计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2018.20180418
55(10):2125-2133,2018

# 一个高效安全三方带通配符模式匹配协议

魏晓超　郑志华　王　皓
(山东师范大学信息科学与工程学院　济南　250358)
(wxc@sdnu.edu.cn)

# An Efficient and Secure Three-Party Wildcard Pattern Matching Protocol

Wei Xiaochao, Zheng Zhihua, and Wang Hao
(*School of Information Science and Engineering, Shandong Normal University, Jinan* 250358)

**Abstract**　Secure multiparty computation is an important technique used to achieve distributed security, which mainly considers the cooperative computing between many distinct participants meanwhile guaranteeing the privacy of the input information. Pattern matching has wide application in information retrieval, bioengineering and facial recognition. How to protect the privacy of the retrieval pattern and result when achieving the matching function has attracted more and more attention of the researchers. As a variant of pattern matching, wildcard pattern matching enables the existence of wildcard in the retrieval pattern, such that batch retrieval can be achieved for information with same characteristics. Traditional secure wildcard pattern matching involves only two parties, which are database and user. However, in view of the development of data sharing technique, the above model cannot describe many new application scenarios. In this paper, we firstly research secure three-party wildcard pattern matching for some specific applications. At first, we propose a formal description of a concrete secure three-party wildcard pattern matching functionality and analyze its function. Then, we construct a protocol using secret sharing and outsourced oblivious transfer (OOT) in semi-honest model. Using oblivious transfer extension technique, our protocol requires only 3 rounds, and the computation and communication complexity is respectively $O(k)$ and $O(nm)$, where $n$ and $m$ are the lengths of the data providers, and $k$ is the basic number of OT extension which is much less than $nm$.

**Key words**　secure multiparty computation; wildcard pattern matching; outsourced oblivious transfer (OT) protocol; secret sharing; oblivious transfer (OT) extension

**摘　要**　安全多方计算(secure multiparty computation, SMPC)是实现分布式计算安全的重要技术,其主要考虑在多个相互独立的实体之间协同完成某项任务的计算,同时要实现输入信息的隐私保护.模式匹配在信息检索、生物工程、人脸识别等领域有着广泛应用,在实现匹配功能的同时保证查询模式及结果的隐私是当下研究的重点.带通配符模式匹配是模式匹配的一种类型,其允许查询模式中可以存在某些通配符信息,因此能够实现某一类信息的批量查询.传统的安全带通配符模式匹配协议中主要涉及

**收稿日期**:2018-06-07;**修回日期**:2018-08-03
**基金项目**:中国博士后科学基金项目(2018M632712);国家自然科学基金青年科学基金项目(61602287);山东省重点研发计划(2018GGX101037);山东省科技重大创新工程项目(2018CXGC0702)
This work was supported by the China Postdoctoral Science Foundation (2018M632712), the National Natural Science Foundation of China for Young Scientists (61602287), the Key Research and Development Plan of Shandong Province (2018GGX101037), and the Major Innovation Project of Science and Technology in Shandong Province (2018CXGC0702).
**通信作者**:郑志华(zhengzhihua@sdnu.edu.cn)

数据库方和查询方 2 个实体,鉴于当下数据共享技术的发展,这种模型难以刻画更多的应用场景.以实际应用出发,首次在三方场景下研究安全带通配符模式匹配协议的构造.首先考虑一个具体的安全三方带通配符模式匹配功能函数,并给出其形式化描述和功能性分析;然后,基于秘密分享(secret sharing)和外包茫然传输协议(outsourced oblivious transfer, OOT)在半诚实敌手模型下给出协议构造,并通过茫然传输扩展(oblivious transfer extension)技术提高协议效率,协议仅需要 3 轮交互,且计算和通信复杂度为 $O(k)$ 和 $O(nm)$,其中 $n$ 和 $m$ 是 2 个数据提供方的输入长度,$k$ 是实现 OT 扩展协议的基数,其值远小于 $nm$.

**关键词** 安全多方计算;带通配符模式匹配;外包 OT 协议;秘密分享;OT 扩展

**中图法分类号** TP301

分布式计算场景中存在多个独立的参与方,他们通过网络相互连接并希望使用自己的私有数据协同完成某项任务.随着隐私泄露事件的不断发生以及人们隐私保护意识的不断增强,如何在分布式计算过程中实现隐私数据的保护成为当下分布式计算安全的重要研究内容.在密码学研究中,安全多方计算(secure multiparty computationm, SMPC)[1-3]是实现分布式场景中数据隐私保护的重要技术,其形式化的安全模型、安全性定义、可证明安全技术以及丰富的基础工具为实现分布式安全提供强大的理论和技术支持.此外,随着云计算、大数据等新兴技术的发展,分布式计算的场景和模式也随之而发生变化,急切需要新的解决方式,如云辅助安全多方计算[4]等来应对新模型、新问题的挑战.

模式匹配是计算机科学领域一个非常基本的问题,在信息检索、生物信息学等领域应用广泛.它本质上是一个搜索问题,即查找一个给定的模式 $p\in[\Sigma]^m$ 在文本 $t\in[\Sigma]^n$ 中出现的位置,其中 $\Sigma$ 是一个字母表集合,譬如假设 $p$ 和 $t$ 均为二进制表示,则 $\Sigma=\{0,1\}$.我们将上述问题成为精确模式匹配问题,即寻找整个模式串出现的位置.除此之外,模式匹配问题还有诸多变种,譬如近似模式匹配、带通配符的模式匹配等.近似匹配广泛出现在人类基因测序和拼接,由于核苷酸多态性或排序偏差,不同基序列可以组合成相同的基因,只要满足他们的差异在某个阈值以内,因此只需要近似相等就可以识别出一个人.此外,当我们想查询某一类带有共同特性的数据时,可以使用通配符来表示数据中不同的信息,只保留相同的数据信息,进而使用带通配符的模式匹配方案来完成批量数据的搜索,提高搜索的效率.

