软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, 2022,33(12):4771–4783 [doi: 10.13328/j.cnki.jos.006361]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

抗主动攻击的保密比较协议*

李顺东[1], 王文丽[1], 陈明艳[2], 汪榆淋[2]

[1](陕西师范大学 计算机科学学院, 陕西 西安 710119)

[2](陕西师范大学 数学与统计学院, 陕西 西安 710119)

通信作者: 李顺东, E-mail: shundong@snnu.edu.cn

摘 要: 互联网、物联网和大数据的迅速发展, 为数据共享带来了无限的机遇, 也给私有数据的隐私保护带来了严峻的挑战. 安全多方计算是数据共享中隐私保护的关键技术, 是密码学的一个重要研究方向, 也是国际密码学界研究的热点. 保密比较两个数的大小是安全多方计算的一个基本问题, 是构建其他隐私保护协议的一个基本模块. 当比较的数较小时, 还没有可靠的能够抵抗主动攻击的保密比较问题解决方案. 很多应用场景中的参与者可能会发动主动攻击, 因为尚没有抗主动攻击的保密比较协议, 这些场景中的保密比较问题还无法解决. 因而研究抗主动攻击的保密比较问题解决方案有重要理论与实际意义. 提出了一种加密-选择安全多方计算模式和编码+保密洗牌证明的抵抗主动攻击方法. 在此基础上, 设计了半诚实模型下安全的保密比较协议, 用模拟范例证明了协议的安全性; 分析了恶意参与者可能实施的主动攻击, 结合 ElGamal 密码系统的乘法同态性、离散对数与保密洗牌的零知识证明设计阻止恶意行为的措施, 将半诚实模型下安全的保密比较协议改造成抗主动攻击的保密比较协议, 并用理想-实际范例证明了协议的安全性. 最后分析了协议的效率, 并通过实验验证协议是可行的.

关键词: 安全多方计算; 保密比较; 主动攻击; 模拟范例; 理想-实际范例; 保密洗牌; 零知识证明

中图法分类号: TP306

中文引用格式: 李顺东, 王文丽, 陈明艳, 汪榆淋. 抗主动攻击的保密比较协议. 软件学报, 2022, 33(12): 4771–4783. http://www. jos.org.cn/1000-9825/6361.htm

英文引用格式: Li SD, Wang WL, Chen MY, Wang YL. Comparing Protocol Against Active Attacks. Ruan Jian Xue Bao/Journal of Software, 2022, 33(12): 4771–4783 (in Chinese). http://www.jos.org.cn/1000-9825/6361.htm

Comparing Protocol Against Active Attacks

LI Shun-Dong[1], WANG Wen-Li[1], CHEN Ming-Yan[2], WANG Yu-Lin[2]

[1](School of Computer Science, Shaanxi Normal University, Xi'an 710119, China)

[2](School of Mathematics and Statistics, Shaanxi Normal University, Xi'an 710119, China)

Abstract: The rapid development of the Internet, IOT, and big data brings great chance to share data owned by different entities, but it also brings severe challenge to privacy-preserving of private data. Secure multiparty computation is a key privacy-preserving technology, an important field of cryptography, and a focus of international cryptographic community. Privately comparing two numbers is a basic problem of secure multiparty computation. The protocols for this problem are building blocks to construct other privacy-preserving protocols. If the two numbers to be compared is small, there is no reliable solution to this problem that can resist active attacks. In many scenarios, the participants may be malicious and they may actively attack a protocol. If this is the case, there is no solution that can be used to privately compare the numbers. Therefore, it is of important theoretical and practical significance to design a protocol that can resist active attacks. This study first proposes a new technique called encrypt-and-choose and a new technology to resist active attacks: encoding+secure shuffle. Based on these techniques, a secure comparison protocol is first designed that is secure in the semi-honest model. Its security is proved by using the simulation paradigm. All possible active attacks are analyzed that the protocol may suffer from, and

* 基金项目: 国家自然科学基金(61272435)
收稿时间: 2020-11-18; 修改时间: 2021-03-02; 采用时间: 2021-04-26

ElGamal multiplicative homomorphism and zero-knowledge proof of discrete logarithm and secure shuffle are used to resist possible active attacks. The protocol is then converted to one that can resist active attacks, and it is proved that it is secure against active attacks by using the ideal-real paradigm. Finally, the efficiency of the protocol is analyzed and tested. The experimental results demonstrate that the protocol is practical.

Key words: secure multiparty computation; secure comparison; active attack; simulation paradigm; ideal-real paradigm; secure shuffle; zero-knowledge proof

在数据时代, 人们利用数据发现事物之间的关系和发展规律、对事物进行分类聚类, 利用数据进行决策并采取相应的行动. 但由于各方面的限制, 任何机构都无法获得自己所需要的全部数据, 因而不得不利用其他机构的数据, 这就需要不同机构之间进行数据共享. 共享数据联合计算能够形成双赢的局面, 但共享的数据可能包含大量的隐私信息, 如果不加保护地共享, 就会泄露隐私信息, 导致严重的后果. 隐私保护和数据共享之间的矛盾日益凸显. 如何在充分保护数据隐私的前提下实现隐私数据共享, 最大程度地发挥数据的效益, 这既是密码学一个重要的基础问题, 也是重大的信息安全实践问题. 安全多方计算(secure multiparty computation, SMC)是实现隐私数据共享的核心技术.

SMC 源于 1982 年姚期智教授[1]提出的百万富翁问题. 图灵奖获得者 Goldwasser[2]与著名密码学家 Cramer[3]认为: SMC 是密码学中极其重要的、强有力的工具, 将发展成为计算科学中不可或缺的组成部分. SMC 的广泛应用前景和对于保护隐私的数据共享的巨大需求, 使得 SMC 受到密码学界的高度重视, 并发展成为解决数据共享与外包计算中隐私保护问题的关键技术[4]. SMC 是密码学中一个最重要、最活跃的研究领域[5], 也是国际密码学界的研究热点之一, 是美密会、欧密会、亚密会和计算机与通信安全国际会议(CCS)上论文数量最多或次多的研究方向(参看最近 3 年的美密会、欧密会及 CCS 会议的论文集, 2020 年欧密会的 SMC 专题有论文 8 篇, 美密会的 SMC 专题有论文 14 篇, 都是论文最多的).

姚期智教授提出百万富翁问题之后, 又提出了基于 garbled circuit 和不经意传输的 SMC 问题的通用解决方案[6]. Goldreich 等人也提出了基于算术电路和秘密共享的通用解决方案[7]. 在通用解决方案之外, 人们针对各种具体的 SMC 问题提出了具体的解决方案, 解决了许多数据共享联合计算中的隐私保护问题, 但目前的解决方案大多数是半诚实模型下安全的. 半诚实模型假设参与者会忠实地执行协议, 但由于好奇心的驱使, 他们会记录下协议执行过程中所收到的所有信息, 并在协议执行后试图利用记录的信息推算其他参与者的隐私信息. 这是一种比较低的安全保证, 只能抵抗被动攻击和防止由于粗心大意导致的隐私泄露. 而对于很多重要的隐私信息, 攻击者往往不是出于好奇, 而是受利益或者其他恶意目的驱使, 他们有更强烈的主动攻击意识和欲望, 愿意付出更大的成本实施攻击以获得相应的隐私数据. 在这样的应用场景中, 现在的多数方案都不能保证相应隐私数据的隐私性. 因此未来若干年内, 研究存在主动攻击时安全的 SMC 解决方案, 将是 SMC 研究的重点.

