# 利用 RSA 密码体制解决安全多方多数据排序问题

邱 梅[1,2],罗守山[1,2],刘 文[1,2],陈 萍[3]
(1. 北京邮电大学网络与交换技术国家重点实验室信息安全中心,北京 100876;
2. 西安电子科技大学综合业务网理论及关键技术国家重点实验室,陕西西安 710071;
3. 北京邮电大学电信工程学院,北京 100876)

**摘 要**: 本文研究了姚氏百万富翁问题的一个推广问题,安全多方多数据排序问题:假设有 $n$ 方 $P_1, P_2, \cdots P_n$,他们分别拥有一个保密数据集 $D_{P_1}, D_{P_2} \cdots D_{P_n} \subset \{1,2,\cdots,N\}$. 我们对这多个数据集的并集 $D = D_{P_1} \cup D_{P_2} \cup \cdots \cup D_{P_n}$ 中所有的数据进行一个安全的排序,要求在排序结束后各方能够知道他们各自拥有的数据在 $D$ 中的次序,并且任意一方都不知道其它方拥有的数据的任何信息. 我们提出了一个基于 RSA 同态密码体制的解决安全多方多数据排序问题的方案,并在半诚实模型下对该协议的正确性、安全性和效率进行了分析.

**关键词**: 密码学; 安全多方计算; 计算不可区分; 同态加密体制; 数据排序
**中图分类号**: TN309 **文献标识码**: A **文章编号**: 0372-2112 (2009) 05-1119-05

# A Solution of Secure Multi-Party Multi-Data Ranking Problem Based on RSA Encryption Scheme

QIU Mei[1,2], LUO Shou-shan[1,2], LIU Wen[1,2], CHEN Ping[3]
(1. *Information Security Center, National Key Laboratory of Network and Change, Beijing University of Posts and Telecommunications, Beijing* 100876, *China*;
2. *National Key Laboratory of Integrated Service Networks, Xidian University, Xi'an, Shaanxi* 710071, *China*;
3. *School of Telecommunication Engineering, Beijing University of Posts and Telecommunications, Beijing* 100876, *China*)

Abstract: In this paper, we extend the Yao's millionaire problem to the multi-party multi-data ranking problem, which involves n parties $P_1, P_2, \cdots, P_n$ and eachhas a private data set $D_{P_1} \subset \{1,2,\cdots,N\}$. It forms a ranking problem in $D = D_{P_1} \cup D_{P_2} \cup \cdots \cup D_{P_n}$, which requires the $P_i$ can get no more information beyond the orders of the elements in $D_{P_1} \subset \{1,2,\cdots,N\}$. We propose a protocol based on RSA homomorphic encryption in semi-honest model for this problem and analysis the correctness, security and efficiency.

Key words: encryption; secure multi-party computation (SMC); computationally indistinguishable; homomorphic encryption; data ranking

## 1 引言

对一个约定的函数 $f: D^n \to D^{'}$, $n$ 个互不信任的人用一种特殊的方法保密地输入 $x_1, x_2, \cdots, x_n$,并正确地计算出 $f(x_1, x_2, \cdots, x_n)$,同时确保除函数值外不泄露任何有关保密输入 $x_1, x_2, \cdots, x_n$ 的信息,这就是安全多方计算[1]. 它最早由 A. C. yao 于 1982 年提出[2].

已经研究的具体 SMC 问题包括保护私有信息科学计算问题[3],保护私有信息计算几何问题[4],保密数据挖掘问题[5],安全多方统计分析问题[6]等.

在保护私有信息的数据比较方面,前人工作如下:文献[2]就比较大小问题给出了一个协议,但效率极低,对比较相等问题没有讨论. 文献[7]对比较大小给出了一个有效且公平的协议,并对安全性进行了证明,但却没有涉及比较相等的问题;文献[8,9]虽然分别对比较相等给出了一些协议,但对协议的安全性却没有证明;秦静等人针对以前的协议在"公平性"上考虑欠缺的不足,在文献[10]中提出了一个可以解决比较相等,并且同时解决公平性的协议;李顺东等人在文献[11]利用不经意传输协议和自己定义的特殊函数给出了一个效率比较高的解决两方比较的问题的协议,并给出了安全性证明和效率分析. 但以上的协议都是针对双方的,且各自拥有的数据是一个,我们没有见到过多方多数据的比较协议,本文在此进行探索.

