doi :10.3969/j.issn.1671-1122.2013.09.001

基于分布式线性方程组求解的安全多方计算协议

陈莉，林柏钢

（福州大学数学与计算机科学学院，福建福州 350108）

摘 要：不同组织之间的合作关系常用求解线性方程组或线性不等式组等线性代数问题来建模。当合作过程中涉及到数据的隐私保护时，一般直接求解线性方程组的方法将不再适用。文章为解决此类问题，基于同态加密方式设计了求解分布式线性方程组的安全两方计算协议和安全多方计算协议。与以往基于茫然传输协议设计的安全计算协议相比，降低了协议的通信复杂度。文章给出了协议的正确性、安全性和复杂度分析。

关键词：安全多方计算；同态加密；分布式线性方程组；安全计算协议

中图分类号：TP309 **文献标识码：**A **文章编号：**1671-1122（2013）09-0002-04

Secure Protocols for Resolving Distributed System of Linear Equations

CHEN Li,LIN Bo-gang

(*College of Mathematics and Computer Science,Fuzhou University,Fuzhou Fujian* 350108, *China*)

Abstract: The cooperation between different organizations is often modeled as solving linear algebra problems, such as linear system of equations or linear inequalities. If participants concern about the data privacy in the process of cooperation, common method of directly solving linear system of equations will no longer apply. This paper designs a secure two-party protocol and a secure multi-party protocol based on homomorphic encryption scheme for solving distributed linear system of equations. Compared to the protocols based on oblivious transfer protocol, the communication complexity is reduced. This paper also gives the correctness, security and complexity analysis.

Key words: secure multi-party computation; homomorphic encryption; distributed system of linear equations; secure computation protocols

0 引言

随着社会的发展，经济的全球化，越来越多的共赢需要通过合作来实现。但是合作的过程涉及到自己的隐私和利益保护时，那么这种合作将受到限制。例如，多个经济组织为了共同的利益计划在某一个项目上进行合作，每个组织都希望自身的需求得到最大满足，然而这些需求可能反映了该组织的一些财务状况，经济统计特性，战略规划和某些预测信息（如利率、通货膨胀率以及某些商品价格的未来演变）等。每个经济组织都不希望将这些有价值的私有数据泄露给其他参与者，甚至是可信第三方。那么这多个经济组织之间该如何在共有关系项目上进行合作，而又不泄露自己的隐私信息呢?

对于解决类似上述问题，Yao 最早在文献 [1] 中提出了安全多方计算（secure multi-party computation，SMC）概念。简单地说是指在一个互不信任的多用户网络中，各用户通过网络来协同完成一个可靠的计算任务，同时又保持各自数据的安全性。但是理论上存在的通用解决方法往往在实际计算效率上是不可行的，对于一些具体的问题需要进行特定协议的设计。因此，针对安全多方计算问题研究高效实用的安全多方计算协议是目前的热门课题之一。该课题属于安全多方计算研究的一个特定领域：保护隐私的科学计算 [2-5]。基于茫然传输协议，Du 和 Atallah 最早在两方情形下给出了一系列分布式线性代数问题的解决方案 [2]，如求解两方线性方程组、线性规划等。罗文俊、李祥在文献 [3] 中将 Du 等人的部分工作推广到多方情形，同样基于茫然传输协议设计了多方安全矩阵乘积协议，并以此设计了求解多方安全线性方程组的计算协议。由于茫然传输协议的通信复杂度高，本文

收稿日期：2013-07-21

基金项目：国家自然科学基金 [60175022]、福建省安全课题 [822711]

作者简介：陈莉（1988-），女，福建，硕士研究生，主要研究方向：安全多方计算；林柏钢（1954-），男，福建，教授，博士生导师，主要研究方向：编码与密码。

基于同态的公钥加密体制，设计了求解分布式线性方程组的安全两方计算协议和安全多方计算协议，降低了协议的通信复杂度。文中还对协议的正确性、安全性以及复杂度进行了分析。

