau}\alpha_{x_i}\prod_{j\in B\setminus\{x_i\}}\frac{x-x_j}{x_i-x_j}=\sum_{i=1}^{\tau}\frac{\alpha_{x_i}}{\prod_{j\in B\setminus\{x_i\}}x_i-x_j}\prod_{j\in B\setminus\{x_i\}}x-x_j\pmod q \quad (1)$$

展开多项式整理的 x^k 项的系数

$$a_k=\sum_{i=1}^{\tau}\frac{\alpha_{x_i}}{\prod_{j\in B\setminus\{x_i\}}x_i-x_j}\lambda_{k,x_i}\pmod q,$$

其中 $k\in\{0,\cdots,\tau-1\}$. 因此，按照式(1)可展开求解出 λ_{k,x_i}，$k\in\{0,\cdots,\tau-1\}$.

(2) 对所有的 $k\in\{0,\cdots,\tau-1\}$，做以下计算：

$$g^{a_k}\equiv g^{\sum_{i=1}^{\tau}\frac{\alpha_{x_i}}{\prod_{j\in B\setminus\{x_i\}}x_i-x_j}\lambda_{k,x_i}\pmod q}\equiv\prod_{i=1}^{\tau}g^{\frac{\alpha_{x_i}}{\prod_{j\in B\setminus\{x_i\}}x_i-x_j}\lambda_{k,x_i}\pmod q}\equiv\prod_{i=1}^{\tau}(S_{x_i}^{\lambda_{k,x_i}})^{\frac{1}{\prod_{j\in B\setminus\{x_i\}}x_i-x_j}\pmod q}\pmod p$$

输出：$k_0=g^{a_0},k_1=g^{a_1},\cdots,k_{\tau-1}=g^{a_{\tau-1}}$.

3.3 方案思想

为了不依赖于可信分发中心，需要所有成员 $P_i(i=1,\cdots,n)$ 联合生成秘密. 采用同态结构，所有秘密的值都可由一个多项式的相应系数来计算，门限个成员可通过线性插值来重构秘密. 为了获取可公开验证的能力，采用的方法考虑下面两种情况：

(1) 当秘密个数不大于门限值，即 $m\le t$ 时. 在秘密产生阶段，各成员构造的多项式次数是 $t-1$，公开对多项式系数的承诺，并公开加密后的子秘密份额，该成员构造证据以便使他人相信他所公开的子秘密份额都是正确的，任何成员都可以通过公开信息验证子密钥的正确性，最后，成员可通过加密子密钥和他的私钥来计算自己的份额. 在秘密恢复阶段，该成员需向其他成员证明他的份额确实是从加密的子密钥份额中解密出来的，不少于 t 个诚实成员可利用提出的求解多个共享秘密的 CMS 算法来获取重构秘密.

(2) 当秘密个数大于门限值，即 $m>t$ 时. 在秘密产生阶段，各成员所要构造的多项式的次数是 $m-1$，公开该多项式上的 $m-t$ 个解和对应的承诺，其他步骤与 $m\le t$ 时相似. 重构秘密时，首先，通过同态关系，计算 $m-t$ 个公开份额，任何不少于 t 个诚实成员利用 $m-t$ 个公开份额和他们的子密钥，执行提出的求解多个共享秘密的 CMS 算法，其输出即为所有共享的全部秘密.

3.4 方案具体描述

当 $m\le t$ 时，方案描述如下：

(1) 秘密产生阶段

① 成员 P_i 随机选取 $a_{ij}\in_R Z_q(j=0,1,\cdots,t-1)$，构造多项式 $f_i(x)=\sum_{j=0}^{t-1}a_{ij}x^j \bmod q$，计算并公布对系数的承诺 $C_{ik}=g^{a_{ik}} \bmod p(k=0,1,\cdots,t-1)$ 和 $Y_{ij}=y_j^{f_i(j)}\bmod p(j=1,2,\cdots,n)$.

② 成员 P_i 构造证据，以便使其他成员相信他们收到的 Y_{ij} 是有效的，即满足 $X_{ij}=g^{f_i(j)} \bmod p$，$Y_{ij}=y_j^{f_i(j)} \bmod p$. 为此，$P_i$ 任意选取 n 个随机数 $w_{ij} \in_R Z_q$ $(j=1,2,\cdots,n)$，计算

$$a_{1ij}=g^{w_{ij}} \bmod p,\ a_{2ij}=y_j^{w_{ij}} \bmod p,\ j=1,2,\cdots,n.$$

然后构造质询值

$$c_i=H(g \| h \| X_{i1} \| \cdots \| X_{in} \| Y_{i1} \| \cdots \| Y_{in} \| a_{1i1} \| \cdots \| a_{1in} \| a_{2i1} \| \cdots \| a_{2in}) \quad (2)$$

并计算

$$R_i=(r_{i1},\cdots,r_{in})=(w_{i1}-c_i f_i(1)(\bmod q),\cdots,w_{in}-c_i f_i(n)(\bmod q)).$$

公布证据 $Proof_{P_i}=(c_i,R_i)$.

③ 任何成员 $P_l \in \Omega/\{P_i\}$ 都可根据对多项式 $f_i(x)$ 的公开承诺 C_{ik} 来计算

$$X_{ij}=\prod_{k=0}^{t-1} C_{ik}^{j^k} \bmod p,\ j=1,2,\cdots,n.$$

④ 然后，$P_l \in \Omega/\{P_i\}$ 再利用已知的 $c_i, r_{ij}, g, h, X_{ij}, Y_{ij}$ $(j=1,2,\cdots,n)$，计算 a_{1ij} 和 a_{2ij}：

$$a_{1ij}=g^{r_{ij}} X_{ij}^{c_i} \bmod p,\ a_{2ij}=y_j^{r_{ij}} Y_{ij}^{c_i} \bmod p,\ j=1,2,\cdots,n.$$

验证式(2)是否成立. 若成立，则表明成员 P_i 分发的份额是正确的. 否则，广播对成员 P_i 的指控，经过不少于 t 个成员的确认后，将 P_i 加入不诚实成员集合 A，$A=A \cup \{P_i\}$.