收稿日期:2008-04-17;修回日期:2008-12-17
基金项目:国家 973 重点基础发展规划(No. 2007CB311203);国家自然科学基金(No. 60821001);北京市自然科学基金(No. 4073037);教育部博士点基金(No. 20060013007);西安电子科技大学综合业务网理论及关键技术国家重点实验室开放课题(No. ISN7-01)

对于一个集合 $S \subset \{1,2,\cdots,N\}$ 排序是指将所有 $x \in S$ 的数值按照从小到大的顺序排成一个序列,从而很容易确定任意一个数值 $x \in S$ 在该序列中的位置. 现假设有 $n$ 方 $P_1, P_2, \cdots, P_n$,分别拥有一个保密数据集合 $D_{P_1} = \{m_1^1, m_2^1, \cdots, m_{k_1}^1\}$, $D_{P_2} = \{m_1^2, m_2^2, \cdots, m_{k_2}^2\}$, $\cdots$, $D_{P_n} = \{m_1^n, m_2^n, \cdots, m_{k_n}^n\}$, $D_{P_1}, D_{P_2}, \cdots, D_{P_n} \subset \{1,2,\cdots,N\}$. 我们考虑将并集 $D = D_{P_1} \cup D_{P_2} \cup \cdots \cup D_{P_n}$ 中所有的数据按照从小到大的顺序安全地排成一个序列. 安全排序的含义是:在排序结束后,各方能够知道他们各自拥有的 $D_{P_i}$ 中的数据在并集 $D$ 中,按照小到大的顺序排成的序列中的位置,并且任意一方都不知道其它方数据集合的任何信息. 这个问题是姚氏百万富翁问题[2]的一个推广问题,本文我们将这个问题称为安全多方多数据排序问题(Secure Multi-party Multi-data Ranking Problem,缩写为 SMMR).

SMMR 问题有着很强的应用背景. 例如,有 $n$ 个经营相同产品的公司,如都生产 MP3(有各个系列,各种价位),他们想知道自己的不同系列 MP3 的销售额排名,在这种情况下,各个公司产品的销售额应该是保密的,最后按照销售额计算出的排名才是大家所关注的.

本文利用 RSA 密码体制提出了一个解决 SMMR 问题的协议,并在半诚实模型下对该协议的正确性和安全性给出了证明. 我们的协议可以保证:

(1) 公平性:指 $n$ 方可以同时独立计算并知道结果;

(2) 安全性:虽然使用了可信第三方 $T$,但是 $T$ 只负责随机数的产生和分发工作,并不参与计算;在 $n$ 方各自知道自己的结果时,并不能从计算的中间结果知道其他方的最终计算结果,甚至无法知道其他方有多少个数参与了计算,即对其他方的任何信息一无所知;

(3) 有效性:在多方多数据的排序中的某些情况下,本文提出的协议与多次使用 A. C. yao 的协议相比,在安全性和效率上都有很大提高.

本文假设参与排序的各方都是"半诚实的",即参与各方能严格执行协议的规程,不会中途强行退出或恶意掺入虚假数据. 但在协议执行过程中他们可能会保留所有能搜集到的关于其它参与方的信息,以期望在协议结束后推断出其它参与方的输入信息. 人们对安全多方协议的研究有很多是基于半诚实的,对半诚实模型下安全协议的研究是有意义的.

## 2 预备知识

### 2.1 信息论安全和密码学安全

信息论安全:若协议能够抵抗具有无限计算能力的攻击者,则称为信息论安全或者称为无条件安全. 协议若要达到无条件安全,则要求通信信道是安全的.

密码学安全:若协议只能抵抗具有多项式时间计算能力的攻击者,则称其为密码学安全,或者称为计算安全. 本协议就是满足密码学安全的.

### 2.2 计算不可区分[12]

设 $\Omega_k$ 包含在 $\{0,1\}^k$ 是一个 $k$ 比特数字的有限域. 设 $D_1 = D_1(\Omega_k)$ 和 $D_2 = D_2(\Omega_k)$ 分布在 $\Omega_k$ 上的. 对于给定的 $x \in \Omega_k$,设 $A_k(x)$ 是一个可以返回真或假的算法. 我们定义随机变量 $x \in \Omega_k$ 的分布 $D_1$ 和分布 $D_2$ 是计算不可区分的,如果对于任意多项式算法 $A_k(x)$,任意多项式 $p(k)$ 和任意足够大的数 $k$,都有:$\Pr[A_k(x) \mid x \sim D_1] - \Pr[A_k(x) \mid x \sim D_2] < 1/p(k)$ 成立(其中 $x \sim D_1$ 表示 $x$ 是根据 $D_1$ 分布的,$\Pr[A_k(x)]$ 是 $A_k(x)$ 返回值为真的可能性).

### 2.3 RSA 密码体制

RSA 密码体制是一种语义安全的同态公钥加密体制,其数学基础是初等数论中的 Euler 定理,其安全性建立在大整数因子分解的困难性之上.

在 RSA 密码体制中需要产生一个密钥,密钥的产生过程如下:选择两个互异的大素数 $p$ 和 $q$,计算 $n_r = p \times q$(公开),$\phi(n_r) = (p-1) \times (q-1)$(保密),选择一个随机整数 $e(0 < e < \phi(n_r))$,满足 $\gcd(e, \phi(n_r)) = 1$(公开). 计算 $d = e^{-1} \bmod \phi(n_r)$(保密). 确定公钥 $n_r, e$,私钥 $d$. 加密消息 $M$ 时,计算密文:$C = M^e \bmod n_r$;解密时,计算:$M = C^d \bmod n_r$.