1 背景知识

1.1 合作模型

参与方 Alice 和 Bob 的合作形式根据矩阵的共享形式的不同而不同[2]。如图 1 所示。实际情况中，(b) 和 (c) 两种合作模型与 (d) 相比更具有意义，但它们均是 (d) 的特殊形式。故而在本文中两方安全协议解决的是(d)所示的混合合作形式。推广到多方合作情形为$[M_1+M_2+...+M_n]X=[b_1+b_2+...+b_n]$，其中$M_i$和$b_i$分别为参与方$P_i(1\leqslant i\leqslant n)$的私有矩阵和私有向量。共同合作的参与方最终得到线性方程组的解 X，且每一个参与方都无法得知其他参与方的秘密输入信息。

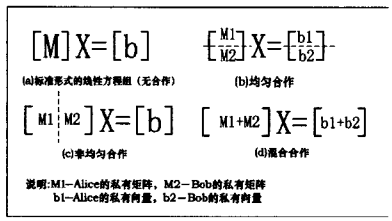

图1 两方合作模型

1.2 半诚实模型

半诚实模型是指协议中所有参与方都是半诚实的。所谓半诚实是指参与方能够严格执行协议规程，不会中途强行退出或者恶意参入虚假数据，但却可以保留自己的计算记录，并可能通过这些记录尽力去获得其他参与方的秘密信息。本文假设的参与方都是半诚实的。

1.3 基于语义安全的同态加密体制

设加密算法为 $E(\cdot)$，相应的解密算法为 $D(\cdot)$，其中加密密钥 pk 公开，解密密钥 sk 保密。明文空间 $M\subseteq Z$，$E(\cdot)$ 满足下述两个性质：

1) 语义安全性。对任意两个消息$m_1,m_2\in M$，不存在任何多项式时间算法区分 $E(m_1),E(m_2)$；

2) 加法同态性。对任意两个消息$m_1,m_2\in M$，任意常数$k\in Z$，若$m_1+m_2\in M$，且$km_1\in M$，则$E_{pk}(m_1)+_h E_{pk}(m_2)=E(m_1+m_2)$（“$+_h$”为加法同态运算符），$k\times_h E_{pk}(m_1)=E(km_1)$（“$\times_h$”为乘法同态运算符）。

本文以下分析采用的是 Paillier 的同态密钥方案[6]。首先考虑同态性质可以推广到矩阵运算上：

令向量$\mathbf{v}\in F^n$是一个列向量（n为向量 $\mathbf{v}$ 的维数），行向量用$\mathbf{v}^T$表示。$Enc(\mathbf{v})$表示对向量 $\mathbf{v}$ 里的每个元素进行加密。例如，$\mathbf{v}=\langle a_1,...,a_n\rangle$，其中$a_1,a_2,...,a_n\in F$，那么$Enc(\mathbf{v})=\langle Enc(a_1),...,Enc(a_n)\rangle$。类似地对于矩阵$\mathbf{A}\in F^{m\times n}$，用$Enc(\mathbf{A})$表示对矩阵中的每个元素进行加密，即$Enc(\mathbf{A})[i,j]=Enc(\mathbf{A}[i,j])$。根据同态加密体制的性质，在不知解密密钥的情况下可以进行以下操作：(1) 当给定两个加密向量$Enc(\mathbf{v}_1)$和$Enc(\mathbf{v}_2)$，可以有效计算$Enc(\mathbf{v}_1+\mathbf{v}_2)$。(2) 当给定一个加密向量$Enc(\mathbf{v})$和一个常数$c\in F$，可以有效计算$Enc(c\mathbf{v})$。(3) 当给定一个加密矩阵$Enc(\mathbf{A})$和一个矩阵 $\mathbf{B}$，可以有效计算$Enc(\mathbf{AB})$和$Enc(\mathbf{BA})$。(1)和(2)对于矩阵也是类似的。

