计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2017.20170461
54(10):2153-2169,2017

隐私保护集合交集计算技术研究综述

申立艳[1,2] 陈小军[2] 时金桥[2] 胡兰兰[2]
[1](中国科学院大学网络空间安全学院 北京 100049)
[2](中国科学院信息工程研究所 北京 100093)
(shenliyan@iie.ac.cn)

Survey on Private Preserving Set Intersection Technology

Shen Liyan[1,2], Chen Xiaojun[2], Shi Jinqiao[2], and Hu Lanlan[2]
[1] (*School of Cyber Security, University of Chinese Academy of Sciences, Beijing* 100049)
[2] (*Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093)

Abstract The private set intersection (PSI) is a specific application problem that belongs to the field of secure multi-party computation. It not only has important theoretical significance but also has many application scenarios. In the era of big data, the research on this problem is in accord with people's increasing privacy preserving demands at the same time to enjoy a variety of services. This paper briefly introduces the basic theory of secure multi-party computation, and highlights the two categories of current mainstream research methods of PSI under the framework of secure multi-party computation: the traditional PSI protocols based on the public key encryption mechanism, garbled circuit, oblivious transfer and the outsourced PSI protocols based on the untrusted third party service provider. Besides, we have briefly summarized the characteristic, applicability and complexity of those protocols. At the same time, the application scenarios of privacy preserving set intersection problem are also explained in detail, which further reflects the practical research value of the problem. With the deep research on the PSI problem, researchers have designed a set of private protocols that can quickly complete set intersection of millions of elements in the semi-honest model.

Key words private set intersection (PSI); secure multi-party computation; oblivious transfer; garbled circuit; oblivious pseudorandom function evaluation; oblivious polynomial evaluation; cloud computing

摘 要 隐私保护集合交集(private set intersection, PSI)计算属于安全多方计算领域的特定应用问题,不仅具有重要的理论意义也具有很强的应用背景,在大数据时代,对该问题的研究更是符合人们日益强烈的在享受各种服务的同时达到隐私保护的需求.对安全多方计算基础理论进行了简要介绍,并重点介绍了目前主流的安全多方计算框架下2类PSI研究技术:传统的基于公钥加密机制,混乱电路,不经意传输的PSI协议和新型的云辅助的PSI协议,并对各类协议的过程、适用性、复杂性进行简要分析总结.同时,也对隐私保护集合交集问题的应用场景进行详细说明,进一步体现对该问题的实际研究价值.随着对该问题的不断深入研究,目前已经设计了在半诚实模型下快速完成上亿元素规模的隐私集合求交集协议.

收稿日期:2017-06-11;**修回日期**:2017-07-28
基金项目:国家自然科学基金项目(61602474)
This work was supported by the National Natural Science Foundation of China (61602474).
通信作者:陈小军(chenxiaojun@iie.ac.cn)

关键词 隐私保护集合交集;安全多方计算;不经意传输;混乱电路;不经意伪随机函数计算;不经意多项式计算;云计算

中图法分类号 TP309

随着通信技术、网络技术等的快速发展以及移动计算、云计算、分布式计算等广泛应用,虚拟网络和人们的生活联系更加紧密,互联网大数据的各种应用渗透到人们的社交、购物、出行等方方面面.这些应用让人们享受到更加便捷的服务,但同时大量有价值的客户信息、个人的隐私记录、企业运营数据不断被挖掘,人们的隐私受到越来越强烈的威胁.因此,大数据时代的隐私保护问题成为学术界及工业界关注的焦点.

从数据生命周期角度来看,数据经历了数据发布、数据存储、计算挖掘和数据使用等环节,据此研究者提出了大数据发布隐私保护技术、大数据存储隐私保护技术、大数据计算和挖掘隐私保护技术以及大数据访问控制技术等来从各个环节保护数据的隐私[1].而在这些大数据隐私保护技术里,更基本的技术是隐私保护的数据计算,指能够满足人们在不泄露额外信息的同时,完成隐私数据之间的计算任务,比如相似性计算[2-3]、距离计算[4-6]等任务.本文主要关注其中的一类即隐私保护集合交集计算技术.

常用的大数据隐私保护技术手段包括匿名、失真、加密、安全多方计算等的方法[1].安全多方计算作为一种特殊的分布式环境下的隐私保护计算方法在近年来受到研究者的普遍关注,已经成为密码学领域的重要研究方向.安全多方计算是由图灵奖获得者姚期智在1982年提出[7],主要解决在分布式计算场景下互不信任的数据拥有者安全协同计算问题,既实现了资源共享又保证了数据隐私性需求.安全多方计算包含的内容十分丰富,包括基础理论模型研究[8-10]、基础密码协议研究、特定应用协议等的研究.基础密码协议包含混乱电路(garbled circuit, GC)、不经意传输(oblivious transfer, OT)、秘密共享(secret sharing, SS)、同态加密(homomorphic encryption, HE)、零知识证明(zero knowledge proof)、掷币协议(coin tossing)、比特承诺(bit commitment)协议等,这些密码协议都可以看作特殊的安全多方计算问题,由一组存在各种信任关系的参与者通过交互或非交互操作来完成某一功能函数的安全计算;特定应用协议研究包括隐私集合运算[11-12]、隐私保护数据挖掘[13-16]、隐私保护信息检索[17-19]、几何计算[20-21]等.基础密码协议成为应用协议设计的基本工具,其安全性和效率直接影响调用它们的应用协议的安全性和效率.

隐私保护集合交集(private set intersection, PSI)计算是安全多方计算领域的一个重要方面,不仅在科学计算中有很好的体现,在现实生活中很多数据都可以用集合来表示,并用集合间的隐私保护计算来完成一些数据计算问题,因此PSI计算有很广泛的应用场景.对该问题的具体定义是在不泄露各自参与方输入信息的前提下,协同计算输入集合的交集.典型的应用场景有隐私保护相似文档检测、私有联系人发现、安全的人类基因检测[22]、隐私保护的近邻检测[23]、隐私保护的社交网络关系发现[24]、在线推荐服务和婚恋网站配对等.在数据由不同管理者持有的条件下,PSI计算达到一种保护隐私与信息共享的共赢.

PSI计算研究由Freedman等人在2004年提出[25],借助不经意多项式求值和同态加密实现.这种实现受限于基础密码协议的计算代价,传统的应用系统还是采用不安全Hash协议① 实现PSI,即对参与方的集合分别进行Hash映射,在Hash值基础上求集合交集.显而易见,这种方法很容易受到碰撞攻击.随着近年来对PSI问题的深入研究,在NDSS, USENIX, CCS等著名国际信息安全会议上涌现了大量的相关研究成果.对该问题的研究不仅推动安全多方计算基础理论的研究发展,也推动了PSI计算实际应用问题研究的发展.目前某些PSI协议的性能已经得到了很大的提高,在通信和计算复杂度上达到了与不安全的Hash协议同样的量级[26].

PSI计算技术根据是否有第三方参与主要分为两大类:1)传统的PSI计算技术,参与方直接交互执行真实的协议,达到隐私计算集合交集的目的,这类技术根据实现原理又包含基于公钥加密机制、混乱电路和不经意传输等3类PSI协议;2)云辅助的PSI计算技术,也叫外包的隐私保护集合交集计算(outsourced private set intersection, OPSI),主要

① https://www.eff.org/deeplinks/2012/09/deep-dive-facebook-and-datalogix-whats-actually-getting-shared-and-how-you-can-opt

利用云服务器的强大计算资源,完成安全交集计算.与理想协议中的可信第三方一样,云服务器主要承担计算交集的作用,但没有任何输入输出信息.表1为2类方法异同点对比.

Table 1 The Comparison of Two Research Methods for PSI

表1 PSI两类研究方法对比

Comparative Items	The Traditional PSI Protocols	The Outsourced PSI Protocols
Similarities	The PSI protocols are based on the underlying modular cryptography protocols, including symmetric, asymmetric, Hash, digital signature and other algorithms, and all the input information is encrypted and confused. Security model definition and security proof method under the framework of secure multi-party computation are also applied in the two kinds of PSI protocols.	The PSI protocols are based on the underlying modular cryptography protocols, including symmetric, asymmetric, Hash, digital signature and other algorithms, and all the input information is encrypted and confused. Security model definition and security proof method under the framework of secure multi-party computation are also applied in the two kinds of PSI protocols.
Participants	Participants	Participants and the Untrusted Third Party Service Provider
Complete the Calculation	Participants	The Untrusted Third Party Service Provider
Guarantee the Correctness	Participants	The Untrusted Third Party Service Provider
Guarantee the Security	Participants	Participants

本文拟对PSI技术的研究现状进行全面的梳理,针对传统的PSI计算技术和云辅助的PSI技术的具体研究思路、研究方法、适用的安全假设、敌手模型和性能开销进行详细介绍和对比,并对基于PSI计算技术的应用研究做了简要的整理,对当前PSI计算技术存在的问题和研究趋势进行总结和建议.

1 基础原语

1.1 安全性证明方法

任何提出的安全协议都需要进行安全性证明,一般来说,在安全多方计算中提出的安全协议通过与理想模型进行对比来证明其安全性.所谓理想模型是指存在可信的第三方,其计算能力是概率多项式时间,通过安全信道获得各参与方的隐私输入并计算功能函数 f,将结果秘密发送给协议参与方.对于所提出的安全协议在证明其安全性时,需要对比理想模型来说明每一个参与方都不会获得更多的信息,这种安全性证明方法叫理想-现实模拟方法[27];如果放宽安全性要求,只需某一参与方遵循安全要求的话,称为单边模拟方法.

1.2 安全模型

在协议的安全性证明过程中,有2类非常重要的基础模型,分别是标准模型和随机预言模型.标准模型(standard model, Std)指不依赖任何假想模型,仅依赖于被广泛接受的困难性假设(如因子分解、离散对数等),这些数学难题是攻击者在多项式时间内不可解的.仅使用困难性假设证明安全的机制称为在标准模型下是安全的.随机预言模型(random oracle model, ROM)[28]比标准模型多了一个随机预言机的假设,随机预言机假设存在一个公共的、随机选择的函数(理论上的黑盒子),只能通过问询的方式进行计算,对每一次查询都从输出域中均匀随机地输出一个响应,对相同的查询总是得到相同的响应.由于现实中没有函数能够实现真正的随机函数,因此随机预言模型下可证明安全的方案在实际应用中通常用Hash函数进行实例化.

1.3 敌手模型

根据敌手腐败参与方的行为方式,敌手模型一般分为3类[27]:

1) 半诚实模型(honest but curious adversary, HbC).协议的各参与方遵守协议的执行过程,但可以在协议执行过程中,根据输入和协议的输出信息推断其他参与者的信息.

