文章编号： 1006-3080(2025)01-0089-09 **DOI：** 10.14135/j.cnki.1006-3080.20240228003

基于同态加密支持模糊查询的高效隐私信息检索协议

严字冬， 徐 贤
（华东理工大学信息科学与工程学院，上海 200237）

摘要： 隐私信息检索技术可以在进行数据检索的同时保护用户查询隐私，然而现有的隐私信息检索协议仍然不够高效，且功能较为薄弱。基于同态加密和数论原理，提出了一种高效的密文匹配机制，并在此之上设计了一个支持模糊查询的无交互对称隐私数据检索协议，从效率和功能性两个角度着手，显著提升了隐私信息检索的效率，并扩展了协议的功能。在本方案中，对上万条记录进行的单次查询仅需要秒级别的延时，在服务端延时方面优于当下最高效的关键词隐私信息检索方案，并且本方案允许进行带有通配符的模糊查询以及多关键词的查询，可以在保护用户和数据隐私的同时，获得趋近于明文查询的体验。

关键词： 隐私信息检索；同态加密；密文匹配；模糊查询；云服务

中图分类号： TP **文献标志码：** A

当下，伴随着云服务和大数据等技术的广泛应用，越来越多的数据被存储在云端，并对外提供服务，在云端进行数据检索和分析已经变得越来越普及[1]。然而，越来越多的信息安全问题也接踵而至[2]，人们对于数据安全和个人隐私保护的需求日益强烈，如何在使用这些服务的同时保障信息安全成为了一个重要的课题。信息检索是最常见的云端数据应用场景，当用户输入关键词在云端数据库中检索所需的数据时，其搜索内容可能会直接或间接地包含用户个人隐私信息，例如当病人通过搜索某疾病来查询医院或专家时，云端服务可以很容易地推测出病人所患的疾病，这是病人所不想看到的。

对于这类问题，隐私信息检索（Private Information Retrieval, PIR）[3] 作为安全多方计算的分支，在上世纪末被引入[4]，主要研究在保证用户的检索信息不被泄露的前提下，如何高效地进行查询操作。PIR 问题中一般有两个参与者（客户端和服务端），服务端存储一个数据库 L，其中包含的数据被视为一个数组，在进行查询时基于索引查询，例如，客户端需要 L 中的第 11 个元素的内容，双方通过执行 PIR 协议后，客户端可以获取到所需的值，而服务端无法得知用户获取的是哪一个元素。PIR 也衍生出了各类不同的变体。除了基于索引的 PIR 之外，基于关键字匹配的 PIR（keyword PIR）[5] 支持通过关键字匹配进行查询，适用于键值对存储类型的数据库。对称 PIR（Symmetrical PIR, SPIR）[6] 支持在保护用户隐私的同时，也保护数据库的数据隐私。在一些场景下，云服务仅提供存储计算能力，其中的数据由数据提供者加密后外包在云端，使用 SPIR 可以保证数据提供者的数据隐私。

然而，目前的 PIR 方案效率较低且功能较为薄弱，无法满足实际应用场景的需求，需要精确输入与键值相同的关键词才能获得结果，当用户只知道关键词的部分字词时，无法给出正确的搜索结果，明文查询中许多常用且重要的功能依然缺失。本文聚焦于设计并构造一个高效且具有强大检索能力的对称隐私信息检索协议，支持多关键词联合查询以及模糊查询。不同于传统的 keyword PIR，支持模糊查询的 PIR 在进行查询时，用户可以输入关键词进行模糊查询，例如输入“肺炎”，可以将“肺炎”，“支原体肺

收稿日期： 2024-02-28
基金项目： 国家自然科学基金（61872142, 61772200, 61173048）
作者简介： 严字冬（1998—），男，上海人，硕士生，主要研究方向为信息安全、密码学。E-mail: Y30211043@mail.ecust.edu.cn
通信联系人： 徐 贤，E-mail: xuxian@ecust.edu.cn
引用本文： 严字冬，徐 贤. 基于同态加密支持模糊查询的高效隐私信息检索协议 [J]. 华东理工大学学报（自然科学版），2025, 51(1): 89-97.
Citation: YAN Zidong, XU Xian. An Efficient Private Information Retrieval Protocol Based on Homomorphic Encryption Supporting Wildcard Query[J]. Journal of East China University of Science and Technology, 2025, 51(1): 89-97.