1.4 复杂度分析

Alice 和 Bob 之间的一次交互称为一轮通信。协议的轮复杂度即为两者之间交互的次数。在每一轮通信中，Alice 和 Bob 之间会有数据进行传递（Alice 发送给 Bob，或 Bob 发送给 Alice）。那么协议的通信复杂度即为在整个协议的执行过程中，Alice 和 Bob 传送的数据总量（总的比特数）。本文约定协议的通信复杂度为 Alice 和 Bob 之间传送的加密数据的个数。对于计算代价，本文通过统计协议中使用的同态加密方案的加解密次数以及调用同态加法、同态乘法的次数来衡量。

2 线性方程组模型的安全两方计算协议

2.1 问题描述

两方安全线性方程组问题：Alice 有一个矩阵 $\mathbf{A}_1$ 和一个向量 $\mathbf{v}_1$，Bob 有一个矩阵 $\mathbf{A}_2$ 和一个向量 $\mathbf{v}_2$。$\mathbf{A}_1$，$\mathbf{A}_2$ 是 $n\times n$ 维矩阵，$\mathbf{v}_1$，$\mathbf{v}_2$ 是 n 维向量。Alice 和 Bob 希望在不泄露各自保密输入信息的情况下，求解线性方程组 $(\mathbf{A}_1+\mathbf{A}_2)\mathbf{x}=\mathbf{v}_1+\mathbf{v}_2$。

2.2 安全解决的设计思路

在不考虑安全的情况下，最直接的解决方法即是其中一方（如 Bob）将矩阵 $\mathbf{A}_2$ 和向量 $\mathbf{v}_2$ 发送给对方（Alice），由 Alice 单独求解线性方程组的解，再将结果发送给 Bob。但若 Bob 考虑到自己数据的隐私性，这种直接的解决方案就不适用了。那么 Bob 需要做的即是在将自己的数据发送之前先进行数据的伪装，使得 Alice 无法推导出他的数据。$\mathbf{J}$，$\mathbf{Q}$ 是 Bob 随机选取的两个可逆矩阵，用来对自己的数据进行伪装。由于$(\mathbf{A}_1+\mathbf{A}_2)\mathbf{x}=\mathbf{v}_1+\mathbf{v}_2$的解与$\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q}\mathbf{Q}^{-1}\mathbf{x}=\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2)$的解是等价的，如果 Alice 能获得$\mathbf{A}'=\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q}$和$\mathbf{v}'=\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2)$，那么 Alice 可以单独求解线性方程组$\mathbf{A}'\hat{\mathbf{x}}=\mathbf{v}'$。这样可通过计算$\mathbf{x}=\mathbf{Q}\hat{\mathbf{x}}$恢复出原始方程组的解。如何使 Alice 获得 $\mathbf{A}'$ 和 $\mathbf{v}'$ 是关键需要解决的，本文采用同态加密方案进行构建 $\mathbf{A}'$ 和 $\mathbf{v}'$。

2.3 协议设计

输入：Alice 有一个矩阵$\mathbf{A}_1\in F^{n\times n}$和一个列向量$\mathbf{v}_1\in F^n$；Bob 有一个矩阵$\mathbf{A}_2\in F^{n\times n}$和一个列向量$\mathbf{v}_2\in F^n$。

输出：Alice 和 Bob 获得 $\mathbf{x}$，满足线性方程组$(\mathbf{A}_1+\mathbf{A}_2)\mathbf{x}=\mathbf{v}_1+\mathbf{v}_2$。

1) Alice 生成它的同态公钥加密系统的密钥 (pk,sk)，利用 pk 计算加密矩阵 $Enc(\mathbf{A}_1)$ 和 $Enc(\mathbf{v}_1)$，并将pk, $Enc(\mathbf{A}_1)$, $Enc(\mathbf{v}_1)$发送给 Bob。