2) 恶意模型(malicious adversary, Mal).参与者不遵守协议的执行过程,可能拒绝参与协议、修改隐私的输入集合信息、提前终止协议的执行等,因此需要使用更多的密码协议或技术(位比特承诺协议、零知识证明等)来保证计算结果的正确性.

3) 隐蔽敌手模型(covert adversary).是一种安全性介于半诚实模型和恶意模型之间的更符合真实场景的模型,由于担心恶意行为被协议检测出来并受到惩罚,隐蔽敌手使其恶意行为混淆在正常行为中,只能以一定的概率被检测到.

1.4 基础协议

不经意传输协议(OT)是基于公钥密码体制的密码学基本协议,是安全多方计算的基石.最基本的二选一 OT 协议的发送方 Bob 输入 2 个随机位串 (x_0,x_1),接收方 Alice 输入选择向量 $\boldsymbol{c}$;协议结束后 Alice 获得选择向量对应的位串 x_c,对另一个位串 x_{1-c} 一无所知;Bob 的输出为空[29].在 1988 年 Impagliazzo 和 Rudich 证明了从不经意传输协议到单向函数的黑盒式规约蕴含着另一个难以被证明的问题,即 $P\neq NP$ 的问题,并表示不存在黑盒方式的 OT 协议[30],因此 OT 协议通常是基于公钥密码体制来构造.然而在安全多方计算应用中一般需要大量的 OT 协议作为子协议,计算复杂的模指数运算,使得 OT 协议的实用价值不高.1996 年 Beaver 依据混合加密构想提出了第 1 个非黑盒方式的不经意传输扩展协议[31],可以执行少数基础 OT 协议(传统的基于公钥加密算法的 OT 协议)来构造大量的 OT 协议,然而 Beaver 提出的协议需要计算复杂的伪随机发生器,在实际中也不高效,但是扩展协议的思想具有重要的影响.基于 OT 扩展协议的思想 Ishai 等人在 2003 年提出了以黑盒方式构造的 OT 扩展协议[32],将基础 OT 协议和随机预言模型相结合,把少量基础 OT 的计算代价通过对称加密操作均摊到大量的 OT 操作,可以达到 1 min 执行数百万次的 OT 协议,该协议可以同时满足实用性和安全性需求,具有重要的意义并得到很广泛的应用,例如 GMW 协议、姚氏混乱电路、PSI 等.随着人们对实用性要求越来越高,OT 扩展协议得到了快速发展,主要包括对 N 选 1 不经意传输扩展协议[33]及随机 OT[34]协议的提出.

混乱电路(GC)模型最早是由图灵奖获得者姚期智在 1986 年提出的半诚实模型下的姚氏电路[35],用来解决著名的百万富翁问题.姚氏电路主要是将任意功能函数转化为布尔电路,由 Alice 生成混乱电路表、Bob 计算混乱电路;针对每一个电路门进行两重对称加解密运算,调用四选一 OT 协议进行混乱电路表中秘钥信息交换.早期的安全函数计算问题主要采用混乱电路来解决,由于混乱电路对每一位进行电路门计算并且电路门数量巨大,导致计算效率较低,例如计算 AES 加密大约需要 30 000 个电路门,计算 50 个字符串的编辑距离大约需要 250 000 个电路门[36].混乱电路作为通用的安全多方计算工具,可以用来计算任意的功能函数,相对于特定问题的安全协议计算效率较低.针对这些问题,研究者提出了一系列的电路优化策略[37],包括 Free-XOR、行约减[38-39]、Half-Gate 技术[40].此外还提出了新的混乱电路协议,包括由 Goldreich 等人提出基于秘密共享和 OT 协议的 GMW 编译器[8]以及基于剪切-选择技术(cut and choose)[41]的适用于恶意模型的混乱电路等.除了对布尔电路的研究人们也提出了算术电路,完成有限域上加法或乘法运算.混乱电路方法作为安全多方计算问题的一般通用解决方法,近年来得到了快速的发展,已经有很多实用的安全多方计算工具,例如 Fairplay[42], FastGC[43], Oblivm[44], SEPIA[45], VIFF[46], Sharemind[47]等.

同态加密(HE)是满足同态性质的公钥加密技术,属于语义安全的公钥加密体制范畴.同态加密对密文进行某种算术操作(加或者乘),满足对密文计算结果的解密值与对明文进行同样算术操作的值是相同的性质.由于计算是在密文上进行,因此同态加密是一种常用的实现隐私保护的方法.

秘密共享(SS)将秘密以适当的方式分为 n 份,每一份由不同的管理者持有,每个参与者无法单独恢复秘密,只有达到指定数目的参与者才能恢复秘密.构建秘密共享系统的关键是设计好的秘密拆分和恢复方式.第 1 个秘密共享方案是 (t,n) 门限秘密共享方案,由 Shamir[48]和 Blakley[49]分别在 1979 年各自独立提出,他们的方案分别是基于拉格朗日插值法和线性几何投影性质设计的.此后很多研究者提出了不同的秘密共享实现方法,如基于中国剩余定理的秘密共享策略、可验证的秘密共享等.秘密共享在密钥管理分布式数据安全领域有许多应用,如电子投票、密钥托管、电子支付协议等,可以防止密钥存储过于集中,是一种兼顾机密性和可靠性的方法.

1.5 符号说明

为了方便后续的讨论,本文对所有的数学符号含义进行统一约定,如表 2 所示:

Table 2 The Notation and Description

表 2 符号说明

Notations	Description
C,S	Represent the client and the server
X,Y	Represent the sets of client and server
v,w	The size of the collection of client and server
$i\in[1,v]$	The index of client set elements
$j\in[1,w]$	The index of server set elements
x_i,y_j	The ith and jth element of client and server
H,H'	The random oracle
ℓ	The bit length of the string of OT protocols
σ	The bit length of the set elements
β	The number of the Hash bins

2 传统的 PSI 计算技术

传统的 PSI 协议主要是参与方之间通过一系列底层的密码学技术直接进行交互计算,根据底层所采用技术的不同主要分为 3 类协议,分别为基于公钥加密机制的 PSI、基于混乱电路的 PSI、基于 OT 协议的 PSI.

2.1 基于公钥加密体制

基于公钥加密体制的方法主要对集合元素进行复杂的公钥加密操作如同态加密,并在密文上进行计算.根据协议设计思想的不同又分为基于不经意多项式计算(oblivious polynomial evaluation, OPE[50])的 PSI、基于不经意伪随机函数(oblivious evaluation of pseudorandom functions, OPRF)的 PSI 和基于盲签名的 PSI.

2.1.1 基于不经意多项式计算的 PSI

不经意多项式计算的 PSI 协议主要是将参与方集合元素表示为多项式的根,利用多项式的数学性质来计算交集,并采用同态加密算法加密交互过程中的信息来保证协议的隐私性.

最早由 Freedman 等人在 2004 年提出的 PSI 协议就是不经意多项式计算的 PSI 协议[25].其协议的主要过程为:客户端生成同态加密秘钥对(p_k,s_k)并发送公钥给服务器端,将输入集 $X=\{x_1,x_2,\cdots,x_v\}$ 表示为多项式 P 的根 $P(z)=(x_1-z)(x_2-z)\cdots(x_v-z)=\sum_{u=0}^{v}a_uz^u$,利用插值法求得多项式系数$\{a_0,a_1,\cdots,a_v\}$,将多项式系数用 Paillier[51] 或 ElGamal[52] 同态加密算法加密发送给服务器端;服务器端输入集合为 $Y=\{y_1,y_2,\ldots,y_w\}$,对集合中的每一个元素 y,利用同态加密性质计算 $Enc(r.P(y)+y)$,并将计算的密文混淆发送给客户端;客户端解密所有的密文,依次判断解密的结果是否和输入集合 X 中的某一元素 x 相等,相等则说明 x 属于集合交集.该协议中,多项式的次数过高,会导致同态加密运算中指数的计算代价太大.因此作者又采用 Hash 函数将集合中元素映射到 B 个桶中,每个桶最多 M 个元素,在客户端生成 M 个低次多项式,服务器端采用同样的 Hash 函数将元素进行映射,客户端将相对应的桶里的元素和多项式进行集合交集判断;为了减小 M 的值,作者采用了负载均衡 Hash[53] 方法.2014 年 Freedman 等人进一步在文献[25]的基础上改进 PSI 协议[54],客户端和服务器端分别采用不同的 Hash 函数将集合元素进行映射来减少协议的计算复杂度,并将随机 Hash、负载均衡 Hash、布谷鸟散列[55]进行实验对比,其中应用负载均衡 Hash、布谷鸟散列进行集合元素计算的复杂度较低.以上协议被证明在标准模型下对半诚实敌手是安全的.

与文献[25]中的方法不同,2005 年 Kissner 等人提出不经意多项式计算 PSI 协议的另一种实现[11].该协议依据多项式的数学性质来求集合交集:任意 2 个集合 S_A,S_B 表示为多项式 P_A,P_B,那么 $gcd(P_A,P_B)$表示集合交集 $S_A\cap S_B$,gcd 表示最大公约数.协议的计算过程为:客户端将集合元素表示为 v 次多项式 P_A 的根,在多项式环 $R[x]$上随机选取 v 次多项式 γ_A 并计算 $P_A.\gamma_A$,同理服务端将集合元素表示为 v 次多项式 P_B 的根①,随机选取 v 次多项式 γ_B 计算 $P_B.\gamma_B$ 并将计算结果同态加密发送给客户端;此时,客户端可以计算 $P_A.\gamma_A+P_B.\gamma_B=u.gcd(P_A,P_B)$.根据文献[11]中的定理有:由于 γ_A 和 γ_B 为随机选择的多项式,u 为多项式环上均匀分布的任意 $2v-|S_A\cap S_B|$ 次多项式,其解为集合 S_A 或 S_B 中元素的概率可忽略不计.因此,客户端利用同态加密的性质在密文上完成 PSI 计算.该协议被证明多个参与者在半诚实敌手模型下是安全的.针对恶意敌手模型,文中采用零知识证明技术防止参与方偏离协议执行.基于同样的思想,Dachman 等人提出将多项式进行秘密共享的恶意模型下的 PSI 协议[56].在 2017 年,周素芳等人也在将集合表示成多项式的基础上利用秘密共享协议设计了半诚实模型下高效的 PSI 方案[57].