RSA 密码体制是具有乘法同态性的,即假设对于任意的消息 $m$ 和 $m'$,它们的加密结果为 $m^e$ 和 $(m')^e$,则 $(mm')^e$ 表示对于消息 $mm'$ 的加密结果.

### 2.4 不经意传输协议

文献[13]首先提出了不经意传输的概念,文献[13]中的不经意传输实际是 $OT_2^1$. $OT_n^k$ 不经意传输是 $OT_2^1$ 不经意传输的发展,是一个重要的密码学协议,这个协议能够完成下面的任务:Alice 有 $n$ 个消息(或者数据) $\{m_1, m_2, \cdots, m_n\}$,通过执行 $OT_n^k$ 不经意传输协议,Bob 能够基于自己的选择得到且只能得到排在 $\{\sigma_1, \sigma_2, \cdots, \sigma_k\}$ 位置的 $k$ 个消息 $\{m_{\sigma_1}, m_{\sigma_2}, \cdots, m_{\sigma_k}\}$,而对其他消息一无所知. Alice 对 Bob 选择了哪 $k$ 个消息也一无所知.

文献[14]中的 $OT_n^k$ 不经意传输协议描述如下:

假定:$G_q$ 是 $Z_p^*$ 的子集,$q$ 和 $p = 2q + 1$ 都是素数,$g$ 和 $h$ 是 $G_q$ 的生成元. $x \in_R X$ 表示 $x$ 均一地独立地从集合 $X$ 中选出.

Alice 拥有信息 $\{m_1, m_2, \cdots, m_n\}$,Bob 想得到排在第 $\{\sigma_1, \sigma_2, \cdots, \sigma_k\}$ 位置的消息.

(1) Bob 选择两个多项式 $f(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1} + x^k$ 和 $f'(x) = b_0 + b_1 x + \cdots + b_{k-1} x^{k-1} + x^k$

这里 $a_0, a_1, \cdots, a_{k-1} \in_R Z_q$,并且 $b_0 + b_1 x + \cdots + b_{k-1} x^{k-1} + x^k \equiv (x - \sigma_1)(x - \sigma_2) \cdots (x - \sigma_k) \bmod q$.

(2) Bob →Alice:

$A_0 = g^{a_0} h^{b_0}, A_1 = g^{a_1} h^{b_1}, \cdots, A_{k-1} = g^{a_{k-1}} h^{b_{k-1}}$

(3) Alice 计算 $c_i = (g^{k_i}, m_i B_i^{k_i})$ 这里 $k_i \in_R Z_q^*$ 且 $B_i = g^{f(i)} h^{f'(i)} = A_0 A_1^i \cdots A_{k-1}^{i^{k-1}} (gh)^{i^k} \bmod p (i = 1, 2, \cdots, n)$

(4) Alice →Bob: $c_1, c_2, \cdots, c_n$

(5) 假定 $c_i = (U_i, V_i)$, $R$ 对每一个 $\sigma_i$ 计算 $m_{\sigma_i} = V_{\sigma_i} / U_{\sigma_i}^{f(\sigma_i)} \bmod p$.

协议执行完毕,Bob 就可以得到 $m_{\sigma_i} (i = 1, 2, \cdots, k)$,而 Alice 对 $\sigma_i (i = 1, 2, \cdots, k)$ 一无所知.

## 3　半诚实模型下安全多方多数据排序协议

输入:$n$ 方 $P_1, P_2, \cdots, P_n$ 分别拥有一个已经从小到大排好序的数据集合:$D_{P_1} = \{m_1^1, m_2^1, \cdots, m_{k_1}^1\}$, $D_{P_2} = \{m_1^2, m_2^2, \cdots, m_{k_2}^2\}$, $\cdots$, $D_{P_n} = \{m_1^n, m_2^n, \cdots, m_{k_n}^n\}$ ($D_{P_i} \subset \{1, 2, \cdots, N\}$).

输出:$P_1, P_2, \cdots, P_n$ 各自得到 $D_{P_1}, D_{P_2}, \cdots, D_{P_n}$ 的数据在并集 $D$ 中位置序列(从小到大) $R_{P_1} = \{r_1^1, r_2^1, \cdots, r_{k_1}^1\}$. $R_{P_2} = \{r_1^2, r_2^2, \cdots, r_{k_2}^2\}$, $\cdots$, $R_{P_n} = \{r_1^n, r_2^n, \cdots, r_{k_n}^n\}$.