2) Bob 秘密随机选取可逆矩阵$\mathbf{J},\mathbf{Q}\in F^{n\times n}$，利用$pk$计算加密矩阵 $Enc(\mathbf{A}_2)$ 和 $Enc(\mathbf{v}_2)$。根据同态加密体制的特性，计算$Enc(\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q})$和$Enc(\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2))$，并将两个计算结果发送给 Alice。

3) Alice 利用密钥进行解密获得$\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q}$和$\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2)$，并求解线性方程组$\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q}\hat{\mathbf{x}}=\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2)$，Alice 将$\hat{\mathbf{x}}$传送给 Bob。

4) Bob 计算$\mathbf{x}=\mathbf{Q}\hat{\mathbf{x}}$，并将结果发送给 Alice。

2.4 协议分析

1) 正确性分析

根据加密体制的同态性，Bob 使用同态加法可计算$Enc(\mathbf{A}_1)+Enc(\mathbf{A}_2)=Enc(\mathbf{A}_1+\mathbf{A}_2)$和$Enc(\mathbf{v}_1)+Enc(\mathbf{v}_2)=Enc(\mathbf{v}_1+\mathbf{v}_2)$，再根据同态加法和同态数乘可计算获得$Enc(\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q})$和$Enc(\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2))$。那么 Alice 通过解密即可单独求解线性方程组$\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q}\hat{\mathbf{x}}=\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2)$。由于$(\mathbf{A}_1+\mathbf{A}_2)\mathbf{x}=\mathbf{v}_1+\mathbf{v}_2$的解与$\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q}\mathbf{Q}^{-1}\mathbf{x}=\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2)$的解是等价的，所以原方程组的解可通过计算$\mathbf{x}=\mathbf{Q}\hat{\mathbf{x}}$来获得。可以看出协议的设计是正确的。

2) 安全性分析

基于同态加密体制的安全性，当 Bob 接收到 Alice 的加密数据$Enc(\mathbf{A}_1)$和$Enc(\mathbf{v}_1)$后，由于 Bob 不知道解密密钥，因此他无法得知$\mathbf{A}_1$和$\mathbf{v}_1$的具体值。而 Bob 采用随机可逆矩阵$\mathbf{J}$，$\mathbf{Q}$来伪装方程矩阵，对于 Alice 来说，她只能获得$\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q}$和$\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2)$的值，而对于$\mathbf{JA}_1\mathbf{Q}$，$\mathbf{JA}_2\mathbf{Q}$，$\mathbf{Jv}_1$，$\mathbf{Jv}_2$她全然不知，更无法推导出$\mathbf{J}$，$\mathbf{Q}$，$\mathbf{A}_2$和$\mathbf{v}_2$的值。

3) 复杂度分析

关于计算代价，本协议中 Alice 和 Bob 分别调用了$n(n+1)$次加密算法，Bob 在步骤 2 中计算$Enc(\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q})$时，通过$n^2$次同态加法得到$Enc(\mathbf{A}_1+\mathbf{A}_2)$，而后通过$2n^3$次同态数乘和$2n^2(n-1)$次同态加法得到$Enc(\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2)\mathbf{Q})$。当计算$Enc(\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2))$时，Bob 先计算$n$次同态加法，再通过$n^2$次同态数乘和$n(n-1)$次同态加法得到$Enc(\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2))$。所以 Bob 共需要调用$2n^3$次同态加法和$n^2(2n+1)$次同态数乘，Alice 需要调用$n(n+1)$次解密算法。协议共经过 4 轮通信，轮复杂度为$O(1)$，通信复杂度为$O(n^2)$。

3 分布式线性方程组模型的安全多方计算协议

3.1 问题描述

多方安全线性方程组问题：P_1有一个矩阵$\mathbf{A}_1$和一个向量$\mathbf{v}_1$，…，P_S有一个矩阵$\mathbf{A}_s$和一个向量$\mathbf{v}_s$；$\mathbf{A}_1,\mathbf{A}_2,\cdots,\mathbf{A}_s$是$n\times n$维矩阵，$\mathbf{v}_1,\mathbf{v}_2,\cdots,\mathbf{v}_s$是$n$维向量。在不泄露它们各自的保密输入的情况下，$P_1,P_2,\cdots,P_s$需要共同求解线性方程组

$$(\mathbf{A}_1+\mathbf{A}_2+\cdots+\mathbf{A}_s)\mathbf{x}=\mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_s \quad \cdots\cdots(1)$$