保密比较两个隐私数据的大小问题是 SMC 的一个基本问题[8−11], 解决方案是构造其他隐私保护协议的基础模块. 存在主动攻击时的保密比较问题解决方案对于设计抗主动攻击的隐私保护协议至关重要, 具有重要的理论与实际意义, 但目前这方面的研究很少(这方面的研究见第 7 节的相关工作). 本文研究隐私数据范围已知而且范围不太大的应用场景中抗主动攻击的比较问题的解决方案, 本文贡献如下:

(1) 提出了一种新的编码方法和加密-选择(encrypt-choose)方法, 可以广泛应用于解决保密数据范围较小场景中的隐私保护问题. 这种编码方法可以抵抗一定程度的主动攻击, 本文利用这个方法构造了简单的半诚实模型下的保密比较解决方案, 并用模拟范例证明了方案在半诚实模型下是安全的;

(2) 在分析半诚实模型下的解决方案可能遭受的主动攻击基础上, 充分利用 ElGamal 密码系统的乘法同态性、离散对数和保密洗牌(保密置换)的零知识证明发现主动攻击, 从而迫使协议的参与者只能实施被动攻击, 阻止其主动攻击, 将半诚实模型下的解决方案改造成抗主动攻击的解决方案;

(3) 找到了一种新的抵抗主动攻击的方法: 编码+保密洗牌方法. 用理想-真实范例严格证明了所设计的协议能够抵抗主动攻击, 并对协议进行了安全性、计算复杂性、通信复杂性分析和实验测试. 实验

结果表明, 协议是可行的.

本文第 1 节给出了系统结构和安全模型. 第 2 节介绍了构造方案所需要的预备知识. 第 3 节介绍了加密-选择的原理, 并用加密-选择构造了半诚实模型下的保密比较协议. 第 4 节将半诚实模型下安全的协议改造成抗主动攻击的协议. 第 5 节对协议进行了性能分析, 证明了协议的安全性并测试了协议的效率. 第 6 节是与本文工作有关的工作. 第 7 节是本文的结论.

1 系统结构和安全模型

1.1 系统结构

安全双方计算问题的系统结构如图 1 所示: Alice 拥有隐私数据 x, Bob 拥有隐私数据 y. 他们都不愿意泄露自己的隐私数据, 但都想得到关于 x, y 的函数值 $F(x,y)$, 其中的 x, y 可以是单个数据, 也可以是数据集合; 在不同的问题中, $F(x,y)$可以有不同的定义, 保密比较问题中的 $F(x,y)$定义如下:

$$F(x,y)=\begin{cases}0, & \text{如果} x>y\\ 1, & \text{否则}\end{cases} \tag{1}$$

图 1 系统结构

1.2 安全模型

一个密码协议是否安全完全取决于其应用环境, 我们的协议的应用环境是: (1) 协议的参与者之间没有安全信道; (2) 参与者的计算能力是有限的; (3) 参与者的行为是一致的, 也就是说, 一个参与者在协议执行过程中或者一直是恶意的, 或者一直是半诚实的, 不会在中间发生变化. 在这样的环境中, 根据参与者行为的不同, 有两种安全模型: 半诚实模型和恶意模型.

- 半诚实模型[7], 又称为诚实但好奇模型. 半诚实模型假设参与者会诚实地按照协议的要求严格执行协议, 但由于好奇心的驱使, 他可能记录协议执行过程中收到的所有信息, 并在协议执行之后尝试利用记录的信息推断其他参与者的隐私数据信息. 半诚实参与者对协议的攻击只发生在协议执行之后;
- 恶意模型[7]. 半诚实模型下, 安全的协议可以防止由于自己的粗心大意或者对方的好奇导致的隐私泄露, 但如果信息的价值足够高, 参与者可能有更强的意愿对协议进行攻击. 在这种情况下, 半诚实模型下, 安全的协议不能保证数据的隐私, 必须研究更加安全的协议. 恶意模型认为: 参与者对协议的攻击不仅仅是受好奇心的驱使, 更可能受到利益或其他外力的驱使, 因此攻击者愿意付出更大的成本对协议主动发起攻击.

在恶意模型中, 参与者可以对协议进行任意的攻击, 可能: (1) 拒绝参与协议; (2) 中途退出协议; (3)提供虚假的输入信息; (4) 其他恶意行为. 因为只要参与者有人身自由就可以拒绝参与协议也可以中途退出协议, 所以任何协议都不能阻止行为(1)和行为(2); 因为输入信息是参与者的隐私信息, 要保证隐私就不能泄露真实的输入, 因此也无法验证参与者的输入是真实的还是伪造的, 所以行为(3)也无法阻止, 即使采用理想协议也无法阻止. 我们需要做的是, 参与者在协议执行过程中不能改变其输入值. 因此在恶意模型中, 我们通常不考虑恶意行为(1)~行为(3), 除此之外的恶意行为都要阻止.

恶意参与者还能有什么其他恶意行为呢? 我们知道, 执行协议中, 所有参与者的行为有 3 种: 发送消息、接收消息、自己计算(包括验证消息的计算), 其攻击行为只能隐藏在发送的消息中. 实际上, 只要输入给定、计算中需要的随机数给定, 计算过程的每一步得到什么消息也是确定的. 我们要做的就是要保证每个参与者

每一步发送的消息都是根据所有参与者提供的初始输入和选择的随机数唯一确定的信息, 这样, 恶意参与者就无法实施攻击. 抗主动攻击协议中采取的各种措施, 如零知识证明、认证计算、cut-and-choose、对偶执行等, 都是要保证这一点. 此外, cut-and-choose 涉及协议的多次执行, 每次执行都需要输入, 因此还必须采取措施保证各个参与者每次都提供相同的输入.

要证明一个协议是安全的, 必须证明其符合安全性的定义. 协议的安全性定义分为半诚实模型下的安全性定义和恶意模型下的安全性定义. 广泛接受的安全多方计算协议的安全性定义是 Goldreich 在文献[7]中给出的, 其中, 半诚实模型下的安全性定义的基础是模拟范例, 而恶意模型下的安全性定义的基础是理想-实际范例. 两种情况下的安全性定义都需要先定义一种理想协议模型. 半诚实模型下的安全性定义本质上是说, 如果实际协议不比理想协议泄露更多信息, 实际协议就是安全的; 恶意模型的安全性定义本质上说, 如果实际协议中能够实施可行的主动攻击在理想协议中也能够实施, 实际协议就是安全的(其逆否命题就是在理想协议中无法实施的攻击在实际协议中也无法实施, 因而实际协议是安全的). 关于两种模型下的安全性定义的详细情况请见文献[7].

2 预备知识

本文解决比较问题的主要工具是 ElGamal 密码系统, 主要利用加密算法的乘法同态性. 构造协议主要利用加密选择、重随机化技巧. 迫使参与者诚实需要零知识证明, 证明协议安全需要根据安全性的定义进行. 本节对这些知识进行简单介绍.