⑤ 所有成员都已分发了子秘密份额后，每个成员 $P_j \in \Omega$ 可计算出加密的秘密份额 $Y_j=\prod_{P_i \in \Omega - A} Y_{ij} \bmod p$，再利用私钥 z_j 从加密子密钥 Y_j 中计算出最终的秘密份额 $S_j=Y_j^{z_j^{-1}} \bmod p$.

(2) 秘密恢复阶段

参与秘密恢复的成员 $P_j \in \Omega$ 提供 S_j，并按下面的步骤向其他成员证明 S_j 确实是从 Y_j 中解密出来的.

① 成员 P_j 任意选取 $w_j \in_R Z_q$，并计算 $b_{1j}=h^{w_j} \bmod p$，$b_{2j}=S_j^{w_j} \bmod p$，$r_j=w_j-c_j z_j (\bmod q)$，其中 $c_j=H(b_{1j} \| b_{2j})$.

② 成员 P_j 公布 (r_j,c_j) 作为他知道秘密信息 z_j 的证据.

③ 其他验证者验证 $c_j \equiv H(h^{r_j} y_j^{c_j} \| S_j^{r_j} Y_j^{c_j})$ 是否成立. 若成立，则证明 S_j 确实是从 Y_j 中解密出来的.

④ 任何 t 个证明了他们提供了正确份额 S_{x_i} 的成员 P_{x_i} $(i=1,2,\cdots,t, x_i \in \{1,\cdots,n\})$，通过执行 $CMS(p,q,G_q,g,t,S_{x_1},S_{x_2},\cdots,S_{x_t})$ 算法，输出的 t 个值的前 m 个就是联合产生的随机秘密 $k_0, k_1, \cdots, k_{m-1}$.

当 $m>t$ 时，方案描述如下：

(1) 秘密产生阶段

① 成员 P_i 选取 $a_{ij} \in_R Z_q$ $(j=0,1,\cdots,m-1)$，构造多项式 $f_i(x)=\sum_{j=0}^{m-1} a_{ij} x^j \bmod q$，公开该多项式的 $m-t$ 对解 $(j, f_i(j))$，其中 $j=n+1,\cdots,n+m-t$，并公开对系数的承诺 $C_{ik}=g^{a_{ik}} \bmod p$ $(k=0,1,\cdots,m-1)$，$Y_{ij}=y_j^{f_i(j)} \bmod p$ $(j=1,2,\cdots,n)$.

② 成员 P_i 构造证据，以便使其他成员相信他们收到的 Y_{ij} 是有效的，即满足 $X_{ij}=g^{f_i(j)} \bmod p$，$Y_{ij}=y_j^{f_i(j)} \bmod p$ $(j=1,2,\cdots,n)$. 为此，成员 P_i 任意选取 n 个随机数 $w_{ij} \in_R Z_q$ $(j=1,2,\cdots,n)$，计算

$$a_{1ij}=g^{w_{ij}} \bmod p,\ a_{2ij}=y_j^{w_{ij}} \bmod p,\ j=1,2,\cdots,n.$$

然后构造质询值

$$c_i=H(g \| h \| X_{i1} \| \cdots \| X_{in} \| Y_{i1} \| \cdots \| Y_{in} \| a_{1i1} \| \cdots \| a_{1in} \| a_{2i1} \| \cdots \| a_{2in}) \quad (3)$$

并计算

$$R_i=(r_{i1},\cdots,r_{in})=(w_{i1}-c_i f_i(1)(\bmod q),\cdots,w_{in}-c_i f_i(n)(\bmod q)).$$

公布证据 $Proof_{P_i}=(c_i,R_i)$.

③ 任何成员 $P_l \in \Omega/\{P_i\}$ 都可根据对多项式 $f_i(x)$ 的公开承诺 C_{ik} 来计算

$$X_{ij}=\prod_{k=0}^{t-1} C_{ik}^{j^k} \bmod p,\ j=1,2,\cdots,n.$$

④ 然后，$P_l \in \Omega/\{P_i\}$ 再利用已知的 $c_i, r_{ij}, g, h, X_{ij}, Y_{ij}$ $(j=1,2,\cdots,n)$，计算 a_{1ij} 和 a_{2ij}：

$$a_{1ij}=g^{r_{ij}} X_{ij}^{c_i} \bmod p,\ a_{2ij}=y_j^{r_{ij}} Y_{ij}^{c_i} \bmod p,\ j=1,2,\cdots,n.$$

验证式(3)是否成立. 若成立，则表明成员 P_i 分发的 Y_{ij} 是正确的. 否则，广播对成员 P_i 的指控，经过不少于 t 个成员的确认后，将 P_i 加入不诚实成员集合 A，$A=A \cup \{P_i\}$.

⑤ 任何成员都可以通过下面等式验证 $(j, f_i(j))$，$(j=n+1,\cdots,n+m-t)$ 是否成立

$$g^{f_i(j)} \equiv \prod_{k=0}^{m-1} C_{ik}^{j^k} (\bmod p).$$

若成立，则表明成员 P_i 公开的 $f_i(j)$ $(j=n+1,\cdots,n+m-t)$ 是正确的. 否则，广播对成员 P_i 的指控，经过超过 t 个成员的确认后，将 P_i 加入不诚实成员集合 A，$A=A \cup \{P_i\}$.