2017 年陈振华等人依据离散对数困难问题设计了不依赖加密算法的半诚实模型下的 PSI 协议[58].该协议的主要思想是通过客户端集合元素构造多项式 $P(z)=(x_1-z)(x_2-z)\cdots(x_v-z)=\sum_{u=0}^{v}a_uz^u$,并选择任意 v 对非零坐标$(z_i,P(z_i))$,将 v 对坐标进行离散对数处理$(z_i,g^{P(z_i)})$发送给服务端,服务端利用 $v+1$ 对坐标$(z_i,g^{P(z_i)})_{i=1,2,\ldots,v}$和$(y_j,g^0)$计算新的多项式,并将多项式系数的离散对数发送给客户端,并由客户端判断 y_j 是否属于 X.如果 $y_j\in X$,那么客户端和服务器端构造的多项式相同,即多项式系数相同.此过程循环 w 次,可完成 PSI

① 为描述简单起见,此处假设客户端和服务器端集合大小相同均为 v.

计算.该协议虽然没有应用加密算法,但同样包含很多复杂的模指数操作.

针对恶意敌手模型,文献[25]在其基础上提出对恶意客户端采用 cut and choose 技术,对恶意服务器端采用随机预言模型,保证协议的安全性.2010 年 Hazay 等人对文献[25]中协议进一步改进,提出了一种适用于标准模型下恶意敌手的协议[59].该协议将文献[25]中的 $z=Enc(r.P(y)+y)$ 替换为 $z=Enc(r.P(y)+s)$,其中 s 与 y 相互独立,s 由服务器端随机生成,并将 s 的比特承诺值发送给客户端,s 同时也用来计算伪随机函数[60] $r=F_k(s)$.如果客户端某一元素 x 不属于集合交集,那么 $Dec(z)$ 为一随机数,反之客户端会获得 $s=Dec(z)$,客户端通过对比 $Dec(z)$ 的值和 s 的比特承诺值确定元素 x 是否属于集合交集.进一步,客户端通过与服务器端交互可获得伪随机函数值 $r=F_k(s)$,可以用来检验服务器端传输加密数据的正确性.该协议采用比特承诺协议可以有效防止服务端输入数据不一致以及零知识证明协议保证客户端输入多项式不恒为零.

2.1.2 基于不经意伪随机函数的 PSI

基于不经意伪随机函数的 PSI 协议主要思想是客户端和服务器端利用不经意伪随机函数 $F_k()$ 来计算不经意伪随机函数值集合 $\{F_k(x)\}_{x\in X}$,$\{F_k(y)\}_{y\in Y}$,再完成集合交集计算.其中,在不经意伪随机函数的计算过程中,服务器端拥有计算秘钥 k,客户端输入 X 通过协同计算得到 $\{F_k(x)\}_{x\in X}$,服务器端在本地计算 $\{F_k(y)\}_{y\in Y}$ 并发送给客户端.该过程中客户端不能得到任何有关函数 $F_k()$ 和秘钥 k 的信息,服务器端不能得到任何有关 X 的信息,从而保证信息的隐私性.

2008 年 Hazay 等人提出了标准模型下基于不经意伪随机函数的 PSI 协议[61].其伪随机函数定义为[60] $F_k(x)=g^{a_0\times\prod_{i=1}^{n}a_i^{x_i}}$,其中 $k=(p,q,g^{a_0},a_1,a_2,\cdots,a_n)$,$a_0,a_1,a_2,\cdots,a_n\in Z_p^*$ 为服务器端的计算秘钥,g 为阶数为 q 的 Z_p^* 的生成元.其协议过程为:客户端将元素 x 的二进制表示形式作为选择向量,采用 OT 协议获取秘钥 k 中的 $a_i(i\geqslant 1)$,根据 OT 协议,只有当 $x_i=1$ 时才能获取正确的 a_i,客户端计算 $\prod_{i=1}^{n}a_i^{x_i}$ 并发送给服务器,服务器返回 $g^{a_0\times\prod_{i=1}^{n}a_i^{x_i}}$,此时客户端得到了 $F_k(x)$ 的值.以此为基础,作者设计了适用于恶意敌手模型下单边模拟安全的协议和威慑因子为 1/2 的隐蔽敌手模型下的 PSI 协议.

与文献[61]不同,2009 年 Jarecki 等人基于复合剩余假设提出了另一种 PSI 协议[62],该协议采用 Dodis-Yampolskiy 伪随机函数[63]的形式:$F_k(x)=g^{1/(k+x)}$,利用 Camenisch-Shoup[64] 加同态加密和零知识证明技术实现伪随机函数的计算,进一步在伪随机函数值集合上完成交集运算.该协议依赖于公共参考串模型并限制输入集合定义域大小,是恶意模型下可证明安全的 PSI 协议.

2010 年 Jarecki 等人基于不可预测函数又提出了比文献[62]快 20～100 倍的 PSI 协议[65].不可预测函数可看作一类特殊的不经意伪随机函数,其具体定义为 $F_k(x)=H'(H(x),H(x)^k)$.该协议过程为:客户端将元素 Hash 值盲化处理得到 $h_i=H(x_i)^{\alpha_i}$,$\alpha_i\leftarrow Z_q$,将其发送给服务器端;服务器端将收到 h_i 进行伪随机函数计算 $z_i=(h_i)^k$,将 z_i 发送给客户端,同时对秘钥 k 进行零知识证明(恶意模型下);服务器端也在本地计算 $\{F_k(y)=H'(H(y)^k)\}_{y\in Y}$ 发送给客户端;客户端对 z_i 去盲化因子 $H'(z_i^{1/\alpha_i})$ 得到:$\{F_k(x)=H'(H(x)^k)\}_{x\in X}$,并在 $\{F_k(x)\}_{x\in X}$,$\{F_k(y)\}_{y\in Y}$ 的基础计算集合交集.该协议在随机预言模型的 One-More Gap Diffie-Hellman[66] 假设下是可证明安全的.

2.1.3 基于盲签名的 PSI

基于盲签名 PSI 协议的主要思想是客户端将集合元素盲化处理,然后让服务器端对盲化的消息进行签名,最后客户端对签字除去盲因子,得到服务器端关于集合元素的签名 $\{Sign(x)\}_{x\in X}$;服务器端在本地计算集合元素签名 $\{Sign(y)\}_{y\in Y}$ 并发送给客户端;客户端在 $\{Sign(x)\}_{x\in X}$ $\{Sign(y)\}_{y\in Y}$ 的基础计算集合交集.基于盲签名函数的设计本身可保证协议信息的隐私性.

2010 年 De Cristofaro 等人首先提出了一种基于 RSA 盲签名的 PSI 协议[67].协议的基本过程为:服务器端产生 RSA 密钥对 (e,d),客户端先对集合中元素的 Hash 值 $H(x_i)$ 进行盲化处理:$z_i=H(x_i)\times(R_{c_i})^e$,其中 $\forall i,R_{c_i}\leftarrow Z_n^*$;服务器端对收到的元素进行 RSA 盲签名得到 $z_i'=(z_i)^d$,并将 z_i' 发送给客户端;同时服务器端也计算 Y 中每个元素 y_j 的 Hash 值进行 RSA 盲签名得到 $\{t_j\}_{j\in[1,w]}$,其中 $t_j=H'((H(y_j))^d)$,并发送给客户端.客户端对 $\{z_i'\}_{i\in[1,v]}$ 去除盲化因子,然后与 $\{t_j\}_{j\in[1,w]}$ 进行集合交集计算.该协议是针对半诚实模型的,在此基础上针对恶意敌手模型,De Cristofaro 等人又提出了采用零知识证明的 PSI 协议并实现[68-69].

以上所有的 PSI 协议都泄露了参与方集合的基数,2011 年 Ateniese 等人针对这个问题,提出了基于 RSA 假设的一种隐藏客户端集合大小的 PSI 协

议[70].该协议的过程为:首先客户端将集合元素进行 Hash 处理,生成 RSA 累积值:$A=g^{\prod_{l=1}^{v}H(x_l)}$,并将累积值盲化处理后得到 $T=A^{R_c}=g^{R_c\prod_{l=1}^{v}H(x_l)}$ 发送给服务器端,服务端将 T 作用在自身集合 Y 的每一个元素上,得到 $\{t_j\}_{j\in[1,w]}$,$t_j=H'(T^{R_s\times(1/H(y_j))})$,并将 $Z=g^{R_s}$ 和 $\{t_j\}_{j\in[1,w]}$ 发送给客户端;最后,客户端计算 $x_i'=H'(Z^{R_c\times(\prod_{l=1}^{v}H(x_l))/H(x_i)})$,对比 $\{x_i'\}_{i\in[1,v]}$ 和 $\{t_j\}_{j\in[1,w]}$ 得到集合交集.整个协议过程服务器端没有获得任何有关客户端输入集合及集合大小的信息,是在半诚实模型下可证明安全的.

2.1.4 公钥加密的 PSI 性能比较分析

为了对以上各协议性能有更直观的认识,分别从安全模型(标准模型、随机预言模型)、敌手模型(半诚实、恶意敌手)的角度对客户端和服务器端计算和通信复杂度进行渐进分析.分析结果如表 3 所示:

Table 3 Comparison of Private Set Intersection Protocols

表 3 隐私集合交集协议性能比较

Protocol	Security Model	Adversarial Model	Communication Complexity	Computation Complexity of Client	Computation Complexity of Server
FNP04[25]	Std	HbC	$O(w+v)$	$O(v)$ exps	$O(w\log\log v)$ exps
KS05[11]	Std	Mal	$O(w+v)$	$O(wv)$ exps	$O(wv)$ exps
HL08[61]	Std	HbC	$O(w+v)$	$O(v\sigma)$ exps	$O(v\sigma+w)$ exps
JL09[62]	Std	Mal	$O(w+v)$	$O(v)$ exps	$O(w+v)$ exps
HN10[59]	Std	Mal	$O(w+v)$	$O(w+v)$ exps	$O(w\log\log v)$ exps
DT10[68]	ROM	Mal	$O(w+v)$	$O(v)$ 160 b exps mod 1 024 b	$O(w+v)$ 160 b exps mod 1 024 b

Note: exps denotes modular exponentiations operation.

几种协议的通信复杂度一般都与集合元素成线性关系,协议主要的操作均为复杂的模指数操作,计算复杂度通常与模指数的长度成线形关系,由于大部分协议采用同态加密及其他公钥算法指数和模数一般为 1 024 b 以上,而 De Cristofaro 等人提出的 PSI 协议模指数计算中的指数为 160 b,模数为 1 024 b,因此计算复杂度最低.