2.1 ElGamal密码系统

ElGamal 密码系统是具有乘法同态性公钥加密系统[12], 是一种概率密码系统, 其密钥生成算法、加密算法和解密算法具体描述如下:

- 密钥生成算法 Gen(τ). 给定安全参数τ, Gen(τ)生成一个τ比特的大素数 p 以及 Z_p^*的一个生成元 g, 随机选取 $sk\in Z_p^*$作为私钥, 计算并公布 $h=g^{sk} \bmod p$ 作为对应的公钥;
- 加密算法 Enc(m,r). 为加密消息 $m(m\in Z_p^*)$, 选择一个随机数 r, 计算:

$$C=E(m,r)=(c_1,c_2)=(g^r \bmod p, mh^r \bmod p);$$

- 解密 Dec(C,sk). 对于密文 $C=(c_1,c_2)$, 解密为:

$$m=D(C)=c_2\cdot c_1^{-sk} \bmod p;$$

- 同态性质. ElGamal 密码系统具有乘法同态性:

$$E(m_1,r_1)\times E(m_2,r_2)=(g^{r_1},m_1h^{r_1})\times(g^{r_2},m_2h^{r_2})=(g^{r_1+r_2},m_1m_2h^{r_1+r_2})=E(m_1m_2,r_1+r_2).$$

2.2 重随机化

假设 $c=E(m,r_1)$是 ElGamal 加密算法加密的密文, 根据 ElGamal 加密算法的乘法同态性有:

$$c'=c\cdot E(1,r_2)=E(m,r_1)\times E(1,r_2)=(g^{r_1}\times g^{r_2} \bmod p, mh^{r_1}h^{r_2} \bmod p)=(g^{r_1+r_2} \bmod p, mh^{r_1+r_2} \bmod p)=E(m,r_1+r_2).$$

c, c'看上去好像两个没有任何关系的随机数, 但是 c, c'都被解密成 m(本质上等价于加密 m 时选择了不同的随机数而得到的不同密文). 由 c 到 c'的变换, 称为密文的重随机化或者密文的刷新.

2.3 离散对数的零知识证明

文献[13]提出了一种证明离散对数相等的思想, 文献[14]对其进行了修改. 在基于离散对数的密码系统中, 如果持有解密密钥的一方向另一方证明解密过程使用了正确的私钥而没有进行欺骗, 常常用这种方法. 要求 Alice 向 Bob 零知识地证明没有欺骗, 也是迫使 Alice 诚实的一种手段.

假设 G 是一个循环群, 其阶数为 m 但 m 未知, g 是 G 的一个生成元, $h\in G$, 并且 Alice 知道$\alpha=g^x$, $\beta=h^x$. Alice 要向 Bob 证明 $\log_g\alpha=\log_h\beta$但不泄露 x, 就可以采用这种方法.

具体方法及其正确性证明可见文献[13]. 证明基于这样的事实: 如果 Alice 不知道 x, 就无法从 g^e 计算 g^{ex},

也无法计算 y, 使得 $g^y/\alpha^e=X$, $h^y/\beta^e=Y$. 利用这个事实的逆否命题证明知道 x.

2.4 洗牌零知识证明

文献[15]提出来一种零知识证明方法可以解决这样的问题: Alice 可以生成一个向量 $V=(v_1,\ldots,v_m)$并公开加密为一个密文向量 $E(V)=(E(V_1),\ldots,E(V_m))$. Bob 可以对该密文向量进行保密随机置换+重随机化即洗牌得到 $E(V')$, 并能够用零知识证明的方法向 Alice 证明其 $E(V')$只是 $E(V)$的随机置换+重随机化, 除此之外, 没有进行任何其他操作(即没有更改 V 中的明文). 具体证明过程请见文献[15].

3 半诚实模型下的协议

3.1 加密选择

本节我们提出一种新的 SMC 模式: 加密-选择. 这个方法可以解决许多问题, 而且使问题的解决变得非常简单易行, 尤其是利用这种方法设计的协议很容易改造成恶意模型下的协议. 这种模式一般适合于消息空间较小的应用场景. 加密-选择的原理如下:

假设 Alice 和 Bob 分别有一个保密数据 x, y, 其中, $a_1<x,y<a_m$(即 a_1, a_m 是 x, y 可能范围的下界与上界). 他们要比较 x 和 y 的大小, 即要知道 x 是不是小于 y, 但不能泄露 x, y 的值. 这就是著名的百万富翁问题, 也就是保密比较两个数大小的问题. 这里我们提出一种新的方法, 很容易用这种方法构造半诚实模型下的保密比较协议, 而且也很容易将半诚实模型下的协议改造成恶意模型下的协议.

更直观地假设 $a_1=1$, $a_m=m$, $1<x,y<m$. Alice 首先构造一个 m 维的 2-3 向量:

$$V=(v_1,\ldots,v_m) \tag{2}$$

其中, $v_i=2(i<x)$, $v_i=3(i\geqslant x)$. 假设 $m=10$, $x=8$, $y=5$, Alice 构造的向量如下:

$$V=(2,2,2,2,2,2,2,3,3,3) \tag{3}$$

把向量 V 发送给 Bob.

Bob 选择 v_v, 显然, 如果 $v_v=2$, 那么 $x>y$; 否则, $x\leqslant y$. 在本例中, Bob 将计算 $v_5=2$, 表明 $x>y$.

这个方法的原理显而易见, 但直接这样当然并不安全. 要实现保密比较, 只需要在密文上执行这个过程即可. 因此有下面的协议.

注: 因为这种编码方法将一个隐私数据编码为一个向量, 向量的维数等于集合的势 m, 一方面, 如果 m 未知就无法编码为 m 维的向量; 另一方面, 当数据范围很大, 即 m 很大时, 计算量很大, 协议将变得不实用.

3.2 半诚实模型下的保密比较协议

根据上述加密-选择原理, 我们设计半诚实模型下的保密比较协议如下.

协议 1. 半诚实模型下的保密比较协议.

输入: Alice 和 Bob 分别输入 x 和 y;

输出: $F(x,y)$, 以 $x,y\in\{1,\ldots,10\}$, $x=8$, $y=5$ 为例.

准备: Alice 运行 ElGamal 密码系统的密钥生成算法, 生成大素数 p、Z_p^* 的生成元 g、Alice 的私钥 sk 和公钥 $h=g^{sk} \bmod p$, 并公布(g,p,h).

1. Alice 构造向量 V 如下:

$$V=(2,2,2,2,2,2,2,3,3,3) \tag{4}$$

2. Alice 用 h 加密向量 V 得到向量的密文:

$$E(V)=(E(2),E(2),E(2),E(2),E(2),E(2),E(2),E(3),E(3),E(3))$$

并发送给 Bob, 其中, $E(m)=E(m,r)=(g^r \bmod p, mh^r \bmod p)$;

3. Bob 选择 $E(v_y)=E(v_5)$并重随机化后发送给 Alice;

4. Alice 解密得到 $v_y=2$, 输出 $F(x,y)=0$.

这个协议的正确性是显然的, 无需证明. 关于协议的安全性, 我们有下面的定理.

定理 1. 协议 1 对被动攻击是安全的, 并能够抵抗选择方的恶意攻击.

证明: 这个定理证明比较简单, 构造满足文献[7]所定义的模拟器 S_1, S_2 即可. 因为参与者都是半诚实的, 所以这样的模拟器是比较容易构造的. 我们集中精力证明恶意模型下的协议的安全性, 本定理的证明省略.