⑥ 所有成员都分发了子秘密份额后，每个成员 $P_j \in \Omega$ 可计算出加密的秘密份额 $Y_j=\prod_{P_i \in \Omega - A} Y_{ij} \bmod p$，

再利用私钥 z_j 从加密秘密份额 Y_j 中计算出最终的秘密份额 $S_j=Y_j^{z_j^{-1}} \bmod p$.

⑦ 每个成员 $P_j\in\Omega$ 计算联合生成多项式的另外 $m-t$ 个份额

$$T_j=\prod_{P_i\in\Omega-A} h^{f_i(j)} \bmod p(j=n+1,\cdots,n+m-t).$$

(2) 密钥恢复阶段

参与秘密恢复的成员 $P_j\in\Omega$ 提供 S_j,并按下面的步骤向其他成员证明 S_j 确实是从 Y_j 中解密出来的.

① 成员 P_j 任意选取 $w_j\in_R Z_q$,并计算 $b_{1j}=h^{w_j} \bmod p$, $b_{2j}=S_j^{w_j} \bmod p$, $r_j=w_j-c_jz_j \pmod q$,其中 $c_j=H(b_{1j}\|b_{2j})$.

② 成员 P_j 公布 (r_j,c_j) 作为他知道秘密信息 z_j 的证据.

③ 其他验证者验证 $c_j\equiv H(h^{r_j}y_j^{c_j}\|S_j^{r_j}Y_j^{c_j})$ 是否成立,若成立,则证明 S_j 确实是从 Y_j 中解密出来的.

④ 任何 t 个证明了他们提供了正确份额 S_{x_i} 的成员 $P_{x_i}(i=1,2,\cdots,t,x_i\in\{1,\cdots,n\})$,通过执行 $CMS(p,q,G_q,g,m,S_{x_1},S_{x_2},\cdots,S_{x_t},T_{n+1},\cdots,T_{n+m-t})$ 算法,就可以输出 m 个联合产生的随机秘密 $k_0,k_1,\cdots,k_{m-1}$.

4 安全性分析

定理 1. 当 $n\geq 2t-1$ 时,即使存在多达 $t-1$ 个成员是不诚实的,通过提出方案,各诚实成员在分发阶段仍然可以获得正确的份额,在秘密恢复阶段,仍然可以恢复出正确的秘密.

证明. 首先,考虑 $m\leq t$ 的时候. 在密钥产生阶段,当成员 P_i 分发了份额后,验证者利用公开的 c_{ij}, r_{ij} 计算 a_{1ij}, a_{2ij},然后把得到的 a_{1ij}, a_{2ij} 与 g, h, X_{ij}, Y_{ij} 代入式(2),如果得到的结果与成员 P_i 所公布的 c_{ij} 相同,则说明成员 P_i 分发了正确的份额;否则,当 a_{1ij}, a_{2ij}, g, h, X_{ij}, Y_{ij} 中的任何一个值有误时,等式(2)均不成立,说明成员 P_i 分发了错误的份额. 事实上,如果成员 P_i 分发了正确的份额,那么 $g^{r_{ij}}X_{ij}^{c_{ij}}=g^{w_{ij}-c_{ij}f_i(j)}g^{c_{ij}f_i(j)}=g^{w_{ij}} \pmod p$, $y_j^{r_{ij}}Y_{ij}^{c_{ij}}=y_j^{w_{ij}-c_{ij}f_i(j)}y_j^{c_{ij}f_i(j)}=y_j^{w_{ij}} \pmod p$. 每个成员都可以通过自己的私钥解密 Y_j,获取自己正确的份额 $S_j=Y_j^{z_j^{-1}} \bmod p$. 密钥恢复阶段,只有当 S_j 确实是从 Y_j 中解密出来的,P_j 才能通过①~③的零知识证明步骤,事实上,假设 S_j 是从 Y_j 中解密出来的,也就是 $S_j=Y_j^{z_j^{-1}} \bmod p$,有 $Y_j=S_j^{z_j} \bmod p$,那么就有 $h^{r_j}y_j^{c_j}=h^{w_j-c_jz_j}h^{z_jc_j}=h^{w_j}=b_{1j} \pmod p$, $S_j^{r_j}Y_j^{c_j}=S_j^{w_j-c_jz_j}S_j^{z_jc_j}=S_j^{w_j}=b_{2j}=\pmod p$. 因为 $n\geq 2t-1$,即使存在多达 $t-1$ 个成员是不诚实的,仍然有不少于 t 个成员是诚实的,通过这些诚实成员仍然可以在④步中联合产生共享的秘密.

相似的,在 $m>t$ 时,定理仍然成立. 证毕.

定理 2. 若 Diffie-Hellman 假设成立,则提出方案中的份额加密算法是安全的.

证明. 假设提出方案中的份额加密算法是不安全的,即存在算法 I 能攻破份额加密算法,可以利用算法 I 来攻击 Diffie-Hellman 假设. 假设给定输入 g, h, X_{ij}, y_j, $Y_{ij}(P_i\in\Omega-A)$,算法 I 能以某个不可忽略概率 ε 成功计算出 P_j 的份额 $S_j(=h^{f(j)}=h^{\sum_{P_i\in\Omega-A}f_i(j)})$,注意到 $X_j=\prod_{P_i\in\Omega-A}X_{ij} \bmod p(g^{f(j)})$. 给定 g^α 和 g^β,我们的目标是使用算法 I 计算 $z=g^{\alpha\beta}$. 令 $i'=\{\min(i)\mid P_i\in\Omega-A\}$,选择 α', β_i, $\gamma_i(P_i\in\Omega-A)$,提供给算法 I 一随机输入 g, $h=g^{\alpha\alpha'}$, $X_{i'j}=g^{\beta\beta_{i'}}$, $X_{ij}=g^{\beta_i}(P_i\in\Omega-A$ 且 $i\neq i')$, $y_j=h^\gamma$, $Y_{i'j}=h^{\beta_{i'}\gamma}$, $Y_{ij}=h^{\beta_i}(P_i\in\Omega-A$ 且 $i\neq i')$,则算法 I 能够以 ε 概率输出: $S_j=g^{\alpha\alpha'\beta\prod_{P_i\in\Omega-A}\beta_i}$,由此可以计算 $z=(g^{\alpha\alpha'\beta\prod_{P_i\in\Omega-A}\beta_i})^{\alpha'^{-1}\prod_{P_i\in\Omega-A}\beta_i^{-1}}=g^{\alpha\beta}$.