不难看出模式匹配及其变种问题应用极广,但是在当前分布式计算场景下,数据的隐私性威胁日益严峻,人们不希望将自己查询的信息泄露给其他个人或机构,因此需要我们在完成模式查询的同时,保证数据隐私不被泄露.在上述应用场景中涉及到至少 2 个参与实体,即文本数据持有者和模式持有者,因此我们从协议层面考虑可以使用安全两方计算对模式匹配协议进行形式化描述.具体地,安全模式匹配协议中包含 2 个参与方——数据库和用户,其中数据库持有一个文本 $t\in[\Sigma]^n$,用户持有一个模式 $p\in[\Sigma]^m$.协议希望确定模式 $p$ 出现在文本 $t$ 中的位置,同时要满足 2 个安全属性:1)模式 $p$ 要对数据库方保密;2)用户除了 $p$ 出现的位置之外不知道文本 $t$ 中的其他任何信息.

## 1 相关工作和主要工作

### 1.1 相关工作

针对安全模式匹配协议的研究最早可以追溯到 2007 年 Troncoso-Pastoriza 等人[5]的工作.他们主要针对精确匹配问题,将其转化为茫然自动机求值问题.之后又有诸多文献对此进行效率和安全性的改进[6-9].此外,还有学者使用茫然伪随机计算[10-11]和 Yao 混乱电路[12]等不同技术分别给出不同协议构造.近些年该领域的发展主要表现在安全模式匹配的功能性扩展上,包括近似模式匹配、带通配符模式匹配和外包模式匹配等.

1) 近似模式匹配

近似模式匹配不像精确匹配要求的那么严格,只要模式和文本中子串的不同信息满足一个给定的阈值,就意味着匹配成功.鉴于现实生活中很多数据的采集都不是绝对一致的,因此近似匹配在基因比对、人脸识别等领域应用广泛.2009 年 Jarrous 和 Pinkas[13]首次安全计算 2 个等长字符串的 Hamming 距离,并将其应用到近似模式匹配协议中去.该协议主要基于茫然多项式计算(oblivious polynomial

evaluation)技术,协议通信和计算量均为 $O(nm)$. 之后,Vergnaud[14] 使用快速傅里叶变换(fast Fourier transform, FFT)技术分别构造了可以抵抗半诚实和恶意敌手近似模式匹配协议. Hazay 和 Toft[15-16] 使用编辑距离(edit distance)来衡量 2 个字符串的相似度,基于具有加同态性质的加密方案,并结合知识的零知识证明协议,实现安全计算 2 个等长字符串的编辑距离. 编辑距离指的是将一个字符串变成另一个字符串所需要的操作步数,该协议的核心就是如何针对加密数据安全计算编辑距离. 他们所构造的协议通信和计算复杂度分别为 $O(n\tau)$ 和 $O(nm)$,其中 $\tau$ 是衡量"近似"特性的阈值. Wei 等人[17] 使用门限秘密分享和 OT 协议(oblivious transfer protocol)给出新的协议构造. 其主要思想是用秘密分享份额来表示文本 $t$ 的每一个位信息,并通过 OT 扩展协议完成对份额的传输,最终将匹配问题等价转化为秘密能否重构. 协议的通信复杂度和计算复杂度分别为 $O(nm)$ 和 $O(k)$,其中 $k$ 是 OT 扩展协议中的基础 OT 数目,其值远小于 $nm$.

2) 带通配符模式匹配

通配符意味着其可以被字符集合中的任意字符所代替,在模式中设置通配符位可以起到批量查找的作用. 譬如,学校要查找所有 2017 级的学生信息. 只需将查询学号的前 4 位设置成"2017",后面设置为通配符就可以了,即形如"2017****". 针对安全带通配符模式匹配的研究是近几年的研究热点,其核心问题是如何实现通配符信息与任意字符串的成功匹配. Hazay 和 Toft[15-16] 通过 2 个参与方秘密地将通配符作相同的替换处理,从而将带通配符匹配转化为精确匹配. 使用加同态加密方案,其协议的通信复杂度和计算复杂度分别为 $O(n+m)$ 和 $O(nm)$. Baron 等人[18] 针对非二进制字母表的一般化带通配符模式匹配协议做了研究,其主要思想是基于线性代数公式及加同态加密方案,其通信和计算复杂度与文献[15-16]相同. 此外,文献[19-20]基于对称类同态加密技术构造了安全带通配符模式匹配协议. 他们构造了一种数据包装技术,可以针对加密数据高效计算多个 Hamming 距离. 该协议可以适用于非二进制数据,经过对真实 DNA 序列的实验测试,其可以实现一秒钟查询长度为 16 500 的基因序列. 基于此,Saha 等人[21] 考虑如何在外包数据中实现带通配符模式匹配. 通过改变上述包装技术使其能够适用于外包场景,协议效率有了 $k$ 倍的提升. 2017 年,Kolesnikov 等人[22] 基于 OT 扩展协议构造了一个高效的安全带通配符模式匹配协议. 他们的协议首先使用 OT 协议使得 2 个参与方能共同计算出一个随机值,然后调用安全字符串相等性测试(secure string equality test)协议来确定计算出来的值是否相等,以此来确定模式是否匹配. 协议的计算和通信复杂度分别为 $O(k)$ 和 $O(nm)$,缺点是要额外调用字符串相等性测试协议. 2018 年,Darivandpour 等人[23] 使用轻量级密码给出更为高效的协议构造. 他们的方案可以针对所有的字符集并支持任意规模的输入,但是需要一个离线服务器(云服务器)帮助参与方生成一些协议中使用到的随机数.