2.2 基于混乱电路

理论上讲,任意的功能函数均可用基本的电路门形式来表示,即可以通过电路解决任意功能函数计算问题,因此可以用混乱电路的方法计算 PSI 问题.基于混乱电路的 PSI 协议主要思想有 2 种:1)将元素映射到二进制向量,通过向量的与运算求集合交集;2)通过电路解决隐私保护的元素相等性判断问题,然后通过集合中元素的两两比较以 $O(n^2)$ 复杂度计算集合交集.

2012 年 Huang 等人提出了 3 种基于姚氏混乱电路 PSI 协议的实现[71],分别为 BWA(Bitwise-AND),PWC(Pairwise-Comparisons),SCS(Sort-Compare-Shuffle),分别适用于不同的集合规模和集合定义域大小的计算.其中,BWA 协议将集合中 σ 位长的元素表示成长度为 2^σ 位向量,向量中每一位表示集合中一个元素,位向量中某一位为 1 表示元素在集合中,为 0 表示该元素不在集合中.通过混乱电路对参与方集合向量进行位与运算得出交集集合的位向量,进而得到相应元素.PWC 协议主要通过混乱电路对两方的元素进行两两相等判断,首先将 2 个长度为 σ 位的元素进行 XOR 运算,并将计算结果的每个位进行 OR 运算,由于协议采用了 Free-XOR 技术[72],每对元素间相等性判断只需要实现 $\sigma-1$ 个非异或门电路.SCS 协议的主要过程为:客户端和服务器端分别在本地对各自集合中元素进行排序,并通过混乱电路进行按序合并;对合并后集合中相邻元素进行相等性判断,如果相等则为交集中元素;最后将计算结果乱序(分别采用了 3 种方法实现乱序:不经意排序网络、同态加密、Waksman 网络,乱序目的是防止泄露集合元素位置信息)输出给客户端.该协议通过使用不经意扩展传输协议及其他各种电路优化技术,仅需要 $O(\sigma n\log n)$ 非异或门电路操作,大大加快了协议的计算速度.这些实现方案均假设参与方集合没有重复元素,在半诚实敌手模型下是可证明安全的.

2014 年 Pinkas 等人对 Huang 提出的 PWC,SCS 协议进行优化[73],主要是采用目前最快的随机 OT 协议对混乱电路进行优化.由于混乱电路以 OT 子协议为基础完成安全计算,因此 OT 协议的改进

很大程度减少协议的计算和通信代价. 此外 Pinkas 等人也利用 GMW 混乱电路进行 PSI 协议计算,该协议同样采用随机 OT 协议加快 PSI 的计算. 除了从 OT 协议的角度对姚氏混乱电路和 GMW 混乱电路进行优化外,在 PWC 协议中作者采用 Hash 函数将集合元素进行映射来减少两两元素相等判断比较次数,很大程度减少协议计算复杂度.

2016 年 Pinkas 等人通过电路实现了基于不经意伪随机函数的 PSI[74],该协议的关键在于如何高效地实例化 OPRF,作者采用文献[38]中 AES 实例化 OPRF,并分别用姚氏混乱电路和 GMW 混乱电路计算 AES 函数完成 PSI 计算.

表 4 对以上基于混乱电路的协议进行渐进复杂度分析,分析结果如下:

Table 4 Comparison of Private Set Intersection Protocols

表 4 集合交集协议性能比较

Protocol	Cost in non-XOR Gates	Communication Complexity	Best for
BWA	2^{σ}	$O(\kappa 2^{\sigma})$	Small Element Space
PWC	$O(\sigma n^2)$	$O(\sigma\kappa n^2)$	Large Element Space
SCS	$O(\sigma n \log n)$	$O(\sigma\kappa n \log n)$	Large Element Space

Note: n: the size of sets; κ: the symmetric security parameter.

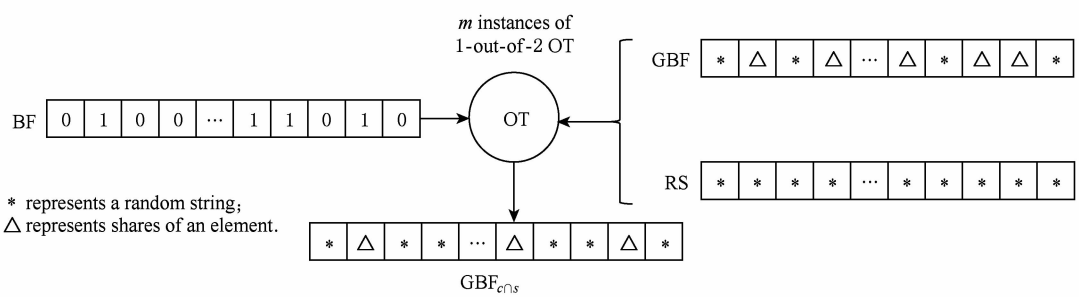
BWA 协议计算代价与 σ 成指数关系,适用于较小的 σ 值. 结合文献[73-74]中实验结果对以上 PWC,SCS 协议进行分析,由于 PWC 协议采用 Hash 函数减少比较次数,无论采用姚氏混乱电路还是 GMW 混乱电路在计算和通信复杂度上都优于 SCS 协议.

尽管混乱电路的 PSI 协议一般具有通用性、灵活性和可扩展性,但功能函数的电路设计一般要求很大的门电路个数和电路深度,因此基于此技术的协议一般计算效率很低,人们更偏向于使用特殊的高效协议.

2.3 基于不经意传输协议

基于不经意传输的 PSI 协议主要是通过 OT 协议进行向量或元素相等性判断来完成集合交集计算. 随着近年来 OT 协议的快速发展,基于 OT 协议的应用协议也得到了快速发展. 特别是基于 OT 协议的 PSI 计算可以快速完成上亿元素规模的交集计算,逐渐可适用于大数据下的应用场景.

2013 年 Dong 等人提出了第 1 个可处理集合元素达到亿级别大小的 PSI 协议[75],该协议基于布隆过滤器(Bloom filter, BF)[76]、秘密共享和扩展不经意传输协议,主要依赖于高效的对称加密操作. 布隆过滤器主要用来判断一个元素是否在集合中,通过引入一定的假阳率来节省大量的存储空间. 由于 2 个集合的布隆过滤器结构直接进行与运算 $BF_c \cap BF_s$ 的结果和集合交集表示成布隆过滤器 $BF_{c\cap s}$ 的结果不一致,通过逻辑与运算求集合交集会泄露集合中的信息,因此作者采用 OT 协议来保证安全性. 该协议的过程为:客户端将集合 n 个元素表示为布隆过滤器结构(布隆过滤器本质为长度为 m 的位向量,每个集合元素用 k 个不同的 Hash 函数$\{h_1, h_2, \cdots, h_k\}$映射到位向量);服务器端将集合元素表示为 GBF(garbled Bloom filter)结构,GBF 与 BF 本质相同,区别在于索引位置存储的是元素基于异或操作的秘密共享份额 r_i,即 $x = r_1 \oplus r_2 \oplus \cdots \oplus r_k$,$x$ 为长度为 σ 位的集合元素;双方执行 m 次 OT 协议,客户端的 BF 结构作为选择向量,服务器端的 GBF 结构为输入数据,由于 2 方采用相同的 Hash 函数映射,相同的元素映射到 BF 和 GBF 上的索引位置一定相同,因此经过 OT 协议,客户端能得到交集中元素的秘密共享份额,进一步通过查询算法可获得集合交集中元素. 该协议示意过程如图 1 所示:



RS represents random string. The client acts as receiver: Uses BF as selection vector. If $BF[i]=0$, the client receives $RS[i]$, if $BF[i]=1$, the client receives $GBF[i]$. The server acts as sender.

Fig. 1 The private set intersection protocol based on Bloom filter.

图 1 基于布隆过滤器的隐私集合交集协议

2016 年 Rindal 等人对文献[75]中协议的恶意模型进行修正,针对恶意的客户端,采用 cut and choose 技术来保证协议的安全并同样采用随机 OT 减少服务器端通信复杂度[77].

2014 年 Pinkas 等人在文献[73]中除了对混乱电路协议优化外,也对 Dong 提出的协议[75]进行优化,使用随机 OT 协议代替扩展 OT 协议,服务器端不需要保存 GBF 结构,节省了大量空间.具体而言,客户端和服务器端分别生成各自集合的布隆过滤器 BF_c,BF_s,并作为 m 次随机 OT 的输入,输出分别为 G_c,G_s,在每一次二选一 OT 计算中,如果 $BF_c[i]=BF_s[i]=1$,那么 $G_c[i]=G_s[i]=random\ string\in\{0,1\}^\ell$;对于 $\forall x_i\in X, y_j\in Y$,客户端计算 $mask_c[i]=\oplus_{l=1}^{k}G_c[H_l(x_i)]$,服务器端计算 $mask_s[j]=\oplus_{l=1}^{k}G_s[H_l(y_j)]$并发送给客户端;客户端检查是否存在 j 满足 $mask_s[j]=mask_c[i]$来判断元素 x_i 是否在交集中.由于随机 OT 比 OT 扩展协议有更低的计算和通信复杂度,优化后的协议效率提升达到 55%~60%.

基于最新提出对 OT 扩展协议的优化技术[33-34],文献[73]中 Pinkas 等人还提出了新的基于 Hash 和随机 OT 协议的 PSI 协议.该协议的基础是通过 σ 次随机 OT 协议对长度为 σ 位的元素进行相等性判断,然后采用 Freedman 在文献[25,54]中提出的 Hash 方法将元素映射到桶中,分桶比较来实现集合交集运算.σ 位元素相等性判断的过程为:利用随机 OT 协议,客户端输出随机串 $s^i_{x[i]}\in(s^i_0,s^i_1)\in\{0,1\}^\ell$,$i=1,2,\cdots,\sigma$,服务器端输出随机串 $s^i_{y[i]}\in(s^i_0,s^i_1)\in\{0,1\}^\ell$,$i=1,2,\cdots,\sigma$.令 $m_s=\oplus_{i=1}^{\sigma}s^i_{y[i]}$,$m_c=\oplus_{i=1}^{\sigma}s^i_{x[i]}$,如果 $m_s=m_c$ 那么元素 x,y 相等.该协议使用随机 OT 协议,效率大大提高,另外由于采用 Hash 算法,相等性判断的复杂度由 $O(n^2)$ 减少到 $O(n\log n)$.