这与 Diffie-Hellman 假设矛盾,所以提出方案中的份额加密算法是安全的. 证毕.

定理 3. 提出的方案中,若 Diffie-Hellman 假设成立,则任意 $t-1$ 个成员利用他们的份额都不能恢复出任何秘密.

证明. 利用反证法,假设攻击者知道 $t-1$ 个成员的子密钥,并能利用这 $t-1$ 个成员的子密钥恢复出某个秘密. 不失一般性,假设这 $t-1$ 个成员是前 $t-1$ 个成员 $P_i(i=1,\cdots,t-1)$,恢复的秘密是 k_1.

首先,考虑 $m\leq t$ 的情况. 假定每个成员 P_i 选择的多项式为 $f_i(x)=\sum_{j=0}^{t-1}a_{ij}x^j \pmod q$,最终联合生成的秘密多项式为 $f(x)=a_0+a_1x+\cdots+a_{t-1}x^{t-1} \pmod q$,因此 $f(x)=\sum_{P_i\in\Omega-A}f_i(x)$. 构造一个模拟系统来模拟攻击者 I 在实际协议中的所有观察,设置如下:

(1) 令 $h=g^\alpha$,对所有 $P_i\in\Omega-A$,随机选择 $C_{i1}(g^{a_{i1}})$ 满足 $\prod_{P_i\in\Omega-A}C_{i1}=g^\beta$. 这就隐含确定了 $C_1=g^\beta(a_1=\beta)$,也隐含确定了 a_{i1}.

(2) 对所有的 $i=1,\cdots,n$，随机选取 $t-1$ 个值 $f_i(1),f_i(2),\cdots,f_i(t-1)\in_R Z_q$，这 $t-1$ 个值和所有 C_{i1} 就固定了函数 $f_i(x)$.

(3) 计算前 $t-1$ 个 X_{ij} 和 Y_{ij} 的值：$X_{ij}=g^{f_i(j)}$，$Y_{ij}=y_j^{f_i(j)}(j=1,2,\cdots,t-1)$.

(4) 下面利用公开值和固定的 C_{i1} 值来计算 C_{ij} $(j=0,2,3,\cdots,t-1)$. 由 $f_i(x)=\sum_{j=0}^{t-1}a_{ij}x^j \bmod q$，可得

$$\begin{pmatrix}1 & 1 & \cdots & 1\\ 1 & 2^2 & \cdots & 2^{t-1}\\ \cdots & \cdots & \cdots & \cdots\\ 1 & (t-1)^2 & \cdots & (t-1)^{t-1}\end{pmatrix}\cdot\begin{pmatrix}a_{i0}\\ a_{i2}\\ \vdots\\ a_{i(t-1)}\end{pmatrix}=$$

$$\begin{pmatrix}f_i(1)-a_{i1}\cdot 1\\ f_i(2)-a_{i1}\cdot 2\\ \vdots\\ f_i(t-1)-a_{i1}\cdot(t-1)\end{pmatrix}\tag{4}$$

记 $\boldsymbol{A}=\begin{pmatrix}1 & 1 & \cdots & 1\\ 1 & 2^2 & \cdots & 2^{t-1}\\ \cdots & \cdots & \cdots & \cdots\\ 1 & (t-1)^2 & \cdots & (t-1)^{t-1}\end{pmatrix}$，因为 $\boldsymbol{A}$ 的秩是 $t-1$，所以，它必有逆矩阵，记为 $\boldsymbol{A}^{-1}$，不妨设求解的 $\boldsymbol{A}^{-1}=\begin{pmatrix}b_{00} & b_{02} & \cdots & b_{0(t-1)}\\ b_{20} & b_{22} & \cdots & b_{2(t-1)}\\ \cdots & \cdots & \cdots & \cdots\\ b_{t-1,0} & b_{t-1,2} & \cdots & b_{(t-1)(t-1)}\end{pmatrix}$，则原方程可以写成另一种形式

$$\begin{pmatrix}a_{i0}\\ a_{i2}\\ \vdots\\ a_{i(t-1)}\end{pmatrix}=\boldsymbol{A}^{-1}\begin{pmatrix}f_i(1)-a_{i1}\cdot 1\\ f_i(2)-a_{i1}\cdot 2\\ \vdots\\ f_i(t-1)-a_{i1}\cdot(t-1)\end{pmatrix},$$

故 $a_{ij}=\sum_{l=1}^{t-1}b_{jl}(f_i(l)-a_{i1}\cdot l)$ $(j=0,2,3,\cdots,t-1)$. 可得

$$g^{a_{ij}}=g^{\sum_{l=1}^{t-1}b_{jl}(f_i(l)-a_{i1}\cdot l)}=\prod_{l=1}^{t-1}g^{b_{jl}(f_i(i)-a_{i1}\cdot l)}$$

$$=\prod_{l=1}^{t-1}(X_{ij})^{b_{jl}}\Big/\prod_{l=1}^{t-1}C_{i1}^{b_{jl}\cdot l}\bmod p.$$

即 $C_{ij}=\prod_{l=1}^{t-1}(X_{ij})^{b_{jl}}\Big/\prod_{l=1}^{t-1}C_{i1}^{b_{jl}\cdot l}(j=0,2,3,\cdots,t-1)$.