3) 外包模式匹配

云计算的发展使得外包计算成为一种潮流和趋势,因此外包模式匹配也成为大家研究的热点. 在外包模式匹配场景中,用户将自己数据加密并发送至云服务器上,在搜索某个模式是否存在时,委托云服务器代劳并将匹配结果秘密返回给用户. 就隐私性而言,用户的数据、查询的模式以及匹配结果都要对服务器保密. 2013 年,Faust 等人[24] 首次将模式匹配问题等价转化为子集和问题,即如果一个值可以被表示为某一个子集中元素的和,则意味着模式存在于文本中. 他们的协议可以抵抗半诚实和恶意敌手,但是需要随机预言机的存在. 与之类似的,Chase 和 Shen[25] 提出子串对称可搜索加密的概念,即使用对称可搜索加密的思想实现外包加密文本中子串的匹配问题. 使用后缀树技术,他们给出的方案相对高效,且可以抵抗恶意敌手攻击. 后续又有诸多工作[26-28] 针对这一全新的功能函数给出更优的实现. 此外,Li 等人[29] 在外包场景下针对隐私保护的模式匹配问题给出了一个安全协议构造,他们的协议主要使用带聚合的加密方案对数据加密,匹配阶段的计算量为 $O(nm)$. 文献[30]首次构造了一个隐私保护的多模式匹配协议,其可以针对加密的模式集合执行秘密检索. 该协议主要基于 Aho-Corasick 自动机,其计算复杂度在最坏情况下为 $O(n+d)$,其中 $n$ 是本文的长度,$d$ 是多个模式中最长模式的长度.

### 1.2 本文的主要工作

本文首次在三方场景下考虑安全带通配符模式匹配协议. 之前的带通配符模式匹配协议往往仅涉及 2 个参与方,即模式持有者查询其模式信息是否出现在某一数据库文件中. 然而,在当今信息化时代,信息的有偿使用也日益成为一种消费形式,即用户希望通过支付方式来获得他人信息的使用权,但前提是要保证信息提供者的隐私性,同时也要保证

用户的结果是保密的. 因此,我们考虑如下带通配符模式匹配场景,如图 1 所示:

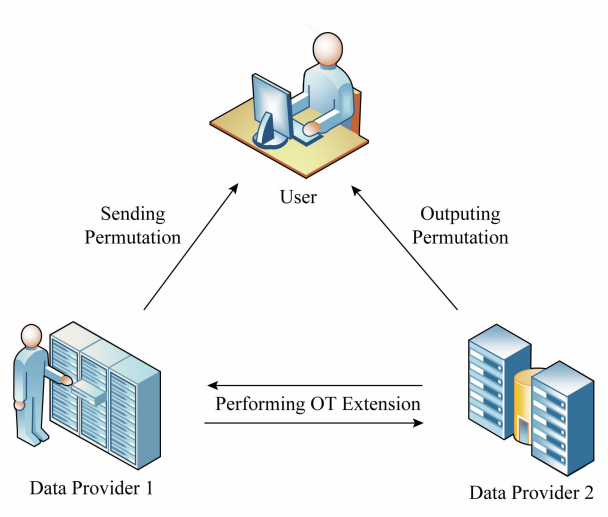


Fig. 1 The system model

图 1 系统模型

图 1 中 2 个数据提供方分别提供数据库文件和带通配符模式信息,用户希望了解模式是否存在于数据库文件中,以此为后续工作奠定基础. 很显然,数据提供方不希望将自己的数据泄露给其他人,而且用户也希望自己得到的结果是保密的. 现实生活中有诸多这样的例子,譬如在购买保险时,保险公司通常会根据投保人是否有某些疾病来设定保险的价格,但是个人的疾病信息是及其隐私的,每个人都不想让保险机构知道这些信息. 这样我们就可以使用图 1 的模型,设定信息提供方分别为医院和个人,其中医院有某些疾病的基因库,个人则提供自己的健康信息. 模型中的用户即为保险公司,它希望知道投保人是否得某种病,即投保人的健康数据是否能与医院的某种疾病匹配成功. 这样一来,在保证隐私的前提下,保险公司就可以完成对投保人疾病信息的核实和检验,从而为后期保费的设定提供依据.

本文的贡献主要包含以下 3 个方面:

1) 首次在三方场景下根据具体的应用需求考虑一种三方带通配符模式匹配协议,从而扩展了带通配符模式协议在当前数据时代的全新应用. 同时,我们对该协议的功能函数给出了形式化的描述和论述;

2) 针对协议功能函数的特点,本文基于秘密分享和外包 OT 协议给出了一个半诚实安全的协议构造. 带通配符模式匹配的难点在于如何使得通配符位可以匹配任意值,因此就需要通过一定的方法将一个带通配符的模式转换成正常的模式(即去除通配符),进而进行精确的模式匹配操作. 鉴于此,我们首次使用秘密分享份额表示法,针对通配符位和非通配符位上的位做 2 种不同的表示. 其中非通配符位按照其真实比特值构造一对数值,满足与真实比特对应位置的数值是合法的秘密分享份额,另一个值为随机数;而针对通配符位,将 2 个数值均设置位合法的秘密分享份额. 通过这种表示方法,并结合外包 OT 协议的使用,使得无论欲匹配的文档中与模式中通配符位对应的位信息是什么,都能得到合法的秘密分享份额,这样就在不泄露通配符位置的情况下完成了通配符的匹配.

3) 本文所构造的安全三方带通配符模式匹配协议需要 3 轮交互. 通过使用 OT 扩展技术,协议中参与方的计算复杂度可以从 $O(nm)$ 降低至 $O(k)$,其中 $n$ 和 $m$ 分别是数据提供方的输入长度,$k$ 是协议所需基础 OT 协议的个数,其值远小于 $nm$,譬如,$nm=10^6$,则 $k$ 仅为 128 即可满足.

## 2 基本知识和安全定义

### 2.1 秘密分享协议

秘密分享(secret sharing)协议最早是由 Shamir[31] 和 Blakley[32] 在 1979 年提出的,并分别基于 Lagrange 插值定义和射影几何理论给出具体构造,其在门限加密、安全多方计算领域有着广泛应用. 在秘密分享协议中考虑将一个秘密值 $s$ 分享至 $n$ 个参与方,其中每个参与方得到的分享值称为分享份额. 协议要求只有当 $n$ 个参与方中的某些授权子集提供正确的分享份额时,才能够成功恢复出正确的秘密值,而那些不在授权子集中的参与方不可能重构出秘密. 如果将授权子集中参与方的个数 $t$ 称之为门限值,则相应的秘密分享协议也被称之为 $(t,n)$-门限秘密分享协议,其要求只有当共享份额的数目至少为 $t$ 个时才能够恢复出正确秘密值.