基于 Hash 和随机 OT 协议的 PSI 协议的计算和通信复杂度和元素的位长度 σ 成正比,为了进一步减少计算和通信开销,2015 年 Pinkas 等人基于置换 Hash 的思想,又提出了效率更高的 PSI 协议[78].该协议采用置换 Hash[79]算法,主要思想是将元素 x 表示为左右 2 部分 $x=x_L|x_R$,$|x_L|=\log\beta$,左半部分主要用来表示元素索引位置 $p(x)=x_L\oplus f(x_R)$,其中 f 为随机函数:$[1\cdots 2^{|x_R|}]\rightarrow[1\cdots 2^{|x_L|}]$,右半部分为 Hash 桶中实际存储的元素部分.Hash 桶中只需要存储 $\sigma-\log\beta$ 比特的元素,因此改进的 PSI 协议可以将一次元素相等性判别过程中随机 OT 子协议的执行次数从 σ 减少为 $\sigma-\log\beta$,这进一步降低了协议的计算和通信复杂度.

基于不经意传输的 PSI 协议其效率主要依赖于 OT 子协议的效率,通过采用 N 选一 OT 协议,可进一步减少 OT 协议执行的次数.在 Kolesnikov 等人提出的 N 选一 OT 协议中[33],采用 Walsh-Hadamard (WH)错误纠正码对选择向量进行编码,使得对长度为 σ 位的元素进行相等性判断时可以将随机 OT 协议执行次数从 σ 次减少为 t 次,其中 $t=\sigma/\log N$,但该协议只适用于较小的 N,如 $N=256$. 2016 年 Kolesnikov 等人又提出一种新的协议[80],该协议对选择向量采用伪随机编码的方式,使元素相等性判断执行的 OT 协议的次数不依赖于 σ,即通过一次 OT 协议就可以判断 2 个元素是否相等.作者采用固定秘钥的 AES 实例化伪随机函数.在 σ 取值为 64 或 128 时,改进的 PSI 协议比文献[78]中的协议快 2.3~3.6 倍.

同样,为了使 PSI 协议的效率只与元素个数成正比而不依赖于元素的位长度 σ,2016 年 Pinkas 等人采用新的错误纠正码实现 OT 协议,并在此基础上改进 PSI[74].因为以上 PSI 协议采用的都是基于随机预言模型和半诚实模型下的 OT 扩展协议,因此 PSI 协议一般适用于随机预言模型和半诚实模型.表 5 为基于 OT 的 PSI 协议渐进复杂度分析:

Table 5 Comparison of Private Set Intersection Protocols

表 5 隐私集合交集协议性能比较

Protocol	Computation Complexity	Communication Complexity
Bloom Filter[75]	$O(nk)$	$O(\kappa nk)$
OT[73]	$O(n\sigma)$	$O(\sigma\kappa n)$

Note: n: the size of sets; κ: the symmetric security parameter; k: the number of the Hash functions.

此外作者通过实验对比了半诚实模型下的不安全 Hash 的 PSI 和基于 RSA 公钥加密算法、混乱电路和 OT 协议的 PSI.表 6 为文献[74]中部分实验结果,该实验中用到的 2 台设备为 Intel Haswell i7-4770K,3.5 GHz CPU 16 GB 内存,通过 OpenSSL (v.1.0.1e)密码学库实现论文中的对称加密操作,Miracl(v.5.6.1)和 GMP 库(v.5.1.2)实现 OT 扩展协议和公钥加密操作①,通过实验结果我们可以对各类协议的性能有更直观的认识.

① https://github.com/encryptogroup/PSI

Table 6 Comparison of Private Set Intersection Protocols [74]

表 6 集合交集协议性能比较

PSI Protocol	Runtime/s	Communication Traffic/MB
Hashing	0.7	10
Public Key	818.3	74
Circuit(PWC/OPRF)	83.9	9 170
OT	5.6	107

Note: Protocols on sets with 2^{20} over Gigabit LAN. σ is 32 b and the security parameter is 128 b.

通过表 6 可以分析出基于公钥加密的协议计算复杂度最高,但是具有较小的通信复杂度,适用于参与方计算能力较强但是通信带宽为瓶颈的情景;基于电路和 GBF 的协议计算复杂度优于基于公钥的协议,但是通信复杂度最高;基于 OT 和 Hash 策略的 PSI 协议在计算和通信复杂度上都达到了和不安全 Hash 一样的量级,既满足安全性要求又达到较高的计算效率,适合处理大数据量问题. 以上分析结果反映出了 PSI 协议的效率直接取决于底层基础协议的效率.

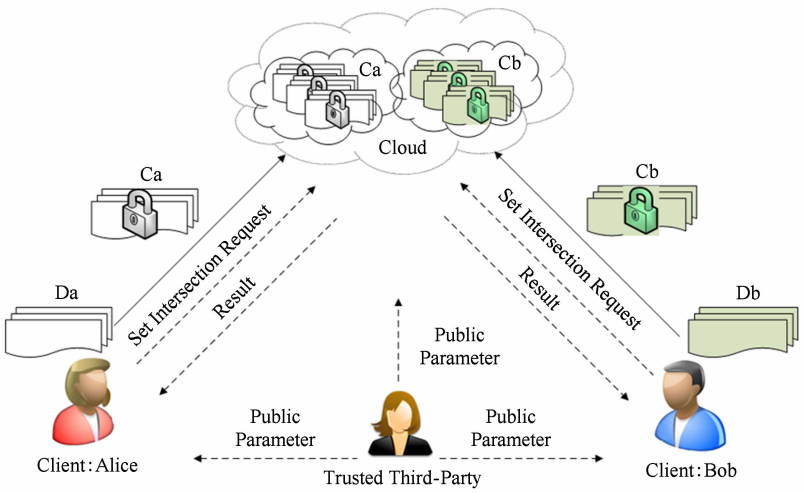
此外,从 2 个角度对协议的适用性进行分析: 1)大规模场景适用性分析. 基于混乱电路的协议需要将电路提前构建并全部加载到内存中,当集合很大时会有内存限制;同样对于布隆过滤器的协议也需要加载全部布隆过滤器结构到内存,因此此类协议不适合处理大数据应用问题. 2)并行性分析. 基于公钥加密的 PSI 处理单元为相互独立的元素因此可以并行化加速计算;基于混乱电路的 PWC,OPRF 协议可并行化处理,而 SCS 协议中门电路都是相互关联的不适合并行化计算;基于 OT 的 PSI 协议由于底层基础模块 OT 扩展协议的可并行化因此也可以进行并行计算.

3 云辅助的 PSI

传统 PSI 协议主要是参与方之间交互完成集合交集计算,然而,有限存储资源和计算资源的终端设备和移动设备无法应对大规模数据集的 PSI 计算需求. 与此同时,随着云计算应用的兴起,大量的存储资源和计算资源以一种按需服务的方式向用户提供,使得大量的应用借助于第三方云计算框架来完成计算. 因此研究者提出了新的借助于云服务器的安全函数计算问题[36]. 其中,外包的隐私集合交集(OPSI)协议作为特殊的云辅助的安全函数计算问题是指存在一个不可信的第三方服务提供商 P,参与方采取一系列方式将输入集合加密盲化处理(如同态加密、伪随机函数、Hash 函数等)后上传到云端,然后由 P 在密文上完成集合交集的计算,在协议执行过程中 P 没有获得任何输入输出信息,主要是利用云服务器的存储和计算资源.

相比于传统的 PSI 协议,在设计云辅助的 PSI 协议时,一般需要满足抗合谋性,因为如果服务器和某一参与方进行合谋,将退化为参与方之间计算能力相差悬殊的传统的安全多方计算协议[81]. 外包的隐私集合交集协议示意图如图 2 所示,其中一部分协议设计需要可信的第三方生成共享秘钥信息.

2012 年 Kerschbaum 提出了 2 种利用单向函数实现抗合谋的外包 PSI 协议[82]:1)安全性较差,允许



Alice and Bob's data sets are denoted by Da and Db, and they respectively encrypt their data sets denoted by Ca and Cb

Fig. 2 Outsourced private set intersection protocol

图 2 外包的隐私集合交集协议

不可信云服务器获得集合交集信息,协议过程为:客户端 A 和 B[①] 分别将集合元素经过 Hash 映射后 $x^*=H(x_1),H(x_2),\cdots,H(x_v)$,$y^*=y_1\oplus H(y_1),y_2\oplus H(y_2),\cdots,y_w\oplus H(y_w)$提交给云端服务器,其中 $H:\{0,1\}^m\rightarrow\{0,1\}^{l+m}$;服务器端计算元素的交叉积 $x^*\times y^*=H(x_1)\oplus y_1\oplus H(y_1),H(x_2)\oplus y_2\oplus H(y_2),\cdots,H(x_w)\oplus y_w\oplus H(y_w)$,当元素属于集合交集时,交叉积的计算结果为元素本身明文值;客户端在第三方服务端的明文计算结果中查询集合交集元素.2)基于 RSA 公钥加密算法,客户端首先协商加密算法秘钥,其中 $(d_{1+}d_2)e=f(\mathrm{mod}\ \phi(n))$,$A$ 获得 d_1,f,B 获得 d_2,f,服务端获得 e;客户端采用 RSA 加密算法将集合元素加密后提交给云服务器,A 的输入 $x^*=x_1^{d_1},x_2^{d_1},\cdots,x_w^{d_1}(\mathrm{mod}\ n)$,$B$ 的输入 $y^*=y_1^{d_2},y_2^{d_2},\cdots,y_{w'}^{d_2}(\mathrm{mod}\ n)$;服务器端计算元素的交叉积 $z=x^*\times y^*=(x_1^{d_1}y_1^{d_2})^e,(x_2^{d_1}y_2^{d_2})^e,\cdots,(x_w^{d_1}y_{w'}^{d_2})^e(\mathrm{mod}\ n)$;客户端 A 在密文上求集合交集 $x\bigcap z$,$x=x_1^f,x_2^f,\cdots,x_v^f(\mathrm{mod}\ n)$.以上 2 种协议分别为随机预言模型和标准模型下可证明安全的,第 1 种方法同不安全 Hash 协议一样存在暴力破解的风险,是牺牲安全性来换取高效性的方案.

Kerschbaum 等人又提出了另一种采用布隆过滤器和同态加密实现的 PSI 协议[83].协议过程为:客户端将集合表示成布隆过滤器结构并采用同态加密技术将布隆过滤器加密后外包给服务器;第三方服务提供商利用加密的布隆过滤器 $Enc(BF_A)$,$Enc(BF_B)$求集合交集并将集合交集的布隆过滤器结构的密文 $Enc(BF_{A\cap B})$返回给客户端,客户端用户对收到的密文解密并结合本地的集合即可计算集合交集.该协议在密文上进行计算主要采用 SYY 技术,SYY 技术[84]是一种在密文上进行某种运算对应于明文的逻辑与运算的技术,在该协议中体现为通过加密布隆过滤器的与运算求明文集合信息,即 $Enc(BF_A)*Enc(BF_B)=Enc(BF_{A\cap B})$,$*$ 表示秘文运算.