炎”,“肺炎症状”等关键词全都查出来,而 keyword PIR 只能搜索出精确词,无法检索出相关词。同时,在模糊匹配 PIR 中,可以使用通配符“?”来表示任意字符,例如“王?明”可以匹配“王阳明”、“王小明”等。方案还额外地支持多关键词查询,单个查询中可以包含多个查询关键词。整个协议只进行一轮传输,客户端创建并发起查询请求,传输给服务端,服务端进行匹配,生成查询结果后,将请求结果传输给客户端,客户端解密后获得查询结果。

在相关的工作中,关于隐私信息检索问题的研究大部分聚焦于在已知索引位置的查询[7-11],具有较大的限制,只能作用于特定的场景,相较之下,能够通过键值进行检索的 keyword PIR[5] 与本文所关注的场景更为相近。Keyword PIR 通常使用隐私求交等多方安全技术实现,通过多轮的交互来查询到最终的结果。近年来许多工作聚焦于改进其性能表现,例如,Chen 等[12] 使用多方安全计算中隐私求交的方式实现键与值之间的映射,Ali 等[13] 引入布谷鸟哈希来降低通信和计算开销,Wu 等[14] 利用隐私求交构造了一个支持载荷传输和批量关键字搜索的 SPIR 协议。然而,keyword PIR 的效率表现始终不尽如人意,在一些使用多方安全计算实现的方案中,需要客户端与服务端之间的多轮交互,造成较高的通信开销,并且在交互过程中一部分计算负担由性能较弱的客户端承担,导致协议的整体延时较高。最近,Mahdavi 等[15] 提出了一种基于同态加密实现的非交互方案,CKP(Constant-weight Keyword PIR)是目前效率最高的非交互方案。他们将数据重新编码为具有相同海明权重(即非 0 比特的数量)的值,再使用同态加密逐位加密,最后在检索时利用其海明权重相同的特点使用特殊的电路算法进行密文比较。然而,这种方式虽然降低了密文比较的理论复杂度,但是其特殊的编码方式会显著增加数据的大小,并且每个比特都需要单独加密,导致其计算和传输的实际开销较大。

除了性能方面的问题之外,keyword PIR 只能进行简单的关键词精确匹配,只有当用户输入精确的关键词时,才会返回正确的结果,无法进行模糊查询或返回相关联的结果,在实际检索应用中体验较差。而本文的方案中,针对这一不足,通过设计一种新的密文匹配机制,实现了使用模式串进行匹配的检索能力,可适应更多实际应用需求。

除了对客户端查询隐私的保护之外,也有不少工作[16–27] 聚焦于数据隐私的保护,主要通过在密态数据上进行搜索实现隐私保护,例如张凯鑫等[26] 基于同态加密构造了加密字符串的匹配机制,但这一匹配结果并不能被应用于后续检索、计算等应用。王缵等[27] 针对边缘计算的场景,设计了新颖且统一的轻量级安全索引结构,提出了基于移动边缘计算的安全 skyline 查询协议,但这一方案仅适用于特定场景下。可搜索加密是这一领域中较为常用的技术,不少方案利用可搜索加密构造了支持通配符查询的方案[20-24],以及支持多关键词的方案[23] 等。然而,这类基于可搜索加密的方案在安全性方面较为薄弱,无法做到对访问模式(Access patterns)的保护,即服务端可以知道用户查询内容所在的位置,并且也会遭受到关键词猜测攻击。因此,这类基于可搜索加密实现的方案主要用于保障数据隐私,但无法很好地达到保护用户查询隐私的目的。

综上所述,当前缺少一个能够在有效保护查询隐私的同时,提供丰富查询功能的通用方案,以适应不同的应用场景,例如医疗检索、金融检索、搜索引擎等。为了实现这一目标,本文基于同态加密设计了密文匹配并构造了检索协议,充分利用有限域上的性质以及同态加密的批处理能力,实现了一个能满足要求的 PIR 协议。具体地,在本文所构造的 PIR 协议中,不仅允许通过加密保障外包数据隐私,还能保障用户查询隐私不被泄漏,包括用户的查询关键词、查询长度,以及查询内容所在位置等信息,并做到对于客户端无交互的一轮传输。本方案还在性能方面有较好的表现,在服务端计算延时方面显著优于当下主流的方案[15],且通信开销较低。

相比较之下,基于同态加密的其他 PIR 方案(如文献 [15]),无法实现模糊查询的功能;基于隐私集合求交的 PIR 方案(如文献 [14])不具有模糊查询以及无交互的能力;基于可搜索加密的方案(如文献 [20-24])不能保护用户隐私和访问模式隐私,也不具备无交互的能力。