### 2.2 外包茫然传输协议

外包茫然传输协议(outsourced oblivious transfer, OOT)[33] 是在传统茫然传输协议[34] 的基础上,增加一个云服务器作为协议第三方. 在外包 OT 协议中,不再是接收方得到其选择的输出信息,而是由云服务器最终获得这些值. 换言之,传统 OT 协议中的发送方和接收方依然输入相应的信息,只不过输出值最终发送给云服务器. 与此同时,仍需要保证每个参与方的输入信息向其他参与方保密.

OT 扩展协议(oblivious transfer extension)[35]

可以用很少的(如128个)基础OT加上廉价的对称密钥操作实现大规模(如$10^6$)OT的效果。因此OT扩展技术被广泛应用于需要大量使用OT协议的场景中,从而提高协议的整体效率.同理,在外包OT场景中,仍然可以使用OT扩展技术来达到相同的目的,其基本思想是在发送方和接收方之间执行OT扩展协议,然后再将得到的值置换之后发送给云服务器.下面我们给出外包OT扩展协议的具体描述:

**协议1.** 外包茫然传输扩展协议.

输入:Alice输入长度为$n$的的字符串$ea$,Bob输入数对$(x_{0,j},x_{1,j})$,其中$j=1,2,\cdots,n$,云服务器无输入;

输出:云服务器输出Bob选择的相应值.

步骤1. 初始化:Alice生成一个规模为$n\times t$的随机矩阵$\boldsymbol{T}$,Bob生成一个长度为$t$的随机串$s$.

步骤2. 基础OT:Alice和Bob执行$t$个1-out-of-2茫然传输协议,其中Alice输入$(\boldsymbol{T}^i,\boldsymbol{T}^i\oplus ea)$,Bob输入选择比特$s$,$\boldsymbol{T}^i$表示矩阵$\boldsymbol{T}$的第$i$列.Bob将接收到的列信息设置为矩阵$\boldsymbol{Q}$.

步骤3. 选择置换:Alice生成一个长度为$n$的随机串$p$,并将其发送给Bob.

步骤4. 加密输出:Bob将加密后的输出对设置为$(y_{0,j},y_{1,j})$,其中,$y_{b,j}=x_{b,j}\oplus H_1(j,\boldsymbol{Q}_j\oplus(b\cdot s))$,$\boldsymbol{Q}_j$表示矩阵$\boldsymbol{Q}$的第$j$行.

步骤5. 输出置换:Bob将加密后的输出对置换为$y_{0\oplus p_j,j},y_{1\oplus p_j,j}$,然后将最后的数对集合$Y$发送给云服务器.

步骤6. 解密输出:Alice将$h=ea\oplus p$和$\boldsymbol{T}$发送给云,云服务器计算出$z_j=y_{h_j,j}\oplus H_1(j,\boldsymbol{T}_j)$,其中$j=1,2,\cdots,n$,$\boldsymbol{T}_j$表示矩阵$\boldsymbol{T}$的第$j$行.

### 2.3 计算不可区分性

假设2个分布总体$X=\{X(a,n)\}_{a\in\{0,1\}^*;n\in\mathbf{N}}$和$Y=\{Y(a,n)\}_{a\in\{0,1\}^*;n\in\mathbf{N}}$是计算不可区分的,表示为$X\equiv Y$,则对每个非均匀多项式时间算法$D$,总存在一个可忽略函数$\mu(\cdot)$,对于每个$a\in\{0,1\}^*$和每个$n\in\mathbb{N}$,有不等式成立:

$$|Pr[D(X(a,n)=1)]-Pr[D(Y(a,n)=1)]|\leqslant\mu(n).$$

### 2.4 安全性定义

本文主要考虑半诚实敌手,即参与方严格按照协议规定执行,但是期望从其视图中获取其他参与方的输入信息.需要说明的是,我们假设诚实方大多数情况,即在三方场景中至多有一个参与方被腐化.我们使用文献[36]中的安全性定义:

**定义1.** 令$f:\{0,1\}^*\times\{0,1\}^*\times\{0,1\}^*\to\{0,1\}^*\times\{0,1\}^*\times\{0,1\}^*$是一个三方功能函数,$\pi$是一个真实的三方协议.若协议$\pi$在半诚实敌手模型下安全计算$f$,必须满足针对现实模型中的每个非均匀概率多项式时间敌手$\mathcal{A}$,总存在一个理想模型中的非均匀概率多项式时间敌手$\mathcal{S}$,对于每个$i\in\{1,2,3\}$,

$$\{IDEAL_{f,\mathcal{S}(z),i}(x,y,z,n,s)\}\equiv\{REAL_{\pi,\mathcal{A}(z),i}(x,y,z,n,s)\},$$

其中,$x,y,z\in\{0,1\}^*$满足$|x|=|y|$,且$n,s\in\mathbb{N}$.

## 3 安全三方带通配符模式匹配协议

### 3.1 三方带通配符模式匹配功能函数

本文所考虑的三方带通配符模式匹配功能函数$\mathcal{F}_{\mathrm{TWPM}}$中主要涉及3个参与方$P_1$,$P_2$和$P_3$,其中$P_1$,$P_2$是传统两方带通配符模式匹配协议中的文本提供方和模式提供方,而$P_3$是我们附加的一个参与方,其希望使用其他2个参与方的数据来实现自己的意图(如统计信息等).下面给出功能函数$\mathcal{F}_{\mathrm{TWPM}}$的形式化描述.

输入:

1) $P_1$输入长度为$n$的二进制字符串$t$及整数$m$;

2) $P_2$输入长度为$m$的带通配符二进制字符串$p$以及整数$n$,其中通配符表示为$*$,且其个数为$\tau$;

3) $P_3$无输入.

输出:

1) $P_3$确定模式$p$是否在$t$中出现,并输出出现的次数;

2) $P_1$,$P_2$无输出.

需要强调的是,在利用数据的同时,$P_1$和$P_2$并不希望将自己的输入泄露给$P_3$,同时$P_3$也不希望自己的查找结果让别人知道,因此功能函数的安全性要保证这2点.