3.2 安全解决的设计思路

多方安全计算线性方程组要解决的一个共同问题也是数据伪装，首先必须找到能伪装式(1)，且与式(1)同解的矩阵方程。我们采用$\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2+\cdots+\mathbf{A}_s)\hat{\mathbf{x}}=\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_s)$来伪装。由于$\mathbf{x}=\hat{\mathbf{x}}$，那么类似于两方情况，我们使其中的一方(如$P_S$)来单独求解原线性方程组的伪装方程组$(\mathbf{A}_1+\mathbf{A}_2+\cdots+\mathbf{A}_s)\mathbf{x}=\mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_s$。

3.3 协议设计

输入：$P_i(1\leqslant i\leqslant s)$拥有$n\times n$维矩阵$\mathbf{A}_i$和$n$维向量$\mathbf{v}_i$。

输出：$P_i(1\leqslant i\leqslant s)$得到$x$，满足线性方程组。

1) $P_j(2\leqslant j\leqslant s)$用$P_S$的公钥加密矩阵$E_k(\mathbf{A}_j)$和$E_k(\mathbf{v}_j)$，并将它们发送给$P_1$。

2) P_1用P_S的公钥计算加密矩阵$E_k(\mathbf{A}_1)$和$E_k(\mathbf{v}_1)$。根据加密性质的同态性，P_1计算$\sum_{i=1}^{s}E_k(\mathbf{A}_i)=E_k(\sum_{i=1}^{s}\mathbf{A}_i)$和$\sum_{i=1}^{s}E_k(\mathbf{v}_i)=E_k(\sum_{i=1}^{s}\mathbf{v}_i)$，接着随机生成一个$n$阶可逆矩阵$\mathbf{J}$，计算$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{A}_i)$和$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{v}_i)$并将它们发送给$P_S$。

3) P_S利用解密密钥对接收到的$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{A}_i)$和$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{v}_i)$进行解密。$P_S$解密可获得$\mathbf{J}\sum_{i=1}^{s}\mathbf{A}_i$和$\mathbf{J}\sum_{i=1}^{s}\mathbf{v}_i$的值。$P_S$求解线性方程组$\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2+\cdots+\mathbf{A}_s)\hat{\mathbf{x}}=\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_s)$。

4) P_S将$\hat{\mathbf{x}}$发送给其余参与方$P_i(1\leqslant i\leqslant s\text{-}1)$。

3.4 协议分析

1) 正确性分析

根据加密体制的同态性，$\sum_{i=1}^{s}E_k(\mathbf{A}_i)=E_k(\sum_{i=1}^{s}\mathbf{A}_i)$和$\sum_{i=1}^{s}E_k(\mathbf{v}_i)=E_k(\sum_{i=1}^{s}\mathbf{v}_i)$是成立的，并且$P_1$可计算获得$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{A}_i)$和$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{v}_i)$的值。由于$P_S$拥有相应的私钥，可以对$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{A}_i)$和$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{v}_i)$进行解密并获得$\mathbf{J}\sum_{i=1}^{s}\mathbf{A}_i$和$\mathbf{J}\sum_{i=1}^{s}\mathbf{v}_i$，所以$P_S$可以单独求解线性方程组$\mathbf{J}(\mathbf{A}_1+\mathbf{A}_2+\cdots+\mathbf{A}_s)\hat{\mathbf{x}}=\mathbf{J}(\mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_s)$。

由于$\mathbf{J}$为n阶可逆矩阵，所以$\mathbf{x}=\hat{\mathbf{x}}$即为线性方程组$(\mathbf{A}_1+\mathbf{A}_2+\cdots+\mathbf{A}_s)\mathbf{x}=\mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_s$的解。