(5) 计算其余的 X_{ij}，$X_{ij}=\prod_{l=0}^{i-1}C_{il}^{j^l}\bmod p(j=t,\cdots,n)$.

(6) 最后，构造 $y_i,Y_{ij}(j=t,\cdots n)$. 随机选取 $u_j\in_R Z_q(j=t,\cdots,n)$，令 $y_j=g^{u_j}$，$Y_{ij}=(X_{ij})^{u_i}(i=t,\cdots,n)$. 则 Y_{ij} 满足 $Y_{ij}\equiv y_j^{f_i(j)}$，这是因为 $Y_{ij}=(X_{ij})^{u_j}=(g^{f_i(j)})^{u_j}=(g^{u_j})^{f_i(j)}=(y_j)^{f_i(j)}$.

通过以上模型系统就可以模拟敌手的所有观察. 由于已假设 $t-1$ 个成员 $P_i(i=1,\cdots,t-1)$ 可以恢复出秘密 $k_1=h^{a_1}$，又因为 $h=g^{\alpha}$，$C_1=g^{\beta}$. 所以，这 $t-1$ 个成员通过模拟器必可以计算出 $g^{\alpha\beta}=(g^{\alpha})^{\beta}=h^{a_1}=k_1$，这与 Diffie-Hellman 假设矛盾.

然后，考虑 $m>t$ 的情况，假定成员 P_i 选择的多项式为 $f_i(x)=\sum_{j=0}^{m-1}a_{ij}x^j(\bmod q)$，所有最终生成的秘密多项式为 $f(x)=a_0+a_1x+\cdots+a_{m-1}x^{m-1}(\bmod q)$，因此 $f(x)=\sum_{P_i\in\Omega-A}f_i(x)$. 构造一个模拟系统来模拟攻击者 I 在实际协议中的所有观察，设置如下：

(1) 令 $h=g^{\alpha}$，对所有 $P_i\in\Omega-A$，随机选择 $C_{i1}(g^{a_{i1}})$ 满足 $\prod_{P_i\in\Omega-A}C_{i1}=g^{\beta}$. 隐含确定了 $C_1=g^{\beta}(a_1=\beta)$，也隐含确定了 a_{i1}.

(2) 对所有的 $i=1,\cdots,n$，根据公开的 $f_i(j)(j=n+1,\cdots,m+n-t)$，计算 $f(j)=\sum_{P_i\in\Omega-A}f_i(j)(\bmod q)$ $(j=n+1,\cdots,m+n-t)$. 随机选取 $t-1$ 个值 $f_i(1)$，$f_i(2),\cdots,f_i(t-1)\in_R Z_q$，这 $t-1$ 个值和 $f_i(j)(j=n+1,\cdots,m+n-t)$ 以及 C_{i1} 就固定了函数 $f_i(x)$.

(3) 计算前 $t-1$ 个 X_{ij} 和 Y_{ij} 的值：$X_{ij}=g^{f_i(j)}$，$Y_{ij}=y_j^{f_i(j)}(j=1,2,\cdots,t-1)$.

(4) 利用公开值和固定的 C_{i1} 值来计算 $C_{ij}(j=0,2,3,\cdots,m-1)$. 由 $f_i(x)=\sum_{j=0}^{m-1}a_{ij}x^j \bmod q$，且根据式(4)求解 a_{ij} 的方法相似，可以知道

$$a_{ij}=\sum_{l=1}^{t-1}b_{jl}(f_i(l)-a_{i1}\cdot l)+\sum_{l=n+1}^{n+m-t}b_{jl}(f_i(l)-a_{i1}\cdot l),$$

这里 $(j=0,2,3,\cdots,m-1)$，b_{ji} 是相应系数矩阵的逆矩阵中的对应元素. 因此

$$g^{a_{ij}}=g^{\sum_{l=1}^{t-1}b_{jl}(f_i(l)-a_{i1}\cdot l)+\sum_{l=n+1}^{n+m-t}b_{jl}(f_i(l)-a_{i1}\cdot l)}$$

$$=\prod_{l=1}^{t-1}g^{b_{jl}(f_i(i)-a_{i1}\cdot l)}\cdot\prod_{l=n+1}^{n+m-t}g^{b_{jl}(f_i(i)-a_{i1}\cdot l)}$$

$$=\prod_{l=1}^{t-1}(X_{ij})^{b_{jl}}\cdot\prod_{l=n+1}^{n+m-t}(X_{ij})^{b_{jl}}\Big/$$

$$\Big(\prod_{l=1}^{t-1}C_{i1}^{b_{jl}\cdot l}\cdot\prod_{l=n+1}^{n+m-t}C_{i1}^{b_{jl}\cdot l}\Big)\bmod p$$

即

$$C_{ij}=\prod_{l=1}^{t-1}(X_{ij})^{b_{jl}}\cdot\prod_{l=n+1}^{n+m-t}(X_{ij})^{b_{jl}}\Big/\Big(\prod_{l=1}^{t-1}C_{i1}^{b_{jl}\cdot l}\cdot\prod_{l=n+1}^{n+m-t}C_{i1}^{b_{jl}\cdot l}\Big),$$

这里，$j=0,2,3,\cdots,m-1$.