### 3.2 安全三方带通配符模式匹配协议构造

在本节中,我们在半诚实敌手模型下构造一个安全三方带通配符模式匹配协议.协议具体表述如下:

**协议2.** 三方带通配符模式匹配协议$\pi_{\mathrm{TWPM}}$.

输入:$P_1$输入一个长度为$n$的二进制字符串$t$和一个整数$m$,$P_2$输入一个长度为$m$的带通配符(个数为$\tau$)的二进制字符串(模式)$p$和一个整数$n$,$P_3$没有输入;

输出:$P_3$输出$p$是否在$t$中出现以及出现次数.

步骤 1. 参与方 $P_2$ 使用秘密分享方案表示其输入,即带通配符的模式 $p$. 具体地,针对 $p$ 的每一个位,使用一组数对来表示. 对于非通配符位(即 0/1 比特),与位对应位置的数值是一个合法的秘密分享份额,另一个数值为随机数;对于通配符位上的比特值,2 个数值均为合法的秘密分享份额. 假设针对某一个秘密 $s$,使用一个秘密分享方案来生成秘密分享份额,其中该秘密分享方案需要满足至少需要 $m$ 个共享份额才能恢复出秘密. 具体表示方法如下:

① 针对非通配符位,使用有序数对$(s_i^0, s_i^1)$来表示 $p$ 的第 $i$ 位信息,其中 $i$ 表示该比特的位置下标. 假设 $\sigma$ 表示该比特的值,以上数对需要满足 $s_i^\sigma$ 是针对秘密 $s$ 的合法分享份额,而 $s_i^{1-\sigma}$ 是一个随机值;

② 针对通配符位,同样使用有序数对$(s_i^0, s_i^1)$来表示 $p$ 的第 $i$ 位信息,其中 $i$ 表示该位的位置下标. 此时需要满足 $s_i^0$ 和 $s_i^1$ 均是针对秘密 $s$ 的合法分享份额.

鉴于 $P_1$ 的输入 $t$ 中有 $n-m+1$ 个长度为 $m$ 的子串,因此参与方 $P_2$ 需要针对每一个子串使用不同的秘密值来表示其自己的输入,即 $P_2$ 要生成如下数值:

$$\begin{pmatrix} (s_{1,1}^0, s_{1,1}^1) & (s_{1,2}^0, s_{1,2}^1) & \cdots & (s_{1,m}^0, s_{1,m}^1) \\ (s_{2,1}^0, s_{2,1}^1) & (s_{2,2}^0, s_{2,2}^1) & \cdots & (s_{2,m}^0, s_{2,m}^1) \\ \vdots & \vdots & & \vdots \\ (s_{n-m+1,1}^0, s_{n-m+1,1}^1) & (s_{n-m+1,2}^0, s_{n-m+1,2}^1) & \cdots & (s_{n-m+1,m}^0, s_{n-m+1,m}^1) \end{pmatrix}$$

这些数值作为下一步茫然传输协议中发送方的输入信息.

步骤 2. 参与方 $P_1$,$P_2$ 和 $P_3$ 共同运行一个外包茫然传输协议(OOT),其中 $P_1$ 代表接收方 $R$,$P_2$ 代表发送方 $S$,$P_3$ 代表最终获得输出的服务器方. 考虑 OOT 协议中的输入信息,作为接收方的 $P_1$ 输入字符串 $t$,而作为发送方的 $P_2$ 输入步骤 1 中生成的所有数值(除此之外还需将生成秘密份额所使用的所有 $n-m+1$ 个秘密值按照顺序发送给 $P_3$),参与方 $P_3$ 无输入. 执行完 OOT 协议之后,参与方 $P_3$ 会得到所有的输出,即根据 $P_1$ 的输入从 $P_2$ 生成的所有数对中得到的与之匹配的值.

步骤 3. 参与方 $P_3$ 根据其从 OOT 协议中得到的输出值分别执行秘密重构操作,并比较重构出的秘密值和接收到的秘密值,最终确定模式 $p$ 是否在 $t$ 中有出现,具体地:

① 若重构出的数值中存在与接收到的秘密值相等的,则输出 1,表示匹配成功,并输出相等的个数;

② 否则,输出 0,表示不存在匹配成功的值.

### 3.3 正确性分析

在证明协议安全性之前,我们首先分析一下协议的正确性,即在运行完上述协议之后,参与方 $P_3$ 最终会得到正确的结果. 我们从 2 个方面说明其正确性:

1) 若匹配成功,则意味着 $t$ 中至少有一个长度为 $m$ 的子串与模式 $p$ 匹配,这就表明模式 $p$ 中非通配符位上的所有值与该子串中相应位置上的值均相等. 因此,在 OOT 协议中,使用该子串作为接收方输入时,$P_3$ 会接收到所有合法的秘密分享份额,其中非通配符位是因为值相同所致,而通配符位总是会得到合法份额. 最终,只用这些合法的共享份额,$P_3$ 能够重构出相同的秘密值,并确定存在匹配成功的子串信息.

2) 若无成功匹配时,则意味着 $t$ 中所有长度为 $m$ 的子串均与模式 $p$ 不匹配. 以某一个子串为例,这表明模式中的非通配符位中至少有一位的值与该子串中相应位置的值不等. 因此,在 OOT 协议中,在这一位置上 $P_3$ 得到的值是一个随机数,而非一个合法的秘密分享份额. 根据所选秘密分享方案的性质,合法共享份额的数量少于 $m$,这就使得在秘密重构阶段无法正确构造出相应的秘密值,从而也就确定匹配不成功.

### 3.4 安全性分析

协议 2 主要涉及 2 个密码学基础原语的应用,即秘密分享和外包茫然传输. 直观上来看,因为外包茫然传输协议是安全的,因此 $P_1$(即接收方 $R$)的输入信息对于 $P_2$ 和 $P_3$ 而言是保密的,即保证了 $t$ 不被泄露. 此外,根据秘密分享方案的性质,当分享份额数目不足以重构出秘密时,其不泄露分享份额的信息,因此 $P_3$ 根据接收到的值仅仅能知道是否匹配成功,但是并不能推测出 $P_2$ 的每一位的值到底是什么. 这样,通过以上 2 个基础原语的性质,协议的安全性得以保障. 下面,我们根据 2.4 节中的安全性定义,给出协议 $\pi_{\mathrm{TWPM}}$ 的形式化安全证明.