2014 年 Liu 等人提出了一种相对简单但是会泄露集合交集基数的 PSI 协议[85],该协议过程为:客户端用户分别各自生成协议中采用的对称和非对称加密算法秘钥;客户端用户生成随机数 $r^I,r^I\in\mathbb{Z}$,作用在数据集的 Hash 值上$\{V_i^A=H(x_i)+r_i^A\}_{i\in[1,v]}$,$\{V_j^B=H(y_j)+r_j^B\}_{j\in[1,w]}$,并对随机数和原始数据集进行对称加密 $Enc(r^I)$;$Enc(S^I)$,将 $V^I,Enc(r^I)$;$Enc(S^I)$上传给云服务器;客户端用户与服务器端进行交互使服务器获得 $C^I=V^I+r^A+r^B$,通过计算 $|C^A-C^B|$是否为 0 判断元素是否属于交集.该协议是半诚实模型下可证明安全的,但是同不安全 Hash 协议一样存在暴力破解的风险.2015 年 Zheng 等人提供了一种可验证机制 PSI[86].

2014 年 Kamara 等人提出了大数据背景下的半诚实、隐蔽敌手和恶意敌手模型下云辅助集合交集方案[26],并假设云服务器是诚实的,即云服务器和客户端不可合谋.在半诚实模型下,客户端协同产生伪随机函数的密钥 k,通过伪随机函数 $F_k(s)$,$s\in S_i$,分别对集合中每个元素进行随机化处理,再通过伪随机置换 π 扰乱集合元素顺序,将处理后的集合发送给云服务器;服务器在集合元素的随机函数上求交集;用户通过伪随机函数的逆过程 $F_k^{-1}(F_k(s))=s$ 获得集合交集.在恶意模型下,为防止第三方服务器恶意返回错误结果,为客户端用户提供了一种验证机制,用来检验返回结果的正确性.主要是通过客户端将集合中每个元素复制 λ 份,当服务器返回求交结果后,客户端用户检查交集中每个元素是否是 λ 份.3 种协议中客户端都需要进行与集合大小成线性关系的伪随机置换操作,第三方服务器只需要在明文上进行集合求交运算.Kamara 等人采用 AES 实现伪随机函数,在半诚实模型下相比不安全的 Hash 方法最多慢 10%,达到较高的计算效率.

2015 年 Abadi 等人提出了利用同态加密和多项式插值实现的 PSI 协议[87],该协议采用文献[11]中的数学原理进行集合交集计算,允许多个客户端用户求集合交集并且一旦将集合外包给第三方服务器就可以进行无限次 PSI 运算.在该协议中作者采用新的多项式表示方法,即任意 $n(n>t)$个点可以插值计算 t 次多项式,通过点对$\{(x_0,y_0),(x_1,y_1),\cdots,(x_{n-1},y_{n-1})\}$,$y_i=P(x_i)$,$0\leqslant i\leqslant n-1$ 来表示多项式,相比于多项式系数 $\vec{a}=(a_0,a_1,\cdots,a_d)$ 表示方法,这种表示可以将多项式乘的复杂度由 $O(d^2)$降低到 $O(d)$.协议过程为:客户端用户生成 Paillier 同态加密秘钥和伪随机函数秘钥,通过伪随机函数生成随机数 r^I,采用同态加密算法加密生成的随机数 r^I;并将集合元素表示为点对多项式形式,

① 为描述简单起见,基于云辅助的 PSI 将传统的 PSI 协议中客户端和服务器端统称为客户端,并假设客户端用户为 $I=\{A,B\}$,S 为参与方的输入集合,A 集合大小为 v,B 集合大小为 w

通过随机数 r_i^I 多项式点对进行盲化处理提交给第三方服务器;服务器端生成 d 次多项式环上的随机多项式 w_I,根据集合交集公式:$S_A \cap S_B = P_A.w_A + P_B.w_B$ 并利用同态加密性质在密文进行集合交集运算并将计算结果返回客户端;客户端之间进行交互获得盲化集合元素的随机数 r_i^I 的加密值,解密处理获得集合交集.在此工作基础上 Abadi 又提出了可验证的云外包 PSI 协议[88],既保证客户端用户输入数据的隐私性又可保证输入数据的完整性.

2016 年李顺东等人利用哥德尔编码和 ElGamal 同态加密算法构造了一种半诚实模型下适用于云计算的集合交集计算方案[89].该方案解决了多个集合间的交集并集问题,但不能解决 2 个集合间的交集问题.

表 7 为云辅助 PSI 协议的计算和通信渐进复杂度分析.分析结果如下:

Table 7 Comparison of Private Set Intersection Protocols

表 7 集合交集协议性能比较

Protocol	Communication Complexity	Computation Complexity of Client	Computation Complexity of the Third Party Server
Ref[82]	$O(w+v)$	$O(m)$ exps	$O(vw)$ exps
Ref[83]	$O(k(w+v))$	$O(m)$ exps	$O(kvw)$ exps
Ref[85]	$O(w+v)$	$O(m)$ sym	$O(w+v)$ add
Ref[26]	$O(w+v)$	$O(m)$ sym	$O(w+v)$ sym
Ref[87]	$O(w+v)$	$O(m)$ exps	$O(w+v)$ exps

Note: exps denotes modular exponentiations operation; sym: symmetric cryptographic operations; add denotes modular additional operation; $m=\max(v,w)$ k: the number of the Hash functions.

通过对以上具体协议的分析,目前已有的云辅助 PSI 主要采用公钥加密和私钥加密 2 种类型,在协议的安全性和高效性上各具有很鲜明的特点,其中 Kamara 提出的协议全部采用对称加密操作,计算复杂度最低,适合大规模数据集计算,但是不能抵抗第三方服务器和参与方的合谋攻击;其他的采用同态加密等公钥加密算法对数据集进行加密计算的 PSI 协议,计算复杂度较高但是达到了较高的安全性,可抵抗合谋攻击.此外,传统的 PSI 协议一般可以经过简单调整转化为云辅助的 PSI 协议,在采用相同的底层密码学技术实现 PSI 协议时,2 类协议的复杂度基本不会有很大偏差.

4 PSI 应用场景

从参与方的角度考虑,PSI 协议应用场景可以分为 2 类:1)场景中参与方均是大型企业或医疗政府军事机构等,都拥有大量隐私、有价值和具有核心竞争力的数据,双方通过安全协作计算在保护自身数据隐私性的同时达到信息共享的目的;2)场景中参与方是不对等的关系,其中一方是通过收集用户的隐私数据来提供更好服务的企业,一方是享受服务的用户,双方通过安全协作计算是为了提供/享受服务的同时保护用户隐私.因此 PSI 的研究具有重要意义,一方面促进具有利益冲突的企业、机构等更好的进行合作;另一方面减轻大数据时代隐私泄露问题带来的危害.对 5 个典型的应用场景简要说明:

1) 隐私保护相似文档检测.隐私保护相似文档检测技术主要应用于如医疗病历共享、一稿多投行为审查等.例如会议或者期刊组织方在收到提交稿时,希望在其他会议或期刊检查一下作者是否存在一稿多投现象.但通常未发表的提交稿是不能对其他会议或期刊组织者公开的,针对这个问题可以通过隐私保护相似文档检测协议进行计算,首先将双方的文档表示成文档指纹集合,通过 PSI 协议求得 2 篇文档的指纹集合交集,进而计算集合的 Jaccard 距离得到 2 篇文档的相似度[90].

2) 安全的人类基因检测.人类基因测序研究及其应用越来越普遍,在很多医疗行业有广泛的应用,比如疾病预测、个人医疗、亲子鉴定等.基因信息的共享和交换有利于加速科学研究发展,提升人类医疗保健质量,不少基因组织通过基因工程项目比如 Personal Genome Project 和 HapMap 致力于建立庞大的公共基因数据库.然而,基因信息中也包含大量的个人隐私,比如人种信息、显性体征以及疾病信息等,基因数据滥用或者非法散播会造成严重的隐

私泄露.为了实现在基因数据上隐私保护的信息共享,Baldi 等人基于 PSI-CA,APSI 和 PSI 协议构建了有效的基因数据隐私保护计算算法,并应用在亲子鉴定、个人医疗和基因兼容测试等场景上[22].

3) 在线广告转化率评估.在线广告作为企业营销策略的重要组成部分,蕴藏着巨大的商机.传统的评估广告影响力指标是广告点击率,但是随着人们对广告了解的深入,点击人数越来越少,点击率不能真正反映广告的效果.针对这个问题提出了广告转化率这个指标,指受网络广告影响而发生购买、注册或信息需求行为的浏览者占总广告点击人数的比例,用来反映网络广告对产品销售情况影响程度.然而广告浏览数据、交易完成数据分别掌握在广告商如谷歌、Facebook 手中和商家手中,这些数据都包含有顾客的隐私信息,不能在明文信息进行交集计算,因此,可以通过信用卡号、email 地址等身份表示信息进行 PSI 计算,在保护顾客隐私的同时实现对在线广告转化率的评估.

4) 私有联系人发现.很多手机社交类的 APP 要求访问新注册用户的手机通信录,通过对比通信录和 APP 服务器的注册用户,给新用户推荐共同使用该 APP 的好友.但是允许 APP 访问用户的手机通信录会将用户所有的联系人信息暴露给 APP 服务商,存在很大的隐私泄露问题.通过在 APP 手机端和 APP 服务器端运行 PSI 协议,计算手机通信录和服务器已注册用户集之间的交集,可以在不泄露用户隐私信息的条件下进行联系人及好友推荐[91].

5) 隐私保护的近邻检测.近邻检测是社交网络中基于位置服务的一种重要应用,但位置服务在给人们带来方便的同时也面临着严重的位置隐私泄露问题.隐私保护的近邻检测可以通过同态加密等技术实现对精准位置数据的密文计算,也可以通过 WiFi、蓝牙、GPS 等信号给位置打上的标签信息进行隐私集合交集计算,完成对近邻关系的评估[23].

5 总 结

随着信息化和网络化的不断发展,隐私保护成为一个越来越重要的课题,不仅关系到公民个人利益更关乎到国家的安全.安全多方计算作为隐私保护的一个重要密码学研究方法,也由最初的采用通用协议解决安全函数计算问题发展到设计特殊的针对具体问题的高效协议.PSI 作为安全多方计算的一个重要应用问题,近年来也得到快速发展.目前,为了适应大数据时代的计算需求,PSI 协议的发展历程主要由最初的基于公钥加密机制到基于混乱电路、不经意传输协议,以及新的计算模式——云计算环境下的协议设计.无论是传统的方法还是云辅助的 PSI 协议,对该问题的主要研究趋势是均衡安全性和高效性、可扩展性,研究思路主要是由最初的基于公钥加密机制到依赖更多的对称加密操作.