本协议的实现思想如下:首先由可信中心产生随机数和公私钥并分发;第 1 步各个参与方根据自己拥有的数据产生 $N$ 维的向量并加密,分发加密后的数据;第 2 步各个参与方收到其他方发来的消息后计算,并发给其他各方解密;第 3,4 步,各参与方用自己的私钥解密,并用不经意传输协议从可信中心 T 得到与自己拥有的数据相对应的随机数,计算得到自己拥有数据的排序.

协议准备,可信中心 T 做如下工作:

(1) 产生 RSA 加密算法所需要的公钥 $n_r, e$,和私钥 $d$,并将 $d$ 分成 $n$ 份即 $d = \sum_{i=1}^{n} d_i$. 将 $n_r, e$ 公开,并将 $d_i$ 通过秘密通道发给 $P_i$.

(2) 选取 $N$ 组整数,每组包括 $n$ 个数$\{l_{1i}, l_{2i}, \cdots, l_{ni}\} (i = 1, 2 \cdots, N)$,并计算 $L_i = l_{1i} \times l_{2i} \times \cdots \times l_{ni} (i = 1, 2, \cdots, N)$,将这 $N$ 组整数排成一个矩阵,如下:

$$\begin{bmatrix} l_{11}, l_{12}, \cdots\cdots, l_{1i}, \cdots\cdots l_{1N} \\ l_{21}, l_{22}, \cdots\cdots, l_{2i}, \cdots\cdots l_{2N} \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ l_{i1}, l_{i2}, \cdots\cdots, l_{ii}, \cdots\cdots l_{iN} \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ l_{n1}, l_{n2}, \cdots\cdots, l_{ni}, \cdots\cdots l_{nN} \end{bmatrix}$$

显然,第 $i$ 列的整数相乘是 $L_i$ 的值. 可信中心 T 将矩阵的第 $i$ 行 $(l_{i1}, l_{i2}, \cdots, l_{iN})$ 发给 $P_i (i = 1, 2, \cdots, N)$. $L_1, L_2, \cdots, L_N$ 由可信中心保留,不公布.

(3) 可信中心选定一个随机数 $\rho$,发给各个参与方 $P_1, P_2, \cdots, P_n$.

协议准备完毕,下面是各个参与方共同执行协议的过程:

(1) 对于 $i = 1, 2, \cdots, n$ 各个参与方 $P_i$ 执行以下步骤:

(a) 根据数据集 $D_{P_i} = \{m_1^i, m_2^i, \cdots, m_{k_i}^i\}$ 产生一个 $N$ 维向量 $v_i = (\rho^{b_{i1}}, \cdots, \rho^{b_{im_1^i}}, \cdots, \rho^{b_{ik}}, \cdots, \rho^{b_{im_{k_i}^i}}, \cdots, \rho^{b_{iN}})$. 该向量 $v_i$ 的各分量组成情况描述如下:

当 $1 \le k < m_1^i$ 时, $b_{ik} = 0$;

当 $m_1^i \le k < m_2^i$ 时, $b_{ik} = 1$;

当 $m_2^i \le k < m_3^i$ 时, $b_{ik} = 2$; $\cdots$;

当 $m_{k_i}^i \le k \le N$ 时, $b_{ik} = k_i$;

(b) 对于向量 $v_i$ 的各个分量(共 $N$ 个),结合随机数,利用 RSA 算法,用公钥加密得:$v_i' = ((\rho^{b_{i1}} l_{i1}))^e, \cdots, (\rho^{b_{im_1^i}} l_{im_1^i})^e, \cdots, (\rho^{b_{ik}} l_{ik})^e, \cdots, (\rho^{b_{im_{k_i}^i}} l_{im_{k_i}^i})^e, \cdots, (\rho^{b_{iN}} l_{iN})^e)$ 然后将序列 $v_i'$ 发送给其它参与方 $P_1, P_2, \cdots, P_{i-1}, P_{i+1}, \cdots, P_n$;

(2) 对于 $j = 1, 2, \cdots, n$ 各个参与方 $P_j$ 执行以下步骤:

(a) 将收到的 $n - 1$ 个序列 $v_1', v_2', \cdots, v_{j-1}', v_{j+1}', \cdots, v_n'$ 和 $v_j'$ 作为行向量组成一个 $n \times N$ 的矩阵如下:

$$A_j = \begin{pmatrix} (\rho^{b_{11}} l_{11})^e, \cdots, (\rho^{b_{1m_1^j}} l_{1m_1^j})^e, \cdots, (\rho^{b_{1m_{k_j}^j}} l_{1m_{k_j}^j})^e, \cdots, (\rho^{b_{1N}} l_{1N})^e \\ (\rho^{b_{21}} l_{21})^e, \cdots, (\rho^{b_{2m_1^j}} l_{2m_1^j})^e, \cdots, (\rho^{b_{2m_{k_j}^j}} l_{2m_{k_j}^j})^e, \cdots, (\rho^{b_{21}} l_{21})^e \\ \cdots\cdots \\ (\rho^{b_{n1}} l_{1n1})^e, \cdots, (\rho^{b_{nm_1^j}} l_{nm_1^j})^e, \cdots, (\rho^{b_{nm_{k_j}^j}} l_{nm_{k_j}^j})^e, \cdots, (\rho^{b_{n1}} l_{n1})^e \end{pmatrix}$$