4 恶意模型下的协议

协议 1 在半诚实模型下是安全的, 但是在恶意模型下不安全. 我们分析恶意参与者会如何攻击这个协议, 在此基础上设计阻止或发现这些攻击的方案, 使得恶意敌手的主动攻击或者无法实施或者被发现, 从而迫使恶意参与者以半诚实的方式参与协议. 最终, 将这个协议改造抗主动攻击的保密比较协议. 首先分析针对协议 1, Alice 和 Bob 分别能采取什么恶意行为.

4.1 双方可能的恶意行为

在协议 1 中, Alice 和 Bob 可以实施下述恶意行为, 我们无法阻止也无法发现:

- Alice 在第 2 步加密的向量不是协议规定的向量, 比如她可以加密向量:

$$V_x=(1,2,3,4,5,6,7,8,9,10) \tag{5}$$

  如果这样, 当 Bob 选择第 5 个分量的密文并经过重随机化后发给她, 解密之后得到 5, 她就完全知道 Bob 的 y=5, 但她仍然告诉 Bob 说解密的结果为 3. 这样做不但得到了 Bob 数据的全部信息, 还让 Bob 得到一个错误的结论. 我们必须能够阻止或者发现这种攻击行为, 即必须保证她加密的向量分量只能是 2 和 3, 否则将会被发现;
- Alice 在第 2 步加密的确实只有 2 和 3, 但是向量的形式却不符合公式(4)的形式(每个向量只能是连续若干个 2, 然后是连续若干个 3 的形式). 例如, 她加密的 2-3 向量的形式可能是:

$$V=(2,3,2,3,2,3,2,3,2,3) \tag{6}$$

  或者

$$V=(2,2,2,3,3,3,3,3,2,2) \tag{7}$$

  如果加密形如公式(6)的向量, 她可以知道 Bob 的数是奇数还是偶数; 如果加密形如公式(7)的向量, 她可以根据解密的结果知道 Bob 的 y 的范围. 因此, 必须能够阻止或者发现这样的恶意行为;
- 无论加密形如公式(5)、公式(6)或公式(7)的向量, Alice 都可以在第 4 步告诉 Bob 一个错误的结果. 比如, 她已经知道 y=5, 但她告诉 Bob 解密的结果为 3, 从而使 Bob 得到了错误的结论, 即认为 $x\leqslant y$. 因此, 必须能够阻止或者发现这种恶意行为;
- 当然, 也可能 Alice 加密的向量完全符合公式(4)的形式, 但与自己的 x 不符合. 这种情况可以归约为替换自己的输入, 这是在理想模型中也是无法避免的, 在实际抗主动攻击协议设计中不考虑如何阻止或发现这种恶意行为.

与此对应, 假设 Alice 是诚实的, 但 Bob 是恶意的, 他能够采取什么恶意行为呢? 在这个协议中, Bob 只在第 3 步进行了选择和重随机化, 也只有在这一步可以采取恶意行为. 恶意行为也只能针对选择与重随机化进行. 他可以不选择第 y 个密文, 这可以归约到 Bob 提供虚假输入的情况, 不予考虑. 他可以选择两个密文相乘, 解密的结果将不再是一个素数, 而是一个合数, 因而将会被发现. 所以, Bob 无法实施任何恶意行为.

注: Bob 还可能有这样的恶意行为: 他在选择阶段选择了 $E(v_i)$, $E(v_{i+1})$, $E(v_{i+2})$, 并计算 $E(v_{i+2})E(v_{i+1})E(v_i)^{-1}=E(v_{i+2}\cdot v_{i+1}/v_i)$. 如果这么做而 $v_i\neq v_{i+1}$, Alice 解密时就能够发现这种恶意行为; 如果 $v_i=v_{i+1}$, 解密的结果就是 v_{i+2}, 这等价于 Bob 用 i+2 替换了自己的输入 y, 即提供虚假输入的情况, 因而不予考虑.

4.2 迫使Alice诚实

下面我们针对 Alice 可以实施的主动攻击, 研究阻止或发现主动攻击的方法.

根据前面的分析, Alice 的恶意行为有两种: 加密不符合规定形式的向量或者在解密之后告诉 Bob 一个错

误的结果. 只要使 Alice 无法实施这两种恶意行为, 就可以保证协议在恶意模型下是安全的. 为了阻止或发现 Alice 在解密之后告诉 Bob 一个错误的结果, 可以用零知识证明迫使她只能告诉正确的结果, 否则无法通过零知识证明.

迫使 Alice 加密的向量符合规定的形式基于下面的事实: 如果 Alice 加密的向量符合公式(4)的形式, 那么每一个分量与前一个分量相除的结果即 v_{i+1}/v_i 只有一个不等于 1, 其他全是 1. 只要能够保密验证这一点即可. 而这一点可以利用 ElGamal 密码系统的乘法同态性和洗牌的零知识证明实现. 具体协议如下:

协议 2. 抗主动攻击的保密比较协议

输入: Alice 和 Bob 分别输入 x 和 y;

输出: $F(x,y)$.

准备: Alice 运行 ElGamal 密码系统的密钥生成算法, 生成大素数 p、Z_p^* 的生成元 g、Alice 的私钥 sk 和公钥 $h=g^{sk} \bmod p$, 公布公钥(g,p,h).

1. 假设 $x,y\in\{1,\ldots,m\}$. Alice 构造一个形如公式(4)的 m 维向量 $V=(v_1,\ldots,v_m)$, 加密后得到 $E(V)=(E(v_1),\ldots,E(v_m))$. 将 $E(V)$发送给 Bob;
2. Bob 计算 $C=(c_1,\ldots,c_{m-1})=(E(v_2)E(v_1)^{-1},\ldots,E(v_m)E(v_{m-1})^{-1})=(E(v_2/v_1),\ldots,E(v_m/v_{m-1}))$, 并将 C 发送给 Alice;
3. Alice 生成 C 的一个置换$\pi(C)$, 并利用零知识证明的方法证明$\pi(C)$只是 C 的一个置换. 如果不能证明这一点, 则说明 Alice 试图欺骗, 就中止协议; 如果证明通过, 则继续协议;
4. Alice 解密$\pi(C)$, 用零知识证明的方法证明解密正确. 如果不能证明, 就说明 Alice 在进行欺骗, 中止协议;
5. Bob 验证解密的结果只有一个是随机数, 其他都是 1. 如果符合这种形式, 就继续协议; 否则, 就说明 Alice 生成的 V 不符合规定的形式, 中止协议;
6. Bob 从 $E(V)$中选择 $E(v_y)$, 重新随机化后发给 Alice;
7. Alice 解密密文, 如果解密的结果是合数, 说明 Bob 在进行欺骗, 中止协议; 如果是素数, 继续协议;
8. Alice 向 Bob 证明解密正确, 根据解密结果做出判定;
9. 如果零知识证明被接受, Alice 和 Bob 各自输出 $F(x,y)$.

5 协议性能

我们从协议的安全性和计算复杂性方面来研究协议的性能.

5.1 安全性

假设 Alice 是诚实的, 因为 Bob 只是在 $E(V)$的基础上进行确定的计算, 如果 Bob 在第 2 步不遵守协议, 计算的 C 不是全部由 $E(v_{i+1}/v_i)$组成, Alice 就能够发现 Bob 的恶意行为.