**定理 1.** 假设外包茫然传输协议在半诚实敌手模型下是安全的,秘密分享方案满足仅当共享份额

数目大于等于 $m$ 时才能重构出秘密,则协议 $\pi_{\mathrm{TWPM}}$ 在半诚实敌手模型下安全计算功能函数 $\mathcal{F}_{\mathrm{TWPM}}$.

证明. 我们在混合模型下证明协议 2 的安全性,其中外包模式匹配协议通过一个理想功能函数 $\mathcal{F}_{\mathrm{OOT}}$ 计算实现. 我们分别针对 $P_1$,$P_2$ 和 $P_3$ 被腐化这 3 种情况进行证明.

1) $P_1$ 被腐化. 令 $\mathcal{A}_1$ 为现实世界腐化 $P_1$ 的敌手. 我们构造一个模拟器 $\mathcal{S}_1$ 按照如下方式模拟敌手 $\mathcal{A}_1$:$\mathcal{S}_1$ 调用敌手 $\mathcal{A}_1$ 的输入 $t$,$n$ 和 $m$,并将其发送给理想世界中计算理想功能函数 $\mathcal{F}_{\mathrm{TWPM}}$ 的可信第三方. 之后,模拟器 $\mathcal{S}_1$ 模拟理想功能函数 $\mathcal{F}_{\mathrm{OOT}}$,并以 OOT 协议中诚实发送方 $P_2$ 的身份使用一个随机的模式 $p$ 生成相应的秘密分享份额发送给 $\mathcal{F}_{\mathrm{OOT}}$. 最后,模拟器 $\mathcal{S}_1$ 输出敌手 $\mathcal{A}_1$ 的输出.

以上是 $P_1$ 被腐化的理想模拟过程. 我们要证明在上述理想世界的模拟过程中敌手和模拟器执行的联合输出分布,与在现实世界的真实协议中敌手和诚实参与方产生的联合输出分布式计算不可区分的,即

$$\{IDEAL_{\mathcal{F}_{\mathrm{TWPM}},s_1(z),P_1}((t,m),(p,n,\tau))\}\equiv$$
$$\{HYBRID^{\mathrm{OOT}}_{\pi_{\mathrm{TWPM}},\mathcal{A}_1(z),P_1}((t,m),(p,n,\tau))\}.$$

注意到 $P_1$ 在协议中没有输出,且其视图信息仅包含在 OOT 协议中收到的 $P_2$ 的消息. 此外,理想模拟过程与现实协议的唯一不同是模拟器选择了一个随机的模式 $p$,而不是使用 $P_2$ 的真实输入. 鉴于外包模式匹配协议是半诚实安全的,因此敌手没有能力区分 $P_2$ 的真实输入到底是什么,这就使得随机选择一个模式 $p$ 对于敌手而言是计算不可区分的. 所以上述 $IDEAL$ 分布和 $HYBRID$ 分布是计算不可区分的.

2) $P_2$ 被腐化. 令 $\mathcal{A}_2$ 为现实世界腐化 $P_2$ 的敌手. 我们构造一个模拟器 $\mathcal{S}_2$ 按照如下方式模拟敌手 $\mathcal{A}_2$:$\mathcal{S}_2$ 调用敌手 $\mathcal{A}_2$ 的输入 $p$,$n$,$m$ 和 $\tau$,并将其发送给计算理想功能函数 $\mathcal{F}_{\mathrm{TWPM}}$ 的可信第三方. 之后,模拟器 $\mathcal{S}_2$ 模拟理想功能函数 $\mathcal{F}_{\mathrm{OOT}}$,并以 OOT 协议中诚实接收方 $P_1$ 的身份将一个长度为 $n$ 的随机串 $t$ 发送给 $\mathcal{F}_{\mathrm{OOT}}$. 最后,模拟器 $\mathcal{S}_2$ 输出敌手 $\mathcal{A}_2$ 的输出.

我们欲证明上述模拟过程和真实协议执行分别产生的输出联合分布式不可区分的,即证明

$$\{IDEAL_{\mathcal{F}_{\mathrm{TWPM}},s_2(z),P_2}((t,m),(p,n,\tau))\}\equiv$$
$$\{HYBRID^{\mathrm{OOT}}_{\pi_{\mathrm{TWPM}},\mathcal{A}_2(z),P_2}((t,m),(p,n,\tau))\}.$$

通过比较 $IDEAL$ 和 $HYBRID$ 的执行过程可以发现,唯一的不同在于模拟器选择了一个随机的 $t$ 作为诚实参与方 $P_1$ 在 OOT 协议中的输入. 根据外包茫然传输协议的安全性,$P_2$ 在 OOT 协议中的视图可以在不知道 $P_1$ 真实输入的前提下生成,且生成的视图对于敌手而言是与真实协议中的视图计算不可区分的. 鉴于此,我们推出上述理想模拟和现实协议的输出分布是不可区分的.

3) $P_3$ 被腐化. 令 $\mathcal{A}_3$ 为现实世界腐化 $P_3$ 的敌手. 我们构造一个模拟器 $\mathcal{S}_3$ 按照如下方式模拟敌手 $\mathcal{A}_3$:$\mathcal{S}_3$ 调用敌手 $\mathcal{A}_3$ 的输入(为空),并从理想世界中计算功能函数 $\mathcal{F}_{\mathrm{TWPM}}$ 的可信第三方那里接收到相应的匹配结果. 假设匹配结果为 $t_1,t_2,\cdots,t_i$,其中 $1\leqslant t_1\leqslant\cdots\leqslant t_1\leqslant n-m+1$. 则模拟器就可以根据所接收到的这些信息,模拟理想功能函数 $\mathcal{F}_{\mathrm{OOT}}$ 所输出的消息,也就是敌手 $\mathcal{A}_3$ 从 OOT 协议中接收到的输出信息. 具体的,对于匹配成功的相应子串,模拟器选择一个随机秘密值并将子串中每一位上的秘密分享份额均设置为合法的,相反,对于那些未匹配成功的子串,就可以随机设置分享份额的值(选取等长的随机串即可). 最后,模拟器 $\mathcal{S}_3$ 输出敌手 $\mathcal{A}_3$ 的输入,并终止协议.