虽然对 PSI 协议的研究取得了一定的进展,但是现有协议普遍存在一些问题:1)在恶意模型下,为了验证输入数据的一致性等导致计算通信代价高,不具有实用性;2)大多数传统的高效的 PSI 协议主要依赖于备受争议的随机预言模型,随机预言模型的安全性证明并不能保证实际方案的安全性.

结合当前 PSI 问题的研究进展,未来对 PSI 协议研究的重点包括:1)新的不可信计算模式云计算环境下的 PSI 安全协议研究,随着更多公司和个人对云计算的使用,基于云计算服务的 PSI 协议将成为热点研究问题;2)在当前大数据背景下,已有的 PSI 协议尽管已经取得了很大的突破但仍有性能上的瓶颈,未来需要设计出更高效的协议来满足大数据隐私保护需求.

参 考 文 献

[1] Fang Binxing, Jia Yan, Li Aiping, et al. Privacy preservation in big data: A survey [J]. Big Data Research, 2017, 2(1): 1-18 (in Chinese)
(方滨兴, 贾焰, 李爱平, 等. 大数据隐私保护技术综述[J]. 大数据, 2017, 2(1): 1-18)

[2] Du Wenliang, Atallah M. Secure multi-party computation problems and their applications: A review and open problems [C] //Proc of the 2001 Workshop on New Security Paradigms. New York: ACM, 2001: 13-22

[3] Mu Bin. A survey on secure processing of similarity queries [OL]. (2014-05-01) [2017-09-06]. https://pdfs.semanticscholar.org/d602/831fc377fd9b84366c3e8a5ab736c9c3a5ef.pdf

[4] Erkin Z, Franz M, Guajardo J, et al. Privacy-preserving face recognition [G] //LNCS 5672: Proc of Int Symp on Privacy Enhancing Technologies. Berlin: Springer, 2009: 235-253

[5] Jagannathan G, Wright R N. Privacy-preserving distributed *k*-means clustering over arbitrarily partitioned data [C] // Proc of the 18th ACM SIGKDD Int Conf on Knowledge Discovery in Data Mining. New York: ACM, 2005: 593-599

[6] Bringer J, Chabanne H, Favre M, et al. GSHADE: Faster privacy-preserving distance computation and biometric identification [C] //Proc of the 2nd ACM Workshop on Information Hiding and Multimedia Security. New York: ACM, 2014: 187-198

[7] Yao A C-C. Protocols for secure computations [C] //Proc of the 23rd Annual Symp on Foundations of Computer Science (SFCS'82). Piscataway, NJ: IEEE, 1982: 160-164

[8] Goldreich O, Micali S, Wigderson A. How to play any mental game, or a completeness theorem for protocols with an honest majority [C] //Proc of the 19th Annual ACM STOC. New York: ACM, 1987: 218-229

[9] Goldreich O. Secure multi-party computation [J]. 1998 (2002-10-27) [2017-09-06]. http://www.wisdom.weizmann.ac.il/~oded/PSX/prot.pdf

[10] Goldwasser S. Multi party computations: Past and present [C] //Proc of the 16th Annual ACM Symp on Principles of Distributed Computing. New York: ACM, 1997: 1-6

[11] Kissner L, Song D. Privacy-preserving set operations [G] // LNCS 3621: Proc of Annual Int Cryptology Conf. Berlin: Springer, 2005: 241-257

[12] Sang Yingpeng, Shen Hong. Efficient and secure protocols for privacy-preserving set operations [J]. ACM Trans on Information and System Security, 2009, 13(1): 1-35

[13] Lindell Y, Pinkas B. Secure multiparty computation for privacy-preserving data mining [J]. Journal of Privacy and Confidentiality, 2009, 1(1): 59-98

[14] Agrawal R, Srikant R. Privacy-preserving data mining [C] // Proc of the 2000 ACM SIGMOD Int Conf on Management of Data Record. New York: ACM, 2000: 439-450

[15] Aggarwal C C, Philip S Y. A General Survey of Privacy-Preserving Data Mining Models and Algorithms [M]. Berlin: Springer, 2008: 11-52

[16] Pinkas B. Cryptographic techniques for privacy-preserving data mining [J]. ACM SIGKDD Explorations Newsletter, 2002, 4(2): 12-19

[17] Cao Ning, Wang Cong, Li Ming, et al. Privacy-preserving multi-keyword ranked search over encrypted cloud data [J]. IEEE Trans on Parallel and Distributed Systems, 2014, 25 (1): 222-233

[18] Wang Cong, Cao Ning, Li Jin, et al. Secure ranked keyword search over encrypted cloud data [C] //Proc of the 30th Int Conf on Distributed Computing Systems (ICDCS). Piscataway, NJ: IEEE, 2010: 253-262

[19] Song D X, Wagner D, Perrig A. Practical techniques for searches on encrypted data [C] //Proc of the 2000 IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2000: 44-55

[20] Atallah M, Du Wenliang. Secure multi-party computational geometry [G] //LNCS 2125: Proc of Workshop on Algorithms and Data Structures (WADS 2001). Berlin: Springer, 2001: 165-179

[21] Vaidya J, Clifton C. Privacy preserving association rule mining in vertically partitioned data [C] //Proc of the 8th ACM SIGKDD Int Conf on Knowledge Discovery and Ddata Mining. New York: ACM, 2002: 639-644

[22] Baldi P, Baronio R, De Cristofaro E, et al. Countering gattaca: Efficient and secure testing of fully-sequenced human genomes [C] //Proc of the 18th ACM Conf on Computer and Communications Security. New York: ACM, 2011: 691-702

[23] Narayanan A, Thiagarajan N, Lakhani M, et al. Location privacy via private proximity testing [C] //Proc of the Network and Distributed System Security Symp. Reston, VA: ISOC, 2011

[24] Mezzour G, Perrig A, Gligor V, et al. Privacy-preserving relationship path discovery in social networks [G] //LNCS 5888: Proc of Int Conf on Cryptology and Network Security. Berlin: Springer, 2009: 189-208

[25] Freedman M J, Nissim K, Pinkas B. Efficient private matching and set intersection [G] //LNCS 3027: Proc of Int Conf on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2004: 1-19

[26] Kamara S, Mohassel P, Raykova M, et al. Scaling private set intersection to billion-element sets [G] //LNCS 8437: Proc of Int Conf on Financial Cryptography and Data Security. Berlin: Springer, 2014: 195-215

[27] Goldreich O. Foundations of Cryptography Ⅱ Basic Applications [M]. Cambridge, UK: Cambridge University Press, 2004

[28] Katz J, Lindell Y. Introduction to Modern Cryptography [M]. New York: CRC Press, 2014

[29] Even S, Goldreich O, Lempel A. A randomized protocol for signing contracts [J]. Communications of the ACM, 1985, 28(6): 637-647

[30] Impagliazzo R, Rudich S. Limits on the provable consequences of one-way permutations [C] //Proc of the 21st Annual ACM Symp on Theory of Computing. New York: ACM, 1989: 44-61

[31] Beaver D. Correlated pseudorandomness and the complexity of private computations [C] //Proc of the 28th Annual ACM Symp on Theory of Computing. New York: ACM, 1996: 479-488

[32] Ishai Y, Kilian J, Nissim K, et al. Extending oblivious transfers efficiently [G] //LNCS 2729: Annual Int Cryptology Conf. Berlin: Springer, 2003: 145-161

[33] Kolesnikov V, Kumaresan R. Improved OT extension for transferring short secrets [G] //LNCS 8043: Advances in Cryptology (CRYPTO 2013). Berlin: Springer, 2013: 54-70

[34] Asharov G, Lindell Y, Schneider T, et al. More efficient oblivious transfer and extensions for faster secure computation [C] //Proc of the 2013 ACM SIGSAC Conf on Computer & Communications Security. New York: ACM, 2013: 535-548

[35] Yao A C C. How to generate and exchange secrets [C] // Proc of the 27th Annual Symp on Foundations of Computer Science. Piscataway, NJ: IEEE, 1986: 162-167

[36] Kamara S, Mohassel P, Riva B. Salus: A system for server-aided secure function evaluation [C] //Proc of the 2012 ACM Conf on Computer and Communications Security. New York: ACM, 2012: 797-808

[37] Jiang Han, Xu Qiuliang. Advances in key techniques of practical secure multi-party computation [J]. Journal of Computer Research and Development, 2015, 52(10): 2247-2257 (in Chinese)
(蒋瀚, 徐秋亮. 实用安全多方计算协议关键技术研究进展 [J]. 计算机研究与发展, 2015, 52(10): 2247-2257)

[38] Pinkas B, Schneider T, Smart N P, et al. Secure two-party computation is practical [G] //LNCS 5912: Proc of Int Conf on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2009: 250-267

[39] Naor M, Pinkas B, Sumner R. Privacy preserving auctions and mechanism design [C] //Proc of the 1st ACM Conf on Electronic Commerce. New York: ACM, 1999: 129-139

[40] Zahur S, Rosulek M, Evans D. Two halves make a whole [G] //LNCS 9057: Proc of Annual Int Conf on the Theory and Applications of Cryptographic Techniques (EUROCRYPT 2015). Berlin: Springer, 2015: 220-250

[41] Pinkas B. Fair secure two-party computation [G] //LNCS 2656: Proc of Int Conf on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2003: 87-105

[42] Malkhi D, Nisan N, Pinkas B, et al. Fairplay-secure two-party computation system [C] //Proc of the 13th Conf on USENIX Security Symp. Berkeley, CA: USENIX Association, 2004

[43] Huang Yan, Evans D, Katz J, et al. Faster secure two-party computation using garbled circuits [C] //Proc of the 20th Conf on USENIX Security Symp. Berkeley, CA: USENIX Association, 2011

[44] Liu Chang, Wang Xiao, Nayak K, et al. Oblivm: A programming framework for secure computation [C] //Proc of 2015 IEEE Symp on Security and Privacy (SP). Piscataway, NJ: IEEE, 2015: 359-376

[45] Burkhart M, Strasser M, Many D, et al. SEPIA: Privacy-preserving aggregation of multi-domain network events and statistics [C] //Proc of the 19th Conf on USENIX Security. Berkeley, CA: USENIX Association, 2010