与此对应, 如果 Bob 是诚实的, 因为 Alice 的每一个行为都需要通过零知识证明其行为是符合协议要求的, 任何不诚实的行为都会因为通不过证明而被发现. 协议要执行完毕, 双方都必须是诚实的. 协议的安全性是建立在加密算法安全性基础和零知识证明协议的安全性基础上的, 在此基础上有下面的结论:

定理 2. 协议 2 对恶意参与者是安全的.

证明: 假设在执行协议 2 时, Alice 和 Bob 采取的可接受的多项式时间算法策略对为 $\bar{A}=(A_1,A_2)$. 要证明协议能够抵抗主动攻击, 必须将策略对 $\bar{A}=(A_1,A_2)$ 转化为恶意模型下理想协议中相应的多项式时间算法策略对 $\bar{B}=(B_1,B_2)$, 使得 A_1, A_2 的输出分布与理想协议中 B_1, B_2 的输出分布计算不可区分. 根据安全性的定义, 不允许协议双方都不诚实, 所以对于 Alice 诚实与 Bob 诚实两种情况要分别处理.

首先假设 Alice 是诚实的, 那么策略 A_1 就是协议为她规定的策略, 协议中所有的验证与零知识证明过程都能够通过, A_1 最后输出协议规定的输出. 在这种情况下, 只要构造一个恶意模型下理想协议中的策略 B_2, 使得实际协议中 A_2 的输出与理想协议中 B_2 的输出计算不可区分即可.

如果 A_2 不诚实, 协议有可能在第 7 步中止. 如果协议在第 7 步中止, 那是因为 A_2 选择了欺骗(他不是选择一个密文而是选择两个或两个以上的密文相乘, 实际上他没有提供自己的数据的任何信息), 那么 A_1 并没有收到关于 y 和 $F(x,y)$的任何信息, 只能输出⊥. A_2 得到的信息只是 $E(V)$(因为密码系统的安全性, 他不能从 $E(V)$得到 x 的任何信息)和两次零知识证明的证据信息, 这些信息中也没有关于 x 和 $F(x,y)$的任何信息, 此时的 $\mathbf{REAL}_{\Pi,\bar{A}}(x,y)=\{\perp,\perp\}$. 在理想协议中, B_2 不给 TTP 发送数据即可. 这种情况下, TTP 将会给 B_1 发送一个⊥, B_1 通知 TTP 中止协议.

A_2 不诚实但协议执行完毕的情况. 因为策略 A_2 蕴含着主动攻击策略, 在实际执行协议时, 它可能不提供自己的真实数据 y, 而是根据 y 和策略 A_2 决定一个可能不同于 y 的数据 $A_2(y)$. A_2 与 A_1 执行协议, A_1 得到并输出 $F(x,A_2(y))$; A_2 得到 $F(x,A_2(y))$, 它将利用执行协议过程中所收到的 $F(x,A_2(y))$和其他信息(包括第 1 步收到的 $E(V)$, 第 3 步收到的零知识证明的证据α_1, 第 4 步收到的解密的$\pi(C)$即 $\mathbf{Dec}(\pi(C))$以及收到的零知识证明证据 α_2, 第 7 步收到的 $\mathbf{Dec}(E(v_v))$, 第 8 步收到的零知识证明证据α_3)决定自己的输出. 因此:

$$\mathbf{REAL}_{\Pi,\bar{A}}(x,y)=\{F(x,A_2(y)),A_2(E(V),\mathbf{Dec}(\pi(C)),\alpha_1,\alpha_2,\mathbf{Dec}(E(v_y)),\alpha_3,F(x,A_2(y)))\}.$$

在理想模型中, 只要找到一个可接受的策略对 $\bar{B}=(B_1,B_2)$, 使他们的输出与 $\mathbf{REAL}_{\Pi,\bar{A}}(x,y)$ 计算不可区分即可.

1. 因为 Alice 是诚实的, 理想协议中, B_1 采用实际协议中 A_1 的策略, 但 B_2 会根据 A_2 的策略决定给 TTP 发送什么消息. B_2 需要借助 A_2 来获得自己要发送给 TTP 的消息 $A_2(y)$, 即实际执行协议时 A_2 使用的数据. B_2 将 $A_2(y)$发送给 TTP, TTP 给 B_2 返回 $F(x,A_2(y))$(这个结果也给了 B_1, 因为在理想协议中, 只有在 B_1 得到结果而且没有中止协议的条件下, B_2 才能得到结果);
2. 现在 B_2 要设法从 $F(x,A_2(y))$得到一个与:

   $$(E(V),\mathbf{Dec}(\pi(C)),\alpha_1,\alpha_2,\mathbf{Dec}(E(v_v)),\alpha_3,F(x,A_2(y))).$$

   计算不可区分的:

   $$(E(V'),Dec(\pi(C')),\alpha_1',\alpha_2',\mathbf{Dec}(E(v_y')),\alpha_3',F(x,A_2(y))),$$

   把它交给 A_2, 并输出:

   $$A_2(E(V'),\mathbf{Dec}(\pi(C')),\alpha_1',\alpha_2',\mathbf{Dec}(E(v_y')),\alpha_3',F(x,A_2(y))).$$

   B_2 只需要执行下列步骤即可:
   - B_2 随机选择 x'使 $F(x',A_2(y))=F(x,A_2(y))$, 并以 x'模拟协议, 即 B_2 扮演 A_1 给 A_2 发送其需要的信息 $E(V')$, 获得 A_2 对信息的响应 C'. 因为 $E(V)$, $E(V')$都是用概率加密算法加密的密文, 而概率加密算法是语义安全的, 所以 $E(V)\overset{c}{\equiv}E(V')$;
   - B_2 得到 C'后对其进行置换得到$\pi(C')$, 证明$\pi(C')$只是 C'的置换, 证明过程得到 α_1'. 解密得到 $\mathbf{Dec}(\pi(C'))$. 由 $C\overset{c}{\equiv}C'$ 可以得到 $\pi(C)\overset{c}{\equiv}\pi(C')$, 进而得到 $\mathbf{Dec}(\pi(C))\overset{c}{\equiv}\mathbf{Dec}(\pi(C'))$;
   - B_2 模拟 A_1 与 A_2 完成协议的剩余部分, 得到 α_2', α_3' 和 $F(x',A_2(y))$. 因为 $F(x',A_2(y))=F(x,A_2(y))$, 所以 $\mathbf{Dec}(E(v_y))\overset{c}{\equiv}\mathbf{Dec}(E(v_y'))$. 零知识证明保证 $\{\alpha_1,\alpha_2,\alpha_3\}\overset{c}{\equiv}\{\alpha_1',\alpha_2',\alpha_3'\}$;
3. B_2 用 $(E(V'),\mathbf{Dec}(\pi(C')),\alpha_1',\alpha_2',\mathbf{Dec}(E(v_y')),\alpha_3',F(x,A_2(y)))$ 调用 A_2, 输出:

   $$A_2(E(V'),\mathbf{Dec}(\pi(C')),\alpha_1',\alpha_2',\mathbf{Dec}(E(v_y')),\alpha_3',F(x,A_2(y))).$$