同上所述,我们要证明理想模拟和现实协议所生成的 2 个分布不可区分,即

$$\{IDEAL_{\mathcal{F}_{\mathrm{TWPM}},s_3(z),P_3}((t,m),(p,n,\tau))\}\equiv$$
$$\{HYBRID^{\mathrm{OOT}}_{\pi_{\mathrm{TWPM}},\mathcal{A}_3(z),P_3}((t,m),(p,n,\tau))\}.$$

模拟器在模拟 OOT 协议输出时,唯一不同的地方在于如何处理那些未匹配成功的字符串. 因为秘密分享方案需要至少 $m$ 个合法秘密分享份额才能重构出秘密,因此,当匹配不成功时,也就意味着至少存在一个不合法的秘密分享份额,从而导致不能重构出正确的秘密值. 与此同时,在这种情况下也不会泄露具体哪些位置匹配成功,哪些位置未匹配成功. 所以,模拟器针对这些匹配失败的子串,通过选择随机数的方式来生成输出信息,与真实协议中实际接收到的信息是计算不可区分的. 证毕.

### 3.5 协议效率分析与比较

协议 2 的交互主要体现在外包 OT 扩展协议中,根据 2.2 节中的协议过程可知共需要 3 轮的交互. 考虑计算复杂度,主要包含秘密分享协议和外包 OT 协议中的计算量. 其中,秘密分享协议中主要涉及共享份额的生成以及秘密重构,共需要 $O(mn)$ 次操作. 注意到这里的秘密分享需要全部的分享份额来恢复密钥,因此可以使用随机数异或的形式,这样主要涉及对称密钥操作,因此实现起来更高效. 外包 OT 阶段主要涉及公钥密码操作,因此我们主要

考虑 OT 阶段的计算开销.通过协议可知,参与方之间共需要执行 $O(nm)$次 1-out-of-2 OT 协议来传输分享份额.然而,通过 OT 扩展技术,我们可以将其降低至 $O(k)$的级别,其中 $k \ll mn$.此外,协议的通信复杂度也为 $O(mn)$.表 1 是我们的协议与之前两方带通配符模式匹配协议的比较结果.

**Table 1 Comparison of Wildcard Pattern Matching Protocols**

**表 1 带通配符模式匹配协议比较**

| Protocol | Tools | Round | Computation Complexity |
|---|---|---|---|
| Ref [18] | Additively HE | $O(1)$ | $O(nm)$ |
| Ref [16] | ElGmal PKE | $O(1)$ | $O(nm)$ |
| Ref [22] | OT Extension | $O(1)$ | $O(k)$ |
| Ours | OOT Extension | 3 | $O(k)$ |

从功能性而言,之前的协议均考虑两方场景,我们首次考虑三方带通配符模式匹配协议.其次,考虑使用的密码学工具,文献[16]基于分布式的 ElGmal 加密方案,因此协议最后需要执行一个联合解密协议,文献[18]则使用带有加法性质的同态加密方案,文献[22]和我们的协议均基于 OT 扩展协议,不同的是他们的协议最后需要多次调用安全字符串相等性测试协议.从协议计算复杂度而言,文献[16,18]需要 $O(nm)$的计算量,其中 $n$ 和 $m$ 是 2 个参与方的输入长度;文献[22]和我们的协议的计算复杂度为 $O(k)$,其中 $k$ 是基础 OT 的数目,远小于 $nm$ 的值.

## 4 结束语

本文主要考虑安全带通配符模式匹配协议的高效构造,首次以具体三方场景应用为驱动,构造了一个半诚实安全的三方带通配符模式匹配协议.协议主要基于秘密分享和外包茫然传输 2 个密码学基础工具,通过秘密分享份额的设置,对通配符位和非通配符位进行分别表示,并结合外包 OT 协议,将带通配符模式匹配转化成精确模式匹配.协议需要 3 轮交互,计算和通信复杂度分别为 $O(k)$和 $O(nm)$,其中 $n$ 和 $m$ 是 2 个数据提供方的输入长度,$k$ 是实现 OT 扩展协议的基数,其值远小于 $nm$.

## 参 考 文 献


[1] Yao A. Protocols for secure computations [C] //Proc of the 23rd Annual Symp on Foundations of Computer Science (FOCS82). Los Alamitos, CA: IEEE, 1982: 160-164

[2] Yao A. How to generate and exchange secrets [C] //Proc of the 27th Annual Symp on Foundations of Computer Science (FOCS86). Los Alamitos, CA: IEEE, 1986: 162-167

[3] Goldreich O, Micali S, Wigderson A. How to play any mental game-A completeness theorem for protocols with honest majority [C] //Proc of the 19th Annual ACM Symp on Theory of Computing (STOC'87). New York: ACM, 1987: 218-229

[4] Jiang Han, Xu Qiuliang. Secure multiparty 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)

[5] Troncoso-Pastoriza J R, Katzenbeisser S, Celik M. Privacy preserving error resilient DNA searching through oblivious automata [C] //Proc of the 14th ACM Conf on Computer and Communications Security. New York: ACM, 2007: 519-528

[6] Gennaro R, Hazay C, Sorensen J S. Text search protocols with simulation based security [G] //LNCS 6056: Public Key Cryptography (PKC 2010). Berlin: Springer, 2010: 332-350

[7] Mohassel P, Niksefat S, Sadeghian S, et al. An efficient protocol for oblivious DFA evaluation and applications [G] //LNCS 7178: Topics in Cryptology (CT-RSA 2012). Berlin: Springer, 2012: 398-415