1 预备知识

1.1 同态加密

同态加密[6] 是一种高级加密算法,通常包括密钥生成、加密、解密以及提供密文上计算能力的同态计算,利用其特性可以在不解密的情况下直接在密文上进行运算,其解密后得到的明文,与直接在明文上运算的结果一致。

根据同态加密的不同特性和底层困难问题,同态加密通常被分为 3 类:第 1 类主要包含 BGV[28],BFV[29,30],CKKS[31] 这 3 种方案,他们都基于环上容错

学习问题(Ring-Learning with errors, RLWE)问题[32]构造。BGV 和 BFV 可以提供整数上高效的线性运算，而 CKKS 可以提供非精确数(Approximate number)的高效线性运算。此外，利用 SIMD[33]技术，这类方案还具有批处理的能力，即可以将多个明文打包在同一个密文的 slot 中，大大提升效率；第 2 类主要包含 GSW[34]、TFHE[35] 等方案，他们都依赖于容错学习(Learning with errors, LWE)[36] 构造而来，这类方案支持高效的位运算和布尔运算，适用于非线性电路较多的场景。

1.2　BFV 同态加密方案

由于本文的方法主要依赖于有限域上的费马小定理进行实现，所以使用了 BFV 的加密方式，此处介绍 BFV 算法中必要的知识。

BFV 中，明文域为 $m \in Z[X]_p/(X^N+1)$，其中明文模数 p 是一个整数，多项式的阶 N 是 2 的幂次，而密文属于 $ct \in Z[X]_q/(X^N+1)$，其中 $q \gg p$。除去密钥生成外，加密解密算法 BFV 提供的同态计算如下：

$\mathrm{EvalAdd}(\overline{m_1},\overline{m_2}) \to ct' = \overline{m_1+m_2}$：加法，输入两个密文，输出其对应明文相加结果的密文。

$\mathrm{EvalMult}(\overline{m_1},\overline{m_2}) \to ct' = \overline{m_1 * m_2}$：乘法，输入两个密文，输出其对应明文相乘结果的密文。除两个密文间的操作外，密文与明文也同样可以相加或相乘。

利用 SIMD 技术，通过适当的参数选择，即当满足 $p \equiv 1 \bmod N$ 时，以及数域中中国剩余定理的使用，可以将多个明文打包在一个密文中，视为一组向量，其中每个明文位置被称为一个 slot，即 $\boldsymbol{m}=(m_0, m_1, \cdots, m_{l-1})$，而与其对应的加法和乘法也可以被视为两个明文向量对应于其分量上每个元素的加法和乘法。

2　密文匹配机制构造

在本节中，将会从单个字符的匹配开始，一步步构造一个完整的密文匹配机制，并对其复杂度进行分析。

密文匹配和 PIR 协议中出现的字母、符号、标记等所代表的具体含义，在表 1 中给出。

2.1　单字符匹配

在本协议中，字符串的加密是逐字符进行的，所以单个字符的精确匹配是构造密文匹配和模糊匹配的基石，影响着整体的服务端延迟和数据传输量。

假设单个字符为 a 比特，则单个字符的精确匹配实际是解决在集合 $\mathbb{Z}_p=\{0,1,\cdots,p-1\}(p=2^a)$ 中，

表 1　符号和标记定义

Table 1　Definition of symbols and marks

Symbol and mark	Definition
$\mathbb{Z}_p$	$\{0,1,\ldots,p-1\}, p=2^a$, a is the bit length of p
x_i, s_i, p_i, w_i, etc.	Elements in $\mathbb{Z}_p$, present for one character
$\overline{x}$	Overline indicates the ciphertext
S	String to be matched
P	Pattern string
W	Wildcard vector
l_s, l_p	Length of S and P
EQ()	Ciphertext comparison
$\overline{m}=\mathrm{match}()$	Ciphertext match
$\overline{r}=\mathrm{fetch}()$	Ciphertext retrieval
?, #	Wildcard and placeholder
C	Number of keywords
n	Number of records
N	Batch size
p	Plaintext space
L	Key-value database
k_i	Key of database
v_i	Value of database

判断两个元素的密文是否相等的问题。定义 $\mathrm{EQ}(ct_1, ct_2)$ 为等值比较函数，其定义如下：

$$\mathrm{EQ}(\overline{x_1},\ \overline{x_2}) = \begin{cases} \overline{1}, x_1 = x_2 \\ \overline{0},\ \text{Otherwise} \end{cases} \tag{1}$$

当两个密文所对应的明文相等时，输出 1 的密文，否则输出 0 的密文。为便于阐述，后文中可能会出现参数及输出均为明文的 EQ，其所代表的是同样功能的等值比较函数。