[46] Damgård I, Geisler M, Krøigaard M, et al. Asynchronous multiparty computation: Theory and implementation [G] // LNCS 5443: Proc of Int Workshop on Public Key Cryptography. Berlin: Springer, 2009: 160-179

[47] Bogdanov D, Laur S, Willemson J. Sharemind: A framework for fast privacy-preserving computations [G] // LNCS 5283: European Symp on Research in Computer Security. Berlin: Springer, 2008: 192-206

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

[49] Blakley G R. Safeguarding cryptographic keys [C] //Proc of American Federation of Information Processing Socicties National Computer Conf. Wuhan: SCIRP, 1979: 313-317

[50] Naor M, Pinkas B. Oblivious transfer and polynomial evaluation [C] //Proc of the 21st Annual ACM Symp on Theory of Computing. New York: ACM, 1999: 245-254

[51] Paillier P. Public-key cryptosystems based on composite degree residuosity classes [G] //LNCS 1592: Proc of Int Conf on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 1999: 223-238

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

[53] Azar Y, Broder A Z, Karlin A R, et al. Balanced allocations [J]. SIAM Journal on Computing, 1999, 29(1): 180-200

[54] Freedman M J, Hazay C, Nissim K, et al. Efficient set intersection with simulation-based security [J]. Journal of Cryptology, 2016, 29(1): 115-155

[55] Pagh R, Rodler F F. Cuckoo hashing [C] //Proc of European Symp on Algorithms. Berlin: Springer, 2001: 121-133

[56] Dachman-Soled D, Malkin T, Raykova M, et al. Efficient robust private set intersection [G] //LNCS 5536: Proc of Int Conf on Applied Cryptography and Network Security. Berlin: Springer, 2009: 125-142

[57] Zhou Sufang, Li Shundong, Guo Yimin, et al. Efficient set intersection problem computation [OL]. [2017-06-01]. http://kns.cnki.net/KCMS/detail/11.1826.TP.20170526.1938.014.html (in Chinese)
(周素芳, 李顺东, 郭奕旻, 等. 保密集合相交问题的高效计算[OL]. [2017-06-01]. http://kns.cnki.net/KCMS/detail/11.1826.TP.20170526.1938.014.html)

[58] Chen Zhenhua, Li Shundong, Huang Qiong, et al. Protocols for secure computation of two set-relationships with the unencrypted method [OL]. [2017-06-01]. http://kns.cnki.net/KCMS/detail/11.2560.TP.20170331.2154.001.html (in Chinese)
(陈振华, 李顺东, 黄琼, 等. 非加密方法安全计算两种集合关系[OL]. [2017-06-01]. http://kns.cnki.net/KCMS/detail/11.2560.TP.20170331.2154.001.html)

[59] Hazay C, Nissim K. Efficient set operations in the presence of malicious adversaries [G] //LNCS 6056: Proc of Int Workshop on Public Key Cryptography. Berlin: Springer, 2010: 312-331

[60] Naor M, Reingold O. Number-theoretic constructions of efficient pseudo-random functions [J]. Journal of the ACM, 2004, 51(2): 231-262

[61] Hazay C, Lindell Y. Efficient protocols for set intersection and pattern matching with security against malicious and covert adversaries [G] //LNCS 4948: Theory of Cryptography Conf. Berlin: Springer, 2008: 155-175

[62] Jarecki S, Liu Xiaomin. Efficient oblivious pseudorandom function with applications to adaptive ot and secure computation of set intersection [G] //LNCS 5444: Proc of Theory of Cryptography Conf (TCC 2009). Berlin: Springer, 2009: 577-594

[63] Dodis Y, Yampolskiy A. A verifiable random function with short proofs and keys [G] //LNCS 3386: Proc of Int Workshop on Public Key Cryptography. Berlin: Springer, 2005: 416-431

[64] Camenisch J, Shoup V. Practical verifiable encryption and decryption of discrete logarithms [G] //LNCS 2729: Proc of Annual Int Cryptology Conf (CRYPTO 2003). Berlin: Springer, 2003: 126-144

[65] Jarecki S, Liu Xiaomin. Fast secure computation of set intersection [G] //LNCS 6280: Proc of Int Conf on Security and Cryptography for Networks. Berlin: Springer, 2010: 418-435

[66] Bellare M, Namprempre C, Pointcheval D, et al. The one-more-RSA-inversion problems and the security of Chaum's blind signature scheme [J]. Journal of Cryptology, 2003, 16 (3): 185-215

[67] De Cristofaro E, Tsudik G. Practical private set intersection protocols with linear complexity [G] //LNCS 6052: Proc of Int Conf on Financial Cryptography and Data Security. Berlin: Springer, 2010: 143-159

[68] De Cristofaro E, Kim J, Tsudik G. Linear-complexity private set intersection protocols secure in malicious model [G] //LNCS 6477: Proc of Int Conf on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2010: 213-231

[69] De Cristofaro E, Tsudik G. Experimenting with fast private set intersection [G] //LNCS 7344: Proc of Int Conf on Trust and Trustworthy Computing. Berlin: Springer, 2012: 55-73

[70] Ateniese G, De Cristofaro E, Tsudik G. (If) size matters: Size-hiding private set intersection [G] //LNCS 6571: Proc of Int Workshop on Public Key Cryptography. Berlin: Springer, 2011: 156-173

[71] Huang Y, Evans D, Katz J. Private set intersection: Are garbled circuits better than custom protocols? [C] //Proc of the Network and Distributed System Security Symp. Reston, VA: ISOC, 2012

[72] Kolesnikov V, Schneider T. Improved garbled circuit: Free XOR gates and applications [G] //LNCS 5126: Proc of Int Colloquium on Automata, Languages, and Programming. Berlin: Springer, 2008: 486-498

[73] Pinkas B, Schneider T, Zohner M. Faster private set intersection based on OT extension [C] //Proc of the 23rd USENIX Security Symp. Berkeley, CA: USENIX Association, 2014: 797-812

[74] Pinkas B, Schneider T, Zohner M. Scalable private set intersection based on OT extension [OL]. [2017-06-01]. https://eprint.iacr.org/2016/930.pdf

[75] Dong Changyu, Chen Liqun, Wen Zikai. When private set intersection meets big data: An efficient and scalable protocol [C] //Proc of the 2013 ACM SIGSAC Conf on Computer & Communications Security. New York: ACM, 2013: 789-800

[76] Bloom B H. Space/time trade-offs in Hash coding with allowable errors [J]. Communications of the ACM, 1970, 13 (7): 422-426

[77] Rindal P, Rosulek M. Improved private set intersection against malicious adversaries [G] //LNCS 10210: Proc of Annual Int Conf on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2017: 235-259

[78] Pinkas B, Schneider T, Segev G, et al. Phasing: Private set intersection using permutation-based hashing [C] //Proc of the 24th USENIX Security Symp. Berkeley, CA: USENIX Association, 2015: 515-530

[79] Arbitman Y, Naor M, Segev G. Backyard cuckoo hashing: Constant worst-case operations with a succinct representation [C] //Proc of the 51st Annual IEEE Symp on Foundations of Computer Science (FOCS). Piscataway, NJ: IEEE, 2010: 787-796

[80] Kolesnikov V, Kumaresan R, Rosulek M, et al. Efficient batched oblivious PRF with applications to private set intersection [C] //Proc of the 2016 ACM SIGSAC Conf on Computer and Communications Security. New York: ACM, 2016: 818-829

[81] Jiang Han, Xu Qiuliang. Secure Multi-party computation in cloud computing [J]. Journal of Computer Research and Development, 2016, 53(10): 2152-2162 (in Chinese)
(蒋瀚, 徐秋亮. 基于云计算服务的安全多方计算[J]. 计算机研究与发展, 2016, 53(10): 2152-2162)

[82] Kerschbaum F. Collusion-resistant outsourcing of private set intersection [C] //Proc of the 27th Annual ACM Symp on Applied Computing. New York: ACM, 2012: 1451-1456

[83] Kerschbaum F. Outsourced private set intersection using homomorphic encryption [C] //Proc of the 7th ACM Symp on Information, Computer and Communications Security. New York: ACM, 2012: 85-86

[84] Sander T, Young A, Yung M. Non-interactive crypto-computing for NC/sup 1 [C] //Proc of the 40th Annual Symp on Foundations of Computer Science. Piscataway, NJ: IEEE, 1999: 554-566

[85] Liu Fang, Ng W K, Zhang Wei, et al. Encrypted set intersection protocol for outsourced datasets [C] //Proc of 2014 IEEE Int Conf on Cloud Engineering (IC2E). Piscataway, NJ: IEEE, 2014: 135-140

[86] Zheng Qingji, Xu Shouhuai. Verifiable delegated set intersection operations on outsourced encrypted data [C] // Proc of 2015 IEEE Int Conf on Cloud Engineering (IC2E). Piscataway, NJ: IEEE, 2015: 175-184

[87] Abadi A, Terzis S, Dong Changyu. O-PSI: Delegated private set intersection on outsourced datasets [C] //Proc of IFIP Int Information Security Conf. Berlin: Springer, 2015: 3-17

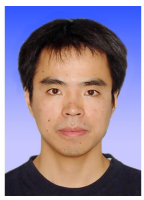
[88] Abadi A, Terzis S, Dong Changyu. VD-PSI: Verifiable delegated private set intersection on outsourced private datasets [G] //LNCS 9603: Proc of Int Conf on Financial Cryptography and Data Security. Berlin: Springer, 2016: 149-168

[89] Li Shundong, Zhou Sufang, Guo Yimin, et al. Secure set computing in cloud environment [J]. Journal of Software, 2016, 27(6): 1549-1565 (in Chinese)
(李顺东, 周素芳, 郭奕旻, 等. 云环境下集合隐私计算[J]. 软件学报, 2016, 27(6): 1549-1565)

[90] Blundo C, De Cristofaro E, Gasti P. EsPRESSO: Efficient privacy-preserving evaluation of sample set similarity [J]. Journal of Computer Security, 2014, 22(3): 355-381

[91] Huang Yan, Chapman P, Evans D. Privacy-preserving applications on smartphones [C] //Proc of the 6th USENIX Conf on Hot Topic in Security. Berkeley, CA: USENIX Association, 2011

Shen Liyan, born in 1992. PhD candidate. Her main research interests include privacy preserving, information security.



Chen Xiaojun, born in 1979. PhD. Member of CCF. His main research interests include big data, privacy preserving, data security.

Shi Jinqiao, born in 1978. PhD, associate professor. Member of CCF. His main research interests include network anonymous communication and network threats detection.

Hu Lanlan, born in 1979. PhD, research assistant. Her main research interests include big data, privacy preserving and information security.