   这样就得到:

   $$\mathbf{IDEAL}_{F,\bar{B}}(x,y)=\{F(x,A_2(y)),A_2((E(V'),\mathbf{Dec}(\pi(C')),\alpha_1',\alpha_2',\mathbf{Dec}(E(v_y')),\alpha_3',F(x,A_2(y)))\},$$

   使得:

   $$\{\mathbf{IDEAL}_{F,\bar{B}}(x,y)\}\overset{c}{\equiv}\{\mathbf{REAL}_{\Pi,\bar{A}}(x,y)\}.$$

现在我们转向 A_2 诚实的情况. 这种情况下, B_2 是协议确定的, 它将根据协议的规定执行协议, 并输出协议规定的输出. 只需要将真实模型敌手 A_1 转化成理想模型敌手 B_1 即可. 在 A_1 不诚实的情况下, 可能在第 4 步中

止协议. (1) 在协议的第 1 步生成的编码不符合约定的形式, 因而在第 3 步通不过零知识证明导致协议中止; (2) 通不过第 4 步的零知识证明而中止; (3) 通不过第 5 步的验证而中止; (4) 通不过第 8 步的零知识证明而中止. 情形(1)~情形(3)等价于 A_1 拒绝参与协议, 没有任何安全问题. 因为到第 5 步为止, 它收到的消息只是用于验证编码是否符合规定形式的消息而没有得到关于 y 的任何消息, 所以不会导致 y 的信息泄露. 它自己发出去的加密 x 的编码是用语义安全的加密算法加密的, 而且只有 A_1 拥有解密密钥, 所以也不会泄露 x 的任何信息. 如果到此中止, 对双方都是安全的.

如果因为通不过第 8 步的零知识证明而中止, 则 A_1 可以得到 $F(x,y)$, 而 A_2 得不到 $F(x,y)$, 只能输出$\perp$. 执行协议Π的时候, A_1 将按照自己的策略输入 $A_1(x)$, 输出什么完全取决于 A_1 的策略和它收到的信息, 此时:

$$\mathbf{REAL}_{\Pi,\bar{A}}(x,y)=\{A_1(x,C,E(v_y),F(A_1(x),y)),\perp\}.$$

如果协议执行完毕没有中止, A_2 将收到 $F(A_1(x),y)$, 此时:

$$\mathbf{REAL}_{\Pi,\bar{A}}(x,y)=\{A_1(x,C,E(v_y),F(A_1(x),y)),F(A_1(x),y)\}.$$

在理想模型中, B_1 接受 x, 局部运行 A_1 得到 A_1 在执行实际协议时将会发送的信息 $A_1(x)$. B_1 将 $A_1(x)$发给 TTP, 如果实际协议中 A_1 没有通过第 8 步的零知识证明而中止, 在理想协议, 在 B_1 通知 TTP 不给 B_2 发送结果, B_2 将得到$\perp$; 如果协议执行完毕, 在理想模型中, B_1 通知 TTP 也给 B_2 发送结果, B_2 将得到 $F(A_1(x),y)$. B_2 遵守协议Π并输出自己从 TTP 得到的 $F(A_1(x),y)$或者$\perp$.

现在 B_1 要利用 $F(A_1(x),y)$, 调用 A_1 得到一个 $A_1(x,C',E(v_{y'}),F(A_1(x),y'))$, 使得:

$$(A_1(x,C,E(v_y),F(A_1(x),y)))\overset{c}{\equiv}(A_1(x,C',E(v_{y'}),F(A_1(x),y'))).$$

B_1 执行下列步骤即可:

1. B_1 随机选择一个 y'使得 $F(A_1(x),y')=F(A_1(x),y)$. B_1 以 y'模拟实际协议, 即 B_1 以 y'扮演成 A_2 与 A_1 执行协议. 给 A_1 提供执行协议Π所需要的信息, 接收 A_1 发送给它的消息. 此过程仍然需要 A_1 生成 ElGamal 密码系统的公钥;
2. B_1 请求 A_1 和它执行协议的第 1 步和第 2 步. 接收 A_1 生成的 $E(V)$, 利用 $E(V)$生成 C'发送给 A_1;
3. B_1 和 A_1 完成协议的剩余部分, 选择 $E(v_{y'})$重随机化之后发给 A_1. 用$(x,C',E(v_{y'}),F(A_1(x),y'))$调用 A_1, 输出 $A_1(x,C',E(v_{y'}),F(A_1(x),y'))$.

如果在(B_1,B_2)参与的理想协议中, B_1 通知 TTP 不给 B_2 发送结果, 我们定义:

$$\mathbf{IDEAL}_{F,\bar{B}}(x,y)=\{A_1(x,C',E(v_{y'}),F(A_1(x),y')),\perp\}.$$

如果 B_1 通知 TTP 给 B_2 发送结果, 我们定义:

$$\mathbf{IDEAL}_{F,\bar{B}}(x,y)=\{A_1(x,C',E(v_{y'}),F(A_1(x),y')),F(A_1(x),y')\}.$$

无论哪种情况, 实际协议和理想协议中, A_2 和 B_2 的输出是相同的. 如果 x 不变, $A_1(x)$就不变, 因而 $A_1(x)$的编码 V 是不变的. 因为 ElGamal 是语义安全的概率加密算法, 因而两次加密的 $E(V)$是计算不可区分的, 进而从 $E(V)$计算得到的 C 和 C'是计算不可区分的. 在解密前 $E(v_y)\overset{c}{\equiv}E(v_{y'})$, 因为 $F(A_1(x),y)=F(A_1(x),y')$, 解密之后 $\mathbf{Dec}(E(v_y))=\mathbf{Dec}(E(v_{y'}))$, 不可能通过解密进行区分. 所以 $E(v_y)\overset{c}{\equiv}E(v_{y'})$ 始终是成立的, 所以:

$$\{\mathbf{REAL}_{\Pi,\bar{A}}(x,y)\}\overset{c}{\equiv}\{\mathbf{IDEAL}_{F,\bar{B}}(x,y)\}.$$

综上所述, 在真实协议中, 任何可接受的概率多项式时间的算法对 $\bar{A}=(A_1,A_2)$ 都存在一个在理想模型中可接受的概率多项式时间的算法对 $\bar{B}=(B_1,B_2)$, 使得:

$$\{\mathbf{IDEAL}_{F,\bar{B}}(x,y)\}\overset{c}{\equiv}\{\mathbf{REAL}_{\Pi,\bar{A}}(x,y)\}.$$

因此, 协议在恶意模型下是安全的.