2) 安全性分析

因为所有参与方都是半诚实的，且所使用的加密体制是安全的，因而当攻击者的支撑结构为$S(A)=\{\{2,\ldots,s\},\{1,\ldots,s-1\}\}$时，协议是安全的。$P_j(1\leqslant j\leqslant s)$采用$P_S$的公钥加密矩阵$E_k(\mathbf{A}_j)$和$E_k(\mathbf{v}_j)$，并发送给$P_1$。$P_1$没有解密密钥，故而$P_1$无法获得$P_j(1\leqslant j\leqslant s)$的保密数据。$P_S$解密获得$\mathbf{J}\sum_{i=1}^{s}\mathbf{A}_i$和$\mathbf{J}\sum_{i=1}^{s}\mathbf{v}_i$的值，因为$\mathbf{J}$是随机的，所以$P_S$也无法获得$P_j(1\leqslant j\leqslant s\text{-}1)$的保密数据。

记$\{P_2,\ldots,P_s\}$和$\{P_1,\ldots,P_{s-1}\}$分别为被攻击者腐败的成员，令$I_1=\{2,\ldots,s\}$，$I_2=\{1,\ldots,s-1\}$。假定Ω为攻击者，进一步分析可以得到：

(1) 当$\Omega_1=\{P_i:i\in I_1\}$，显然它不能攻击$P_1$，因为得不到$\mathbf{A}_1$和$\mathbf{v}_1$的相关信息。

协议结束后，Ω_1得到$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{A}_i)$和$E_k(\mathbf{J}\sum_{i=1}^{s}\mathbf{v}_i)$，并由此得到$\mathbf{J}\sum_{i=1}^{s}\mathbf{A}_i$和$\mathbf{J}\sum_{i=1}^{s}\mathbf{v}_i$。其中$\Omega_1$只知道$\mathbf{A}_i,\mathbf{v}_i(2\leqslant i\leqslant s)$，无法获得$J,\mathbf{A}_1,\mathbf{v}_1$的信息。

(2) 当$\Omega_2=\{P_i:i\in I_2\}$，它也不能攻击$P_S$，因为得不到$\mathbf{A}_S$和$\mathbf{v}_S$的相关信息。

P_S在协议的整个执行过程中，只在最后一步向其余参与方发送最终计算结果$\hat{x}$，Ω_2无法从$\hat{x}$中得到$\mathbf{A}_S$和$\mathbf{v}_S$的信息。

协议存在一定缺陷：由于P_S拥有解密密钥，P_1拥有其他参与方的$E_k(\mathbf{A}_j)$和$E_k(\mathbf{v}_j)$$(2\leqslant j\leqslant s)$，一旦$P_1$和$P_S$合谋，则其他参与方的秘密输入都将完全泄露。

3）复杂度分析

与文献[3]中利用的协议相比，本文提出的协议的轮复杂度显著降低了。另外，在通信复杂度和计算复杂度方面也有所降低。关于计算代价，$P_i(1 \leqslant i \leqslant s)$分别调用$(n+1)n$加密算法。在步骤2中，$P_1$计算$\sum_{i=1}^{s} E_k(M_i)=E_k(\sum_{i=1}^{s} M_i)$和$\sum_{i=1}^{s} E_k(b_i)=E_k(\sum_{i=1}^{s} b_i)$时，需调用$(s-1)(n^2+n)$次同态加法；在计算$E_k(J\sum_{i=1}^{s} M_i)$时，需调用$n^2(n-1)$次同态加法和$n^3$次同态数乘；在计算$E_k(J\sum_{i=1}^{s} b_i)$时，需调用$n(n-1)$次同态加法和$n^2$次同态数乘。故而$P_1$总共需要调用$(s+n-2)(n^2+n)$次同态加法和$n^2(n-1)$次同态数乘。$P_S$在步骤4中需要调用$n^2+n$次解密算法。协议的通信复杂度为$O(sn^2)$。