(b) 将 $A_j$ 的第 $m_1^j, m_2^j, \cdots, m_{k_j}^j$ ($m_1^j, m_2^j, \cdots, m_{k_j}^j$ 为 $P_j$ 拥有的数据)列元素相乘$(s = 1, 2, \cdots, k_j)$,结果组成向量得:

$E_j = ((\rho^{\sum_{i=1}^{n} b_{im_1^j}} L_{m_1^j})^e, (\rho^{\sum_{i=1}^{n} b_{im_2^j}} L_{m_2^j})^e, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j}} L_{m_{k_j}^j})^e)$

(c) 选择随机数 $h_j$ 以及 $h_j$ 个随机数 $r_1^j, \cdots, r_{h_j}^j$,与 $E_j$ 组成向量$(r_1^j, \cdots, r_{h_j}^j, (\rho^{\sum_{i=1}^{n} b_{im_1^j}} L_{m_1^j})^e, (\rho^{\sum_{i=1}^{n} b_{im_2^j}} L_{m_2^j})^e, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j}} L_{m_{k_j}^j})^e)$ 发送给 $P_1, P_2, \cdots, P_{j-1}, P_{j+1}, \cdots, P_n$;

(3) 对于 $k = 1, 2, \cdots, n$ 参与方 $P_k$ 收到 $P_j$ 发来的序列$(r_1^j, \cdots, r_{h_j}^j (\rho^{\sum_{i=1}^{n} b_{im_1^j}} L_{m_1^j})^e, (\rho^{\sum_{i=1}^{n} b_{im_2^j}} L_{m_2^j})^e, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j}} L_{m_{k_j}^j})^e)$ 后,用自己的子密钥 $d_k$ 对其进行解密,得到结果$((r_1^j)^{d_k}, \cdots (r_{h_j}^j)^{d_k}, (\rho^{\sum_{i=1}^{n} b_{im_1^j}} L_{m_1^j})^{ed_k}, (\rho^{\sum_{i=1}^{n} b_{im_2^j}} L_{m_2^j})^{ed_k}, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j}} L_{m_{k_j}^j})^{ed_k})$ 后将结果发给 $P_j$;

(4) 对于 $j = 1, 2, \cdots, n$ 参与方 $P_j$ 执行以下步骤:

(a) 收到从其他 $k = 1, 2, \cdots, j-1, j+1, \cdots, n$ 方发过来$((r_1^j)^{d_k}, \cdots, (r_{h_j}^j)^{d_k}, (\rho^{\sum_{i=1}^{n} b_{im_1^j}} L_{m_1^j})^{ed_k}, (\rho^{\sum_{i=1}^{n} b_{im_2^j}} L_{m_2^j})^{ed_k}, \cdots,$

$(\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j} L_{m_{k_j}^j}})^{ed_k}$) 将他们作为行向量排成一个矩阵如下:

$$B_j = \begin{pmatrix} (r_1^j)^{d_1}, \cdots, (r_{h_j}^j)^{d_1}, (\rho^{\sum_{i=1}^{n} b_{im_1^j} L_{m_1^j}})^{ed_1}, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j} L_{m_{k_j}^j}})^{ed_1} \\ \cdots \\ (r_1^j)^{d_k}, \cdots, (r_{h_j}^j)^{d_k}, (\rho^{\sum_{i=1}^{n} b_{im_1^j} L_{m_1^j}})^{ed_k}, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j} L_{m_{k_j}^j}})^{ed_k} \\ \cdots \\ (r_1^j)^{d_n}, \cdots, (r_{h_j}^j)^{d_n}, (\rho^{\sum_{i=1}^{n} b_{im_1^j} L_{m_1^j}})^{ed_n}, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j} L_{m_{k_j}^j}})^{ed_n} \end{pmatrix}$$

对矩阵中 $f = h_j + 1, h_j + 2, \cdots, h_j + k_j$ 列的数据计算乘积并用自己的子密钥 $d_j$ 解密:

$$((\rho^{\sum_{i=1}^{n} b_{im_1^j} L_{m_1^j}})^{e_k \sum_{i=1}^{n} d_k}, (\rho^{\sum_{i=1}^{n} b_{im_2^j} L_{m_2^j}})^{e_k \sum_{i=1}^{n} d_k}, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j} L_{m_{k_j}^j}})^{e_k \sum_{i=1}^{n} d_k})$$

$$= ((\rho^{\sum_{i=1}^{n} b_{im_1^j} L_{m_1^j}})^{ed}, (\rho^{\sum_{i=1}^{n} b_{im_2^j} L_{m_2^j}})^{ed}, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j} L_{m_{k_j}^j}})^{ed})$$

$$= (\rho^{\sum_{i=1}^{n} b_{im_1^j} L_{m_1^j}}, \rho^{\sum_{i=1}^{n} b_{im_2^j} L_{m_2^j}}, \cdots, \rho^{\sum_{i=1}^{n} b_{im_{k_j}^j} L_{m_{k_j}^j}})$$