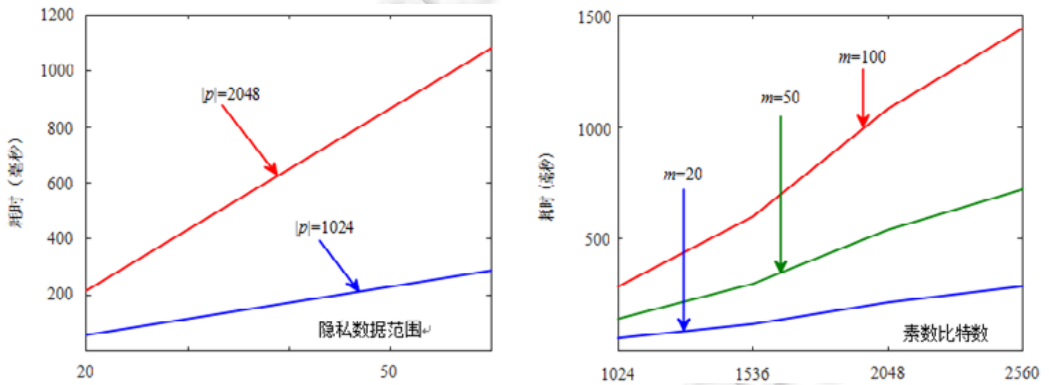
注: 我们强调协议并不能阻止参与者进行欺骗, 但如果参与者进行欺骗, 就能够被发现. 在执行协议时, 如果 Alice 诚实, 她就能够发现 Bob 的欺骗; 如果 Bob 诚实, 也能发现 Alice 的欺骗. 但如果 Alice 和 Bob 是

恶意的, 理论上已经证明: 在这种情况下, 不可能设计出安全的协议[7].

5.2 计算复杂性与通信复杂性

协议的第 1 步, Alice 需要加密 m 个数据, 加密一个数据需要两次模指数运算, 所以第 1 步需要 $2m$ 次模指数运算. 第 2 步生成用于验证的 C, 需要 $2(m-1)$次模指数运算. 第 3 步要证明洗牌需要 $16m$ 模指数运算(经过优化[15]). 第 4 步证明正确解密需要 $4m$ 次模指数运算. 第 6 步重随机化选择的密文需 2 次模指数运算. 第 8 步证明正确解密需要 4 次模指数运算. 协议共需要 $24m+4$ 次模指数运算. 这表明, 协议的计算复杂性与数据范围成正比. 我们用 Pycharm+Python 3.9+gmpy2 编程实现协议并测试了协议的效率. 测试环境是 Intel(R) Core(TM) i5-94000 CPU@2.90GHz 2.9GHz 处理器, 8.0GB 内存, 64 位 Windows 10 操作系统. ElGamal 密码系统的参数 p 为 1024, 2048 比特的素数, 比较的数据满足 $1\leqslant x,y\leqslant m$, 当 m 取不同值时的测试结果如图 2(a)所示, 这与理论分析是一致的.

设置 m=20, 50, 100, p 的比特数为 1024, 1536, 2048, 2560 时, 采用 1 000 次测试的平均时间, 测得的协议运行时间如图 2(b)所示(忽略通信时间). 从图 2(b)可以看出: 对于固定的 m, 模指数运算的模在 1024~2560 比特范围内, 协议的运行时间与模指数运算的模的大小近似成正比; 当数据范围位于 20~100 之间, 协议的主要运算-模指数运算最多耗时不到 2 秒, 因此协议是可行的.



(a) 运行时间随数据范围变化规律 (b) 运行时间随模型数变化规律

图 2 协议效率测试结果

我们用协议发送消息的次数度量通信复杂性, 协议的通信复杂性(包括准备阶段)是 8 次, 是常数次且独立于安全参数和输入数据.

6 相关工作

姚期智教授提出百万富翁问题之后, 又提出了利用不经意传输和混淆电路(garbled circuit)构造任何安全多方计算问题通用解决方案的方法[6]. Goldreich 等人对安全多方计算进行了深入的研究, 提出了利用秘密共享和算术电路构造任何安全多方计算问题通用解决方案的方法[7,16]. 他们的工作证明了任何安全多方计算问题都是可解的, 奠定了安全多方计算的理论基础. 这两种通用解决方案都可以用来解决保密比较问题, 也可以在此基础上借助 cut-and-choose 技术[17]构建抗主动攻击的解决方案[18,19]. 但基于电路的解决方案电路设计困难, 空间复杂性和通信复杂性高, 电路不能重用, 这些问题使得基于电路的解决方案的使用价值很有限. 此外, Goldreich 设计了一个编译器[7,16], 它可以将任意半诚实模型下安全的协议转化为抗主动攻击的协议. 但因为编译器要利用 NPC 问题的零知识证明阻止主动攻击, 这使得转化后协议的计算复杂性提高几个数量级. 这样的编译器有重要的理论意义, 但用编译器生成的抗主动攻击协议解决实际问题也是不实际的.

文献[20]利用门限解密的 lifted ElGamal 密码系统、零测试协议(保密测试一个密文是否是零的密文)和批量相等测试协议(相当于测试两个向量的对应分量是否都相等)构造了一个抗主动攻击的协议, 但是没有给出

安全性定义, 也没有证明协议的安全性. 协议采用了逐比特比较的原理, 在数据比特数较少时计算复杂性高于本文提出的协议, 比特数较多时计算复杂性远高于文献[14]中方案的计算复杂性. 文献[21,22]的方案经过修改可以抵抗主动攻击, 但协议需要半诚实的计算者帮助, 这背离了安全多方计算研究的初衷, 不适用通常的 SMC 场景.

文献[23]声称将文献[8,24]的半诚实模型下的解决方案改造成可以抵抗主动攻击的解决方案. 文献[8,24]声称其方案的主要优势是不需要用比特分解的方法直接比较两个任意正整数的大小(本文的方案也没有用比特分解), 但实际上, 文献[8]给出的方案只适合比较很小的正整数(10 以内); 文献[24]的方案稍好一些, 也只适合比较 1 000 以内的正整数.

文献[8,24]的方案建立在文献[24]提出的密码系统(称为 CEK 密码系统)基础之上, 且文献[24]的方案还需要借助一个加法同态的密码系统. CEK 密码系统的安全性尚没有经过充分分析, 还很难说以此密码系统为基础构建的协议是安全的; 方案的安全性除了依赖 CEK 密码系统的安全性之外, 还依赖一些新的假设, 如小 RSA 子群判定假设、私有 RSA 子群问题; 方案基于的 CEK 密码系统需要多个公钥参数(有 n, a, d, p_0, g, h), 而且各参数都有严格的要求, 参数之间互相约束, 生成这些参数比较困难.

文献[23]没有明确说其协议是以文献[24]的协议为基础设计的, 还是以文献[8]的协议为基础设计的, 但从给出的具体协议看, 应该是以文献[8]的协议为基础设计的(即文献[23]+文献[8]的形式), 因为给出的具体协议没有文献[24]协议的相等判定过程. 这样的话, 协议的效率是很低的. 因为 CEK 密码系统不仅要求 g 在 Z_p^*, Z_q^* 中都是阶数为 p_0^d 的子群的生成元, 要求 h 满足 3 个条件等, 而且还要求 n 随着被比较数的增大呈指数增大, 如果比较的数范围为 0~10, 要求 $n=pq$ 的比特数要大于 2 752, 其中, p, q 为两个大素数. 如果要比较的数范围为 0~100, 要求 n 的比特数要大于 22 912, 少于这个比特数就不安全(相应地要求每个素数的比特数要大于 11 456. 在我们的 Pycharm+Python 3.9+gmpy2 测试环境中, 生成两个 1 024 比特的大素数耗时 0.22 s, 生成两个 4 096 比特的大素数耗时 3.6 s, 生成两个 8 192 比特的大素数耗时 225 s, 要生成两个 11 456 比特的素数耗时 3 061 s, 要生成满足协议要求的素数还要花费更长的时间. 因此, 文献[23]+文献[8]的方案几乎是不可接受的).