[8] Wei Lei, Reiter M K. Third-party private DFA evaluation on encrypted files in the cloud [G] //LNCS 7459: Computer Security (ESORICS 2012). Berlin: Springer, 2012: 523-540

[9] Sasakawa H, Harada H, duVerle D, et al. Oblivious evaluation of non-deterministic finite automata with application to privacy-preserving virus genome detection [C] //Proc of the 13th Workshop on Privacy in the Electronic Society. New York: ACM, 2014: 21-30

[10] 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. Berlin: Springer, 2008: 155-175

[11] Hazay C, Lindell Y. Efficient protocols for set intersection and pattern matching with security against malicious and covert adversaries [J]. Journal of Cryptology, 2010, 23(3): 422-456

[12] Katz J, Malka L. Secure text processing with applications to private DNA matching [C] //Proc of the 17th ACM Conf on Computer and Communications Security. New York: ACM, 2010: 485-492

[13] Jarrous A, Pinkas B. Secure hamming distance based computation and its applications [G] //LNCS 5536: Applied Cryptography and Network Security. Berlin: Springer, 2009: 107-124

[14] Vergnaud D. Efficient and secure generalized pattern matching via fast fourier transform [G] //LNCS 6737: Progress in Cryptology (AFRICACRYPT 2011). Berlin: Springer, 2011: 41-58

[15] Hazay C, Toft T. Computationally secure pattern matching in the presence of malicious adversaries [G] //LNCS 6477: Advances in Cryptology (ASIACRYPT 2010). Berlin: Springer, 2010: 195-212

[16] Hazay C, Toft T. Computationally secure pattern matching in the presence of malicious adversaries [J]. Journal of Cryptology, 2014, 27(2): 358-395

[17] Wei Xiaochao, Zhao Minghao, Xu Qiuliang. Efficient and secure outsourced approximate pattern matching protocol [J]. Soft Computing, 2018, 22(4): 1175-1187

[18] Baron J, El Defrawy K, Minkovich K, et al. 5pm: Secure pattern matching [G] //LNCS 7485: Security and Cryptography for Networks. Berlin: Springer, 2012: 222-240

[19] Yasuda M, Shimoyama T, Kogure J, et al. Secure pattern matching using somewhat homomorphic encryption [C] // Proc of the 2013 ACM Workshop on Cloud Computing Security Workshop. New York: ACM, 2013: 65-76

[20] Yasuda M, Shimoyama T, Kogure J, et al. Privacy-preserving wildcards pattern matching using symmetric somewhat homomorphic encryption [G] //LNCS 8544: Information Security and Privacy. Cham: Springer, 2014: 338-353

[21] Saha T K, Koshiba T. An enhancement of privacy-preserving wildcards pattern matching [G] //LNCS 10128: Foundations and Practice of Security. Berlin: Springer, 2016: 145-160

[22] Kolesnikov V, Rosulek M, Trieu Ni. SWiM: Secure wildcard pattern matching from OT extension [OL]. [2017-07-26]. https://eprint.iacr.org/2017/1150.pdf

[23] Darivandpour J, Atallah M J. Efficient and secure pattern matching with wildcards using lightweight cryptography [J/OL]. Computers & Security, 2018. [2018-06-01]. https://doi.org/10.1016/j.cose.2018.01.004

[24] Faust S, Hazay C, Venturi D. Outsourced pattern matching [G] //LNCS 7966: Automata, Languages, and Programming. Berlin: Springer, 2013: 545-556

[25] Chase M, Shen E. Substring-searchable symmetric encryption [J]. Proceedings on Privacy Enhancing Technologies, 2015, 2015(2): 263-281

[26] Wang Dongsheng, Jia Xiaohua, Wang Cong, et al. Generalized pattern matching string search on encrypted data in cloud systems [C] //Proc of the 2015 IEEE Conf on Computer Communications (INFOCOM). Los Alamitos, CA: IEEE, 2015: 2101-2109

[27] Faber S, Jarecki S, Krawczyk H, et al. Rich queries on encrypted data: Beyond exact matches [G] //LNCS 9327: Computer Security (ESORICS 2015). Berlin: Springer, 2015: 123-145

[28] Jia Nan, Jia Xiaohua, Wang Dongsheng, et al. Structured queries with generalized pattern matching on encrypted cloud data [C] //Proc of the 2016 IEEE Int Conf on Communications (ICC). Los Alamitos, CA: IEEE, 2016: 1-7

[29] Li Dongmei, Dong Xiaolei, Cao Zhenfu. Secure and privacy-preserving pattern matching in outsourced computing [J]. Security & Communication Networks, 2016, 9(16): 3444-3451

[30] Zhang Tao, Wang Xiuhua, Chow S S M. Privacy-preserving multi-pattern matching [C] //Proc of Int Conf on Security and Privacy in Communication Systems. Berlin: Springer, 2016: 199-218

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

[32] Blakley G R. Safeguarding cryptographic keys [C] //Proc of the National Computer Conf. New York: AFIPS, 1979, 48: 313-317

[33] Carter H, Mood B, Traynor P, et al. Secure outsourced garbled circuit evaluation for mobile devices [J]. Journal of Computer Security, 2016, 24(2): 137-180

[34] Rabin M O. How to exchange secrets by oblivious transfer, TR-81 [R]. Cambridge, MA: Harvard University, 1981

[35] Ishai Y, Kilian J, Nissim K, et al. Extending oblivious transfers efficiently [G] //LNCS 2729: Advances in Cryptology (CRYPTO2003). Berlin: Springer, 2003: 145-161

[36] Goldreich O. Foundations of Cryptography: Volume 2-Basic Applications [M]. Cambridge, UK: Cambridge University, 2004

**Wei Xiaochao**, born in 1990. PhD. Lecturer in Shandong Normal University. His main interests include secure multi-party computation, privacy preserving and searchable encryption.

**Zheng Zhihua**, born in 1962. Associate professor in Shandong Normal University. Her main interests include information security, cryptographic protocols and blockchain.

**Wang Hao**, born in 1984. PhD. Associate professor in Shandong Normal University. His main interests include public key cryptography, secure multiparty computation and blockchain (wanghao@sdnu.edu.cn).