(b) $P_j$ 利用 $OT_N^{k_j}$ 协议,从可信中心 T 取到 $L_{m_s^j}$ ($s = 1, 2, \cdots, k_j$);

(c) 将 $(\rho^{\sum_{i=1}^{n} b_{im_1^j} L_{m_1^j}}, \rho^{\sum_{i=1}^{n} b_{im_2^j} L_{m_2^j}}, \cdots, \rho^{\sum_{i=1}^{n} b_{im_{k_j}^j} L_{m_{k_j}^j}})$ 中的 $L_{m_k^j}$ ($s = 1, 2, \cdots, k_j$) 全部消去得到 $(\rho^{\sum_{i=1}^{n} b_{im_1^j}}, \rho^{\sum_{i=1}^{n} b_{im_2^j}}, \cdots, \rho^{\sum_{i=1}^{n} b_{im_{k_j}^j}})$,并计算 $\rho^h$ ($h = 1, 2, \cdots, k_1 + k_2 + \cdots + k_n$);

(d) 将 $(\rho^{\sum_{i=1}^{n} b_{im_1^j}}, \rho^{\sum_{i=1}^{n} b_{im_2^j}}, \cdots, \rho^{\sum_{i=1}^{n} b_{im_{k_j}^j}})$ 每一项与 $\rho^h$ 进行比较,则 $P_j$ 可以得到 $D_{P_j}$ 中的数据在 $D$ 中的所排的位置的集合 $R_{P_j}$.

## 4 协议的性能分析

### 4.1 正确性证明

**定理 4.1** 在半诚实模型中,在 RSA 密码体制的语义安全性假设下,上述 SMMR 协议是正确的.

证明:容易看出在协议第 1 步生成的向量 $v_i$ 的分量中,$b_{ik}$ 的含义可以理解为,在 $D_{P_i}$ 中有 $b_{ik}$ 个数小于等于当前位置 $k$. 所以 $E_j = ((\rho^{\sum_{i=1}^{n} b_{im_1^j}} L_{m_1^j})^e, (\rho^{\sum_{i=1}^{n} b_{im_2^j}} L_{m_2^j})^e, \cdots, (\rho^{\sum_{i=1}^{n} b_{im_{k_j}^j}} L_{m_{k_j}^j})^e)$ 中的 $\sum_{i=1}^{n} b_{im_1^j}, \sum_{i=1}^{n} b_{im_2^j}, \cdots, \sum_{i=1}^{n} b_{im_{k_j}^j}$ 分别表示在数据集合 $D$ 中有 $\sum_{i=1}^{n} b_{im_1^j}, \sum_{i=1}^{n} b_{im_2^j}, \cdots, \sum_{i=1}^{n} b_{im_{k_j}^j}$ 个数据分别小于等于 $m_{m_1^j}^j, m_{m_2^j}^j, \cdots, m_{m_{k_j}^j}^j$. 即 $\sum_{i=1}^{n} b_{im_1^j}, \sum_{i=1}^{n} b_{im_2^j}, \cdots, \sum_{i=1}^{n} b_{im_{k_j}^j}$ 分别为 $(m_{m_1^j}^j, m_{m_2^j}^j, \cdots, m_{m_{k_j}^j}^j)$ 在 $D = D_{P_1} \cup D_{P_2} \cup \cdots \cup D_{P_n}$ 中所有数据按从小到大顺序所排序列中所排的位置 $(r_{m_1^j}^j, r_{m_2^j}^j, \cdots, r_{m_{k_j}^j}^j)$. 接着协议 3-4 步实际上是 $n$ 个参与方联合解密得到 $\sum_{i=1}^{n} b_{im_1^j}, \sum_{i=1}^{n} b_{im_2^j}, \cdots, \sum_{i=1}^{n} b_{im_{k_j}^j}$ 的过程.

### 4.2 安全性分析

在半诚实模型下 SMMR 协议是安全的,指的是各个参与方 $P_i$ 除了知道自己的数据集合 $D_{P_i} = \{m_1^i, m_2^i, \cdots, m_{k_i}^i\}$ 中各个数据的的排序序列 $R_{P_i} = (r_1^i, r_2^i, \cdots, r_{k_i}^i)$ 外,均不能从自己的输入、输出以及在计算过程中搜集到的中间结果中得到关于其它参与方输入的任何信息. 我们采用反证法进行证明. 假设我们所设计的协议是不安全的,即存在一个概率多项式时间的敌手 $D$,该敌手可以从协议计算过程中获得关于其它参与方的输入信息.