(5) 计算其余的 X_{ij}，$X_{ij}=\prod_{l=0}^{t-1}C_{il}^{j^l} \bmod p$ $(j=t,\cdots,n)$.

(6) 最后，构造 y_i, Y_{ij} $(j=t,\cdots,n)$. 随机选取 $u_j\in_R Z_q$ $(j=t,\cdots,n)$，令 $y_j=g^{u_j}$，$Y_{ij}=(X_{ij})^{u_i}$，这里 $i=t,\cdots,n$. 则 Y_{ij} 满足 $Y_{ij}\equiv y_j^{f_i(j)}$，这是因为 $Y_{ij}=(X_{ij})^{u_j}=(g^{f_i(j)})^{u_j}=(g^{u_j})^{f_i(j)}=(y_j)^{f_i(j)}$.

通过以上模型系统就可以模拟敌手的所有观察. 因为已假设 $t-1$ 个成员 $P_i(i=1,\cdots,t-1)$ 可以恢复出秘密 $k_1=h^{a_1}$，又因为 $h=g^{\alpha}$，$C_1=g^{\beta}$，所以这 $t-1$ 个成员通过模拟器必可以计算出 $g^{\alpha\beta}=(g^{\alpha})^{\beta}=h^{a_1}=k_1$，这与 Diffie-Hellman 假设矛盾. 证毕.

5 相关工作的比较

下面该文将提出的方案与文献[3]的方案、文献[6]的方案、文献[7]的方案、文献[8-9]的方案和文献[10]的方案进行比较. 文献[3]的方案能对份额进行验证，但不能公开验证份额，且不能一次对多个秘密进行共享. 文献[6]的方案虽然能公开验证份额，但也不能同时对多个秘密进行共享，且需要可信中心. 文献[7]的方案虽然不需要可信中心，但只能一次对单个秘密进行共享. 文献[8-9]的方案能一次对任意多个秘密进行共享，但不具有可(公开)验证能力，也需要可信中心的参与. 文献[10]的方案能一次对多个秘密进行共享，但不具有公开验证的能力，且需要可信中心的参与. 该文提出的多秘密方案，不仅能公开验证份额，而且不需要可信中心，这使得方案具有更广的应用，如应用于可撤销匿名性的电子支付协议及电子选举协议等. 表 1 给出了几种秘密共享方案在相关方面的比较.

表 1 相关工作

方 案	是否能验证份额	是否具有可公开验证属性	是否能同时共享多个秘密	是否不需要可信中心
文献[3]的方案	是	否	否	否
文献[6]的方案	是	是	否	否
文献[7]的方案	是	否	否	是
文献[8-9]的方案	否	否	是	否
文献[10]的方案	是	否	是	否
提出的方案	是	是	是	是

6 成员的动态加入和删除

在一些实际的网络环境诸如无线自组网中，秘密共享在密钥管理等方面发挥着重要的作用. 然而，这些网络中，难以存在可信中心且成员往往需要动态变化. 因此，如何在无可信中心的情况下实现成员的动态撤出和加入是十分有意义的工作. 上述提出的方案实现成员的撤出是十分容易的，如果成员 P_i 需要撤出多秘密共享方案时，只需所有成员将 P_i 的公钥 y_i 注销，在以后秘密恢复阶段，不再接受 P_i 的份额即可.

当成员 P_{n+1} 加入上述多秘密共享方案时，可利用文献[15]的思想，执行下面的协议完成. 令 B 表示协议执行时不诚实的成员集合，初始化 $B=\varnothing$.

(1) 首先，P_{n+1} 选择他的私钥 $z_{n+1}\in Z_q$，公开对应的公钥 $y_{n+1}=h^{z_{n+1}} \bmod p$.

(2) 每个成员 $P_i(i=1,2,\cdots,n)$ 选择一随机多项式 $g_i(x)=\sum_{l=0}^{t-1}b_{il}x^l \bmod q$，计算并公布承诺 $B_{ij}=g^{b_{ij}} \bmod p$，$0\le j\le t-1$ 和 $Z_{ij}=y_j^{g_i(j)} \bmod p$，$j=1,2,\cdots,n+1$. 令 $E_{ij}=\prod_{l=0}^{t-1}B_{il}^{j^l}=\prod_{l=0}^{t-1}(g^{b_{il}})^{j^l}=g^{\sum_{l=0}^{t-1}b_{il}\cdot j^l}=g^{g_i(j)} \bmod p$，$j=1,2,\cdots,n+1$.

(3) 每个成员 $P_i(i=1,2,\cdots,n)$ 选择 $n+1$ 个随机数 $w'_{ij}\in_R Z_q^*$ $(j=1,2,\cdots,n+1)$，计算 $a'_{1ij}=g^{w'_{ij}} \bmod p$，$a'_{2ij}=y_j^{w'_{ij}} \bmod p$，这里 $j=1,2,\cdots,n+1$.

然后构造质询值

$$c'_i=H(g\|y_1\|\cdots\|y_{n+1}\|E_{i1}\|\cdots\|E_{i(n+1)}\|Z_{i1}\|\cdots\|Z_{i(n+1)}\|a'_{1i1}\|\cdots\|a'_{1i(n+1)}\|a'_{2i1}\|\cdots\|a'_{2i(n+1)}) \tag{5}$$

并计算

$$R'_i=(r'_{i1},\cdots,r'_{i(n+1)})=(w'_{i1}-c'_ig_i(1) \bmod q,\cdots,w'_{i(n+1)}-c'_ig_i(n+1) \bmod q).$$

最后，公布证据 $Proof_{P_i}=(c'_i,R'_i)$.