在有限域中，可以使用费马小定理完成，即，对于素数 p，任意不为 p 倍数的整数 a，满足 $a^{p-1} \equiv 1 (\bmod p)$。在同态加密协议中，明文 x 小于明文模数 p，可以利用费马小定理实现等值比较函数：

$$\mathrm{EQ}(\overline{x_1},\overline{x_2}) = \overline{1} - (\overline{x_1}-\overline{x_2})^{p-1} \tag{2}$$

当 $x_1 = x_2$ 时，$\overline{x_1}-\overline{x_2}=0$，$\mathrm{EQ}(\overline{x_1},\overline{x_2})=\overline{1}$；当 $x_1 \neq x_2$ 时，$(\overline{x_1}-\overline{x_2})^{p-1}=\overline{1}, \mathrm{EQ}(\overline{x_1},\overline{x_2})=\overline{0}$，满足前文密文比较函数的定义。

2.2　密文匹配

密文匹配是需要将进行数据库中加密的字符串

S 与客户端请求中的加密模式串 P 进行匹配。记 $S:(s_1,s_2,\cdots,s_{l_s}),s_i\in\Sigma\bigcup\{\#\}$，长度为 l_s，$P:(p_1,p_2,\ldots,p_{l_p}),p_j\in\Sigma\bigcup\{?\}$，长度为 l_p，$l_p\leqslant l_s$，其中除了包含字符集 Σ 中的普通字符外，P 可能会包含通配字符'?'，S 可能会包含无意义的填充字符'#'。其中，这两个字符可以用字符集中一些私人使用区域中的代码点来表示，例如在 2 字节的 UCS-2 符集中，范围为 U+E000 至 U+F8FF 的区域可用于自定义字符，在本协议中，将通配符号?编码为 E000，将填充字符#编码为 E001。

首先，以没有填充字符和通配符的字符串为最简单的模型来进行叙述，在此场景下，当 S 的某个长度为 l_p 的子字符串与 P 相等时，则满足匹配条件，利用单字符匹配函数 EQ，可以将这一包含关系的匹配 match 视为如下关系进行表达：

$$m=\mathrm{match}(S,P)=\bigvee_{i=0}^{l_s-l_p}\left\{\wedge_{j=1}^{l_p}\mathrm{EQ}(p_j,s_{i+j})\right\} \tag{3}$$

式中，若 S 和 P 匹配，则 $r=1$，否则 $r=0$。下面对其正确性做简要说明，对于偏移量 i，若 S 的子字符串 $S':\left(s_i,s_{i+1},\ldots,s_{i+l_p-1}\right)$ 与 P 相等，则 $\wedge_{j=1}^{l_p}\mathrm{EQ}(p_j,s_{i+j})=1$，若存在某一偏移量 i 满足此关系，则 S 包含 P，$r=1$。

其中的或、和、与关系在同态加密中可以通过如下方式具体实现为：

$$\overline{m}=\mathrm{match}\left(\overline{S},\overline{P}\right)=\overline{1}-\prod_{i=0}^{l_s-l_p}\left\{\overline{1}-\prod_{j=1}^{l_p}\mathrm{EQ}(\overline{p_j},\overline{s_{i+j}})\right\} \tag{4}$$

接下来，将描述模式串中有通配符?的情况。通配符?可以匹配任意字符，所以当 $p_j=?$ 时，需要返回 1。为此引入一个新的向量 $\boldsymbol{W}=\left(w_0,w_1,\ldots,w_{len_p-1}\right)$，当 $p_j=?$ 时，$w_j=1$，否则 $w_j=0$，由客户端生成后，用与模式串 P 相同的方式发送给服务端，利用 $\boldsymbol{W}$，可以通过如下方式实现：

$$\overline{m}=\mathrm{match}\left(\overline{S},\overline{P},\overline{W}\right)=\overline{1}-\prod_{i=0}^{l_s-l_p}\left\{\overline{1}-\prod_{j=1}^{l_p}\overline{w_j}+\mathrm{EQ}(\overline{p_j},\overline{s_{i+j}})\right\} \tag{5}$$

由于通配符?与字符集中其他编码都不相同，所以当 $p_j=?$ 时，$\mathrm{EQ}(\overline{p_j},\overline{s_{i+j}})=0$，而 $w_j=1$，这两部分加和为 1，代表匹配；当 $p_j\neq?$ 时，$w_j=0$，为正常字符匹配逻辑。