假设欺骗方为 $P_1, P_2, \cdots, P_n$ 中的一个或几个,不妨假设为 $\Omega \subset \{P_1, P_2, \cdots, P_n\}$. 敌手 $D$ 可以控制这些欺骗方,通过这些欺骗方知道一些信息,这些信息包括:通过 $P_i \in \Omega$ 自己计算得到的信息: $D_{P_i}, d_i, v_i, v_i', E_i, E_i', \rho^h$ 以及某一参与方 $P_j \notin \Omega$ 向 $P_i \in \Omega$ 发送的信息 $v_j', E_j'$.

假设敌手 $D$ 可以通过 $P_j \notin \Omega$ 向 $P_i \in \Omega$ 发送的信息获得关于参与方 $P_j$ ($P_j \notin \Omega$) 的输入 $D_{P_j}$ 的信息,那么它应该可以区分下面的两种情况:其它 $n-1$ 参与方输入 $\{m_1^1, m_2^1, \cdots, m_{k_1}^1\}, \cdots, \{m_1^{j-1}, m_2^{j-1}, \cdots, m_{k_{j-1}}^{j-1}\}, \{m_1^{j+1}, m_2^{j+1}, \cdots, m_{k_{j+1}}^{j+1}\}, \cdots, \{m_1^n, m_2^n, \cdots, m_{k_n}^n\}$,各自的子私钥 $d_i$ 和公钥 $n_r, e$ 都是确定的的情况下,参与方 $P_j$ ($P_j \notin \Omega$) 分别输入为 $\{m_1^{j\prime}, m_2^{j\prime}, \cdots, m_{k_j}^{j\prime}\}$ 和 $\{m_1^{j\prime\prime}, m_2^{j\prime\prime}, \cdots, m_{k_j}^{j\prime\prime}\}$,且 $\{m_1^{j\prime\prime}, m_2^{j\prime\prime}, \cdots, m_{k_j}^{j\prime\prime}\} \neq \{m_1^{j\prime}, m_2^{j\prime}, \cdots, m_{k_j}^{j\prime}\}$. 在这两种情况中,敌手 $D$ 不可能从 $P_i \in \Omega$ 自己计算得到的信息 $D_{P_i}, d_i, v_i, v_i', E_i, E_i', \rho^h$ 中区分这两种情况. 所以敌手 $D$ 只能从序列 $v_j', E_j'$ 区分这两种情况.

如果敌手 $D$ 可以区分以上两种情况,因为 $v_j', E_j'$ 是用 RSA 加过密的,所以它就可以区分两种情况的 $v_j', E_j'$ 哪一个分别对应 $\{m_1^{j\prime}, m_2^{j\prime}, \cdots, m_{k_j}^{j\prime}\}, \{m_1^{j\prime\prime}, m_2^{j\prime\prime}, \cdots, m_{k_j}^{j\prime\prime}\}$ 的密文. 到此我们就构建成功了一个可以破坏 RSA 密码体制语义安全性的敌手 $D$,显然这与 RSA 密码体制的语义安全性假设是相矛盾.

另外,第 2 步的 (c) 中的 $(E_j')_{j=1}^n$ 中加入了随机数,其余的参与方甚至无法知道 $P_j$ 有几个数据参与了排序.

### 4.3 效率分析

对于多方多数据的比较,目前没有看到解决这种问题的协议. 与多次使用 A. C. yao[2] 的协议相比较,本文所提出的协议在某些情况 ($N$ 很小但 $D$ 很大) 效率上要高.

#### 4.3.1 计算复杂度

使用本文的所提出的协议,对于每个参与方 $P_i$,协议的第 1 步进行 $N$ 次加密运算;第三、四步 $n$ 方联合进

行 $k_1 + k_2 + \cdots + k_n$ 个密文;第四步的不经意传输加解密次数为 $n[N + k_1 + k_2 + \cdots + k_n]$,所以总的次数为 $2nN + (n+1) + (k_1 + k_2 + \cdots + k_n)$,复杂度为 $o(2nN + (n+1) + (k_1 + k_2 + \cdots + k_n))$

一次使用 Yao 的协议,复杂度为 $o(N)$($N$ 为输入的数的范围),而比较好的排序算法的复杂度一般为 $o(n\lg n)$,所以最终的复杂度为 $o(N(k_1 + k_2 + \cdots + k_n)\lg(k_1 + k_2 + \cdots + k_n))$;

当各方拥有的数据数总和 $k_1 + k_2 + \cdots + k_n$ 远远大于 $N$ 时,显然本文所提出的协议要比多次使用 A. C. yao 的协议的在计算复杂度上效率高. 分析结果如表 1 所示.