4 结束语

本文在半诚实模型下基于分布式线性方程组求解模型，参照同态加密体制提出了两方安全计算协议和多方安全计算协议，并对所提出的协议进行了正确性、安全性和复杂度分析。与已提出的基于茫然传输协议设计的安全计算协议相比，明显降低了协议复杂度，提高了计算效果。下一步的工作将继续探讨：1）设计更多的解决其他科学计算问题的多方安全计算协议；2）将半诚实模型推广到恶意模型，即研究恶意模型的多方安全计算。（责编 马珂）

参考文献

[1]Yao A. C. Protocols for secure computation[C]. Proceedings of the 23rd IEEE Symposium on Foundations of Computer Science, 1982: 160-164.

[2]Wenliang Du. Privacy-preserving cooperative scientific computations[C]. Proceedings of the computer security of foundations workshop,2001: 273-282.

[3] 罗文俊，李祥. 多方安全矩阵乘积协议及应用[J]. 计算机学报，2005, 28(7): 1230-1235.

[4]K. Nissim and E. Weinreb. Communication efficient secure linear algebra[C]. In the Third Theory of Cryptography Conference, 2006: 522-541.

[5]E. Kiltz, P. Mohassel, E. Weinreb et al. Secure linear algebra using linearly recurrent sequences[C]. In the 4th Theory of Cryptography Conference, Amsterdam, The Netherlands, 2007: 291-310.

[6]P. Pallier. Public-key cryptosystems based on composite degree residuosity classes[C]. In Advances in Cryptology-EUROCRYPT, 1999: 223-238.

资讯

第10届（2013）信息安全与对抗技术竞赛圆满落幕

2013年5月3日，《第10届（2013）信息安全与对抗技术竞赛》在众多信息安全爱好者的期待中拉开了序幕，本届竞赛共分为两个阶段，即个人挑战赛和分组对抗赛。

个人挑战赛从5月1日至6月30日，历时61天。个人挑战赛分为答题、基础、脚本、破解、溢出、内核、真实7大关卡，关卡考察内容涉及WEB知识、ASP/PHP脚本、缓冲区溢出、软件脱壳破解、系统漏洞利用、社会工程学等信息安全知识，知识点共计160多项。据不完全统计，本届竞赛个人挑战赛注册人数3943多人，参与院校400多所，同时吸引了不少非在校选手（不在评奖范围内）参加，这为竞赛更增添了挑战与机遇！

分组对抗赛于2013年8月14日至15日历时2天，分为“北理工-绿盟科技”平台和“北理工-神州数码”2个竞赛平台。基于“北理工-绿盟科技”平台的竞赛是从个人挑战赛中选择并邀请全国各地共计20名优秀在校学生齐聚北京理工大学，分为4个小组，在封闭的真实复杂网络环境中展开攻防角逐。基于“北理工-神州数码”平台的竞赛采用，邀请了河南和北京的多所院校共计20名学生参加，分为4个小组进行单兵作战、分组夺旗和分组对抗3种模式的竞赛。竞赛充分展示了各位选手的个人水平和小组的协同合作能力。

北京理工大学“信息安全与对抗技术竞赛”（Information Security and Countermeasures Contest，ISCC）是由北京理工大学教务处和团委主办、信息与电子学院信息系统及安全对抗实验中心承办的学科知识和专业技术竞赛，重点考察学生的信息网络攻防知识与技能，旨在提升信息安全意识、普及信息安全知识、实践信息安全技术、共创信息安全环境、发现信息安全人才，同时探索信息对抗技术及其相关专业工程教育的新途径。自2004年首届竞赛成功举办以来，ISCC经过多年的发展，竞赛平台日渐完善、知识范围不断拓展、攻防方式日益丰富。现在，每年一度的ISCC已经成为全面考核参赛选手信息安全与对抗技术水平的综合性平台，更为全国各地信息安全人才提供思想、知识、技术交流的大好机会。（记者 程斌）