(4) 成员 P_{n+1} 对于每个 $j(j=1,2,\cdots,n+1)$，先计算 $E_{ij}=\prod_{l=0}^{t-1}B_{il}^{j^l}=\prod_{l=0}^{t-1}(g^{b_{il}})^{j^l}=g^{g_i(j)} \bmod p$，再计算 $a'_{1ij}=g^{r'_{ij}}E_{ij}^{c'_i} \bmod p$ 和 $a'_{2ij}=y_j^{r'_{ij}}Z_{ij}^{c'_i} \bmod p$. 然后，验证等式(5)是否成立. 若成立，则表明成员 P_i 分发的 Z_{ij} 是正确的，P_{n+1} 计算

$$\delta_{i(n+1)}=(Z_{i(n+1)})^{z_{n+1}^{-1}}=(y_{n+1}^{g_i(n+1)})^{z_{n+1}^{-1}}=h^{g_i(n+1)} \bmod p.$$

否则，广播对成员 P_i 的指控，经过不少于 t 个成员的确认后，将 P_i 加入不诚实成员集合 B，令 $B=B\cup\{P_i\}$. 最后，P_{n+1} 计算 $\delta_{n+1}=\prod_{P_i\in\Omega-B}\delta_{i(n+1)}(\bmod p)$.

(5) 每个成员 $P_k(k=1,2,\cdots,n)$ 对于每个 $j(j=1,2,\cdots,n+1)$ 需验证 P_i 分发的 Z_{ij} 是否正确. 先计算 $E_{ij}=\prod_{l=0}^{t-1}B_{il}^{j^l}=\prod_{l=0}^{t-1}(g^{b_{il}})^{j^l}=g^{\sum_{l=0}^{t-1}b_{il}\cdot j^l}=g^{g_i(j)}(\bmod p)$，再计算 $a'_{1ij}=g^{r'_{ij}}E_{ij}^{c'_i}(\bmod p)$ 和 $a'_{2ij}=y_j^{r'_{ij}}Z_{ij}^{c'_i}(\bmod p)$. 然后，验证等式(5)是否正确. 若成立，则表明成员 P_i 分发的 Z_{ij} 是正确的；否则，广播对成员 P_i 的指控，经过不少于 t 个成员的确认后，将 P_i 加入不诚实成员集合 B，令 $B=B\cup\{P_i\}$.

(6) 每个成员 $P_j(j=1,2,\cdots,n)$ 计算 $S'_j=S_j\cdot\prod_{P_i\in\Omega-B}\delta_{ij}(\bmod p)$.

(7) 每个成员 $P_j(j=1,2,\cdots,n)$ 选择 $w_{j(n+1)}\in_R Z_q^*$，先计算 $a_{1j(n+1)}=h^{w_{j(n+1)}}(\bmod p)$ 和 $a_{2j(n+1)}=S_j'^{w_{j(n+1)}}(\bmod p)$；再计算

$$c_{j(n+1)}=(h\|S'_j\|y_j\|Y_j\prod_{P_i\in\Omega-B}Z_{ij}\|a_{1j(n+1)}\|a_{2j(n+1)}),$$

$$r_{j(n+1)}=w_{j(n+1)}-c_{j(n+1)}\cdot z_j(\bmod q)\qquad(6)$$

最后，公布 $Proof_{P_i}=(c_{j(n+1)},r_{j(n+1)})$.

(8) 每个成员 $P_j(j=1,2,\cdots,n)$ 计算并广播 $\theta_j=S'_j(y_{n+1})^{z_j}(\bmod p)$.

(9) 成员 P_{n+1} 解密 $\theta_j\cdot(y_j^{z_{n+1}})^{-1}=S'_j(y_{n+1})^{z_j}\cdot(y_j^{z_{n+1}})^{-1}=S'_jh^{z_{n+1}z_j}\cdot h^{-z_jz_{n+1}}=S'_j(\bmod p)$，并计算 $a_{1j(n+1)}=h_j^{r_{j(n+1)}}y_j^{c_{j(n+1)}}(\bmod p)$ 和 $a_{2j(n+1)}=S_j'^{r_{j(n+1)}}\left(Y_j\prod_{P_i\in\Omega-B}Z_{ij}\right)^{c_{j(n+1)}}(\bmod p)$，然后，验证等式(6)是否成立. 成员 P_{n+1} 选择 t 个通过上述验证的成员的编号 $j(j\in\Lambda,|\Lambda|=t)$，最后计算它的份额 $S_{n+1}=\left(\prod_{j\in\Lambda}S_j'^{C_{\Lambda j}(n+1)}\right)\Big/\delta_{n+1}(\bmod p)$，这里 $C_{\Lambda j}(n+1)=\prod_{i\in\Lambda}\frac{n+1-i}{j-i}$.

在上述协议的(2)(3)步中，成员 $P_i(i=1,2,\cdots,n)$ 产生随机份额 δ_{ij} 将其加密的版本 Z_{ij} 分发给成员 $P_j(j=1,2,\cdots,n+1)$；在第(4)(5)步中，P_{n+1} 和 $P_k(k=1,2,\cdots,n)$ 验证 $Z_{ij}(j=1,2,\cdots,n+1)$ 是否是正确的加密版本，并分别解密获取 $\delta_{i(n+1)}$ 和 $\delta_{ij}(j=1,2,\cdots,n)$；在(6)步中，$P_j(j=1,2,\cdots,n)$ 计算份额 S_j 的盲化版本 S'_j；在第(7)(8)步中，$P_j(j=1,2,\cdots,n)$ 将 S'_j 加密成 θ_j 发送给 P_{n+1}，并证明它是一个正确的加密版本；在第(9)步中，成员 P_{n+1} 解密 S_j 的盲化版本 S'_j，并验证其正确性，最后插值并解盲化获取其份额 S_{n+1}.