表 1　两个协议的比较

| | 安全性 | 公平性 | 效率 |
|---|---|---|---|
| 多次使用 A. C. yao 的协议 | 不仅知道自己的排序,还能知道谁的数据比自己的小 | 两两比较时,需要一方告诉另一方比较结果 | 在数据数总和很大的情况下,效率迅速降低 |
| 本文的协议 | 仅知道自己的排序位置,无法知道其他任何信息 | 各方同时知道排序的结果 | 在数据数总和很大的情况下,效率高一些 |

## 5　结论

本文我们考虑了一个百万富翁问题的推广问题. 这个问题包含有 $n$ 方 $P_1, P_2, \cdots, P_n$,每个 $P_i$ 都拥有一个保密的输入数据集合 $D_{P_1}, D_{P_2}, \cdots, D_{P_n} \subset \{1, 2, \cdots, N\}$. 所有参与方都希望在不泄漏关于自己输入数据的任何信息的情况下得到自己输入数据在这多个数据集的并集 $D = D_{P_1} \cup D_{P_2} \cup \cdots \cup D_{P_n}$ 中的所有数据按从小到大排序队列中的所排的位置. 本文我们给出了一个在半诚实模型下基于 RSA 密码体制的安全性假设的 SMMR 协议. SMMR 问题的有效解决又使得在线交易、拍卖、竞标等成为可能,我们相信这将在电子商务中有很大的应用前景.

**参考文献**:


[1] O Goldreich. Secure Multi-party Computation (working draft) [EB/OL]; http://www.wisdom.weizmann.ac.il/～oded/pp.html,2000-10.

[2] A Yao. Protocol for secure computations [A]. Proceeding of 23rd IEEE Symposium on Foundations of Computer Science [C]. Los Alamitos, CA: IEEE Computer Society Press, 1982. 160-164.

[3] Du W L, Atallah MJ. Privacy-preserving cooperative scientific computations[A]. Proceedings of the 14th IEEE Computer Security Workshop [C]. Nova Scotia, Canada: IEEE Computer Society Press, 2001. 273-282.

[4] Mikhail J Atallah, Wenliang Du. Secure multi-party computational geometry [A]. In Lecture Notes in Computer Science 2125, [C]Berlin: Springer, 2001. 165-179.

[5] Lindell, Y. and Pinkas. B. Privacy preserving data mining[J]. Journal of Cryptology 2002, 15(3): 177-206.

[6] Du W L, Atallah M J. Privacy-preserving statistical analysis [A]. Proceedings of the 17th Annual Computer Security Applications Conference [C]. New Orleans, Louisiana, USA: IEEE Computer Society Press, 2001. 102-110.

[7] Cachin C. Efficient private bidding and auctions with an oblivious third party[A]. Proceeding of the 6th ACM Conference on Computer and Communications Security [C]. New York: ACM Press, 1999. 120-127.

[8] Fagin R, Naor M, Winkler P. Comparing information without leaking it[J]. Communications of the ACM, 1996, 39(5): 77-85.

[9] Schneier B. Applied Cryptography: Protocols, , Algorithms, and Source Code in C. 2nd ed[M]. USA: John Wiley&Sons, Inc, 1996.

[10] 秦静,张振峰,冯登国,李宝. 一个特殊的安全双方计算协议 [J]. 通信学报, 2004, 25(11): 39-46.
Qin Jing, Zhang Zhenfeng, Feng Dengguo, Li Bao. A protocol of specific secure two-party computation[J]. Journal on Communica-tions, 2004, 25(11): 39-46. (in Chinese)

[11] 李顺东,戴一奇,游启友. 姚氏百万富翁问题的高效解决方案[J]. 电子学报, 2005, 33(5): 769-773.
Li Shundong, Dai Yiqi, You Qiyou. An efficient solution to Yao's millionaires' problem [J]. Asta Sinica Electronica, 2005, 33(5): 769-773. (in Chinese)

[12] R Agrawal, A Evfimievski, R Srikant. Information sharing across private databases [A]. Proceedings of the 2003 ACM SIGMOD international conference on Management of Data [C]. San Diego, CA: ACM Press, 2003. 86-97.

[13] M Rabin. How to exchange secrets by oblivious transfer[R]. USA: Aiken Computation Laboratory, Harvard Univ, 1981.

[14] Wen-Guey Tzeng: Efficient 1-Out-of-n oblivious transfer schemes with universally usable parameters[J]. IEEE Transactions on Computers, 2004, 53(2): 232-240.


**作者简介**:


**邱　梅**　女,1980 年生于山东济宁,北京邮电大学硕士研究生,主要研究方向为信息安全、安全多方计算.
E-mail: qiumei567@gmail.com

**罗守山**　男,1962 年生于北京市. 1985 年、1994 年和 2001 年分别在北京师范大学、北京邮电大学和北京邮电大学获理学学士、理学硕士和工学博士学位. 现为北京邮电大学教授,博士生导师,主要从事信息安全、密码学等方面的研究工作. E-mail: buptlou@263.net