本文的方案是建立在 ElGamal 密码系统基础上的, ElGamal 密码系统经过国际密码学界近 40 年充分分析并被广泛使用, 其安全性已经得到确认; 方案对密码系统参数没有要求, 只是随着被比较的数的增大, 需要的模指数运算次数呈线性增加, 方案基于的密码系统的公钥参数少, 参数的要求比较简单, 因此参数生成容易. 因而本文的方案使用更加灵活. 两种方案的性能对比见表 1.

表 1 解决方案的性能比较

	数据范围	系统的模	系统安全性	困难假设	公钥参数	参数生成
协议文献[23]+文献[8]	0~100	22 912	未经充分分析	RSA, 小 RSA 子群判定	7	困难
本文协议	0~100	1 024	经过 36 年分析	离散对数	3	容易

我们在文献[14]中利用分割选择(cut-and-choose)[17]原理构建了抗主动攻击的保密比较问题解决方案, 也证明了协议的安全性, 但协议只适合双方的数据范围未知或者说范围很大的情况, 当数据范围较小(例如 1~10000)时使用有可能是不安全的. 而实际中, 有很多情况下机密数据的范围已知而且范围较小, 例如人的年龄(0~100)、血压(60~200)、体重(50~100)、工资(4000~10000)、学生的成绩(60~100)、班级的人数(20~60)、气温(0~42)、一个部门的职工人数、旅行的城市个数(100 个城市)等都是在一个较小的范围, 这种情况下还没有解决方案. 因此, 要彻底解决抗主动攻击的保密比较问题还有很多工作要做, 本文是这种努力的一部分.

7 结 论

恶意模型下的 SMC 是目前 SMC 研究的热点也是难点. 要解决恶意模型下任意 SMC 问题, 必须首先解决一些基本问题. 保密比较问题就是这样一个基本问题, 其解决方案是很多隐私保护解决方案必不可少的基础

模块, 对解决这些隐私保护问题有重要的作用. 我们曾利用 cut-and-choose 方法解决了当保密数据的范围未知或者范围很大时恶意模型下的保密比较问题, 本文又利用零知识证明和保密洗牌协议设计了当保密数据范围比较小时的解决方案, 两者合起来彻底解决了保密比较问题, 也找到了一个新的抵抗主动攻击的方法, 为设计更多的抗主动攻击的隐私保护协议奠定了基础. 今后, 我们要在此基础上研究更多恶意模型下的隐私保护问题, 并考虑不需要区分隐私数据范围的通用的抗主动攻击的保密比较问题的解决方案.

References:

[1] Yao AC. Protocols for secure computations. In: Proc. of the 23rd Annual Symp. on Foundations of Computer Science. 1982. 160−164.

[2] Goldwasser S. Multi-Party computations: Past and present. In: Proc. of the 16th Annual ACM Symp. on Principles of Distributed Computing. 1997. 1−6.

[3] Cramer R, Damgard IB, Nielsen JB. Secure Multiparty Computation and Secret Sharing. London: Cambridge University Press, 2015.

[4] Zhao C, Zhao SN, Zhao MH, *et al*. Secure multi-party computation: Theory, practice and applications. Information Sciences, 2019, 476(5): 357−372.

[5] Xu QL, Tang CM. Preface of special issue on secure multi-party computing technology. Journal of Cryptologic Research, 2019, 6(2): 191−193 (in Chinese with English abstract).

[6] Yao AC. How to generate and exchange secrets (extended abstract). In: Proc. of the 27th Annual Symp. on Foundations of Computer Science. 1986. 162−167.

[7] Goldreich O. The Fundations of Cryptography—Vol.2: Basic Applications. London: Cambridge University Press, 2004.

[8] Bourse F, Sanders O, Traoré J. Improved secure integer comparison via homomorphic encryption. In: Proc. of the CT-RSA. 2020. 391−416.

[9] Abspoel M, Bouman NJ, Schoenmakers B, *et al*. Fast secure comparison for medium-sized integers and its application in binarized neural networks. In: Proc. of the CT-RSA. 2019. 453−472.

[10] Gao CZ, Cheng Q, He P, *et al*. Privacy-Preserving naive Bayes classifiers secure against the substitution-then-comparison attack. Information Sciences, 2018, 444: 2−88.

[11] Miyahara D, Hayashi YI, Mizuki T, *et al*. Practical card-based implementations of Yao's millionaire protocol. Theoretical Computer Science, 2020, 803: 207−221.

[12] ElGamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. on Information Theory, 1985, 31(4): 469−472.

[13] Fouque PA, Poupard G, Stern J. Sharing decryption in the context of voting or lotteries. In: Proc. of the Int'l Conf. on Financial Cryptography. 2000. 90−104.

[14] Li SD, Wang WL, Du RM. The protocol for millionaire's problem in the malicious model. SCIENTIA SINICA Informationis, 2020, 50(12): 1−14 (in Chinese with English abstract).

[15] Bayer S, Groth J. Efficient Zero-Knowledge argument for correctness of a shuffle. In: Proc. of the 31st Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. 2012. 263−280.

[16] Goldreich O, Micali S, Wigderson A. How to play any mental game. In: Proc. of the 19th Annual ACM Symp. on Theory of Computing. 1987. 218−229.

[17] Lindell Y. Fast cut-and-choose based protocols for malicious and covert adversaries. Journal of Cryptology, 2016, 29(2): 456−490.

[18] Hazay C, Ishai Y, Venkitasubramaniam M. Actively secure garbled circuits with constant communication overhead in the plain model. In: Proc. of the Theory of Cryptography Conf. 2017. 3−39.

[19] Wang X. A new paradigm for practical maliciously secure multi-party computation [Ph.D. Thesis]. MD: University of Maryland, College Park, 2018.

[20] Peng K, Boyd C, Dawson E, *et al*. An efficient and verifiable solution to the millionaire problem. In: Proc. of the Int'l Conf. of Information Security and Cryptology. 2004. 51−66.

[21] Damgård I, Geisler M, Krøigård M. Homomorphic encryption and secure comparison. Journal of Applied Cryptology, 2008, 1(1): 22–31.

[22] Veugen T. Improving the DGK comparison protocol. In: Proc. of the IEEE Int'l Workshop on Information Forensics and Security. 2012. 49–54.

[23] Eskeland S. Privacy-Preserving greater-than integer comparison without binary decomposition in the malicious model. In: Proc. of the 17th Int'l Joint Conf. on e-Business and Telecommunications. 2020. 340–348.

[24] Carlton R, Essex A, Kapulkin K. Threshold properties of prime power subgroups with application to secure integer comparisons. In: Proc. of the CT-RSA. 2018. 137–156.

附中文参考文献:

[5] 徐秋亮, 唐春明. 安全多方计算技术专栏序言. 密码学报, 2019, 6(2): 191–193.

[14] 李顺东, 王文丽, 杜润萌. 抗恶意敌手的百万富翁问题解决方案. 中国科学(信息科学), 2020, 50(12): 1–14.

李顺东(1963－), 男, 博士, 教授, 博士生导师, 主要研究领域为信息安全, 密码学.

陈明艳(1996－), 女, 硕士, 主要研究领域为信息安全, 密码学.

王文丽(1991－), 女, 博士, 主要研究领域为信息安全, 密码学.

汪榆淋(1997－), 女, 硕士, 主要研究领域为信息安全, 密码学.