7　结　论

本文提出了一个无可信中心的可公开验证多秘密共享方案，在没有可信分发中心的情况下，所有成员可一次性联合生成多个随机秘密，任意 t 个成员只要拿出正确的秘密份额就可恢复出秘密. 方案中，份额的正确性不但能被份额持有者验证，还可以被任何其他成员验证. 本文证明了方案的安全性，也给出了与其他相关工作的对比. 最后，讨论了在无可信中心的条件下成员的动态撤出和加入问题.

参　考　文　献

[1] Shamir A. How to share a secret. Communications of the ACM, 1979, 22(11): 612-613

[2] Blakley G R. Safeguarding cryptographic keys//Proceedings of the National Computer Conference. New York, USA, 1979: 313-317

[3] Feldman P. A practical scheme for non-interactive verifiable secret sharing//Proceedings of the 28th IEEE Symposium on the Foundations of Computer Science. Los Angeles, USA, 1987: 427-437

[4] Pedersen T. Non-interactive and information-theoretic secure verifiable secret sharing//Proceedings of the CRYPTO'91. Snata Barbara, USA, 1991: 129-140

[5] Stadler M. Publicly verifiable secret sharing//Proceedings of the EUROCRYPT'96. Saragossa, Spain, 1996: 190-199

[6] Schoenmakers B. A simple publicly verifiable secret sharing scheme and its application to electronic voting//Proceedings of the CRYPTO'99. Santa Barbara, USA, 1999: 148-164

[7] Gennaro R, Jarecki S, Krawczyk H, Rabin T. Secure distributed key generation for discrete-log based cryptosystems //Proceedings of the Eurocrypt'99. Prague, Czech Republic, 1999: 295-310

[8] He J, Dawson E. Multistage secret-sharing scheme based on one-way function. Electronics Letters, 1994, 30(19): 1591-1592

[9] Yang C C, Chang T Y, Hwang M S. A (t,n) multi-secret sharing scheme. Applied Mathematics and Computations, 2004, 151(2): 483-490

[10] Dehkordi M H, Mashhadi S. New efficient and practical verifiable multi-secret sharing scheme. Information Sciences, 2008, 178(9): 2262-2274

[11] Eslami Z, Ahmadabadi J Z. A verifiable multi-secret sharing scheme based on cellular automata. Information Sciences, 2010, 180(15): 2889-2894

[12] Hu C, Liao X, Cheng X. Verifiable multi-secret sharing based on LFSR sequences. Theoretical Computer Science, 2012, 445(1): 52-62

[13] Herranz J, Alexandre R, Germán S. New results and applications for multi-secret sharing schemes. Designs, Codes and Cryptography, 2013: 1-24

[14] Wu T Y, Tseng Y M. Publicly verifiable multi-secret sharing scheme from bilinear pairings. IET Information Security, 2013, 7(3): 239-246

[15] Yu J, Kong F, Hao R. Publicly verifiable secret sharing with enrollment ability//Proceedings of the SNPD'07. Qingdao, China, 2007: 194-199

YU Jia, born in 1976, Ph. D., professor. His research interests include information security and cryptology.

CHEN Yang-Kui, born in 1984, M. S. candidate. His research interest is information security.

HAO Rong, born in 1976, M. S., experimentalist. Her research interests include information security and cryptology.

KONG Fan-Yu, born in 1978, Ph.D., associate professor. His research interests include information security and cryptology.

CHENG Xiang-Guo, born in 1969, Ph. D., associate professor. His research interests include information security and cryptology.

PAN Zhen-Kuan, born in 1966, Ph. D., professor. His research interests include image processing and information security.

Background

Multi-secret sharing is an important cryptographic technique in which multiple secrets can be simultaneously shared during one computational process. There have been many researches on multi-secret sharing. Jackson et al. firstly classified multi-secret sharing schemes into two types: one-time-use type and multi-use type. In the one-time-use type, we must redistribute the secret shadows to the participants once some secrets are recovered. In the other type, we do not need to change the shadows of the secrets even if some secrets are recovered. They also proposed a multi-secret sharing scheme by one-way function. Yang et al. proposed an efficient MSS scheme with elegant construction. This scheme used two-variable one-way function to resolve the shadows disclosed problem during recovery phase. Unfortunately, it cannot identify whether the dealer and the participants are honest or not. A multi-secret scheme based on cellular automata was also proposed by Eslami et al.

However, in above multi-secret sharing schemes, there is a trusted center to complete the distribution of secret shares. The existence of a trusted center, however, is impossible in many applications such as ad hoc network. Even if there is a trusted center in certain circumstance, it is easily targeted for an adversary and become a blind spot of the system. In addition, in above multi-secret sharing schemes, authors did not consider the publicly verifiable property so one shareholder cannot verify the validity of other shareholders. Publicly verifiable property is very important for many applications, such as electronic voting, key escrow etc. In order to deal with the above problems, this paper proposed a publicly verifiable multi-secret sharing scheme without trusted centers. In this scheme, the validity of shares can be verified not only by shareholders themselves but also by any other participates. Shared multiple secrets are jointly generated by the participations. In order to make the scheme adapt to the new network circumstance such as ad hoc networks, we also discuss the problem of how to dynamically withdraw and add members without trusted centers.

This research is supported by the National Natural Science Foundation of China under Grant Nos. 61272425, 61303197, 61202475, Qingdao Science and Technology Development Project under Grant Nos. 12-1-4-2-(16)-jch, 12-1-4-2-(14)-jch, 13-1-4-151-jch, Huawei Technology Fund No. YB2013120027, the Open Research Fund from Shandong Provincial Key Laboratory of Computer Network No. SDKLCN-2013-03 and the Open Research Fund from the State Key Laboratory of Information Security.