2.3 隐藏长度的密文匹配

在数据库中，每个键 S 的长度可能不一致，同样，在每一次的查询中，模式串 P 的长度也可能会不同，为了不将其长度信息泄露，可以通过填充的方式实现。对于数据库中所有的键，都用特殊填充字符#填充至相同的长度，该长度大于或等于最长键的长度；而对于模式串 P，使用通配符?填充至相同的长度，即为可提供支持查询的最大长度。填充后，所有键 S 长度为 l_s，所有模式串 P 的长度为 l_p。在填充 S 后，式(5)依然正确，但当填充 P 后，模式串长度可能大于实际用户输入关键词长度，无法完整遍历，会导致遗漏匹配的情况，如图 1 所示。

为此，可以对 S 额外增加 l_p-1 个填充，如图 2 所示，此时，式(5)中的 match 函数可以正常进行匹配。

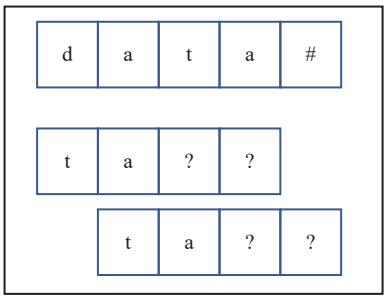

图 1 进行有限填充时无法正确匹配示意图

Fig. 1 Unable to correctly match schematic diagram during limited filling

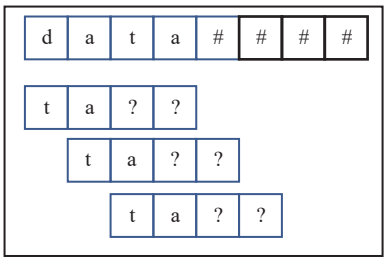

图 2 不泄露长度时对密文的填充方式

Fig. 2 Filling method for ciphertext without leaking length

2.4 复杂度分析

密文匹配过程是在同态加密密文上进行计算的过程，在同态加密中，效率的主要影响因素是乘法次数和乘法深度，所以主要通过各函数的乘法次数和乘法深度来进行计算复杂性的评估。

对于单个字符的匹配函数 EQ（即式(2)），其中 $(\overline{x_1}-\overline{x_2})^{p-1}$ 需要计算 $p-1$ 次方，通过二分法优化，需要 $\lceil\log(p-1)\rceil$ 次乘法，其乘法复杂度也为 $\lceil\log p\rceil$。

对于密文匹配函数 $\mathrm{match}\left(\overline{S},\overline{P},\overline{W}\right)$（即式(5)），首先需要计算 $(l_s-l_p+1)*l_p$ 次 EQ 匹配操作，随后将其结果进行组合相乘，需要 $(l_s-l_p+1)*l_p-1$ 次乘法，消耗 $\lceil\log((l_s-l_p+1)*l_p-1)$ 的乘法深度，故式(5)的总乘法深度为 $\lceil\log(p-1)\rceil+\lceil\log((l_s-l_p+1)*l_p-1)\rceil$。

3 PIR 协议构造

本节将基于密文匹配构造支持模糊查询的 PIR 协议，进一步构造支持多关键词的 PIR 协议，最

后通过批处理技术优化并行度和传输量。

在本文所描述的 PIR 协议中, 共有 3 方参与, 即客户端、服务端、数据拥有者。数据拥有者有一个键值对类型数据库 L, 将其加密后存储在服务端, 并授权客户端可以对数据进行检索。数据库 L 形如 $L:\{(k_1,v_1),(k_2,v_2),\cdots,(k_n,v_n)\}$, 每个键 k_i 是唯一的主键, 通常为长度较短的字符串所代表的字词, 其所使用的字符集可以为任意字符集, 例如 ASCII、Unicode 等, 而其所对应的值 v_i 没有长度限制。本方案主要针对的威胁模型为半诚实敌手模型, 协议参与者会诚实地按照协议执行, 但是可能会通过协议执行过程中获取的内容推测他方的隐私。

3.1　支持模糊查询的 PIR 协议

在 PIR 协议开始前, 协议双方需要确认协议进行过程中的密码学参数和协议参数, 随后协议开始进行, 其大致流程见图 3。

协议过程可以大致分为 4 步, 第 1 步, 客户端对查询预处理, 加密, 发送给服务端; 第 2 步, 服务端执行密文匹配; 第 3 步, 使用密文匹配的结果, 即对于每个键值 0 或 1 的匹配结果, 进行密文提取, 发送给客户端; 第 4 步, 客户端解密, 获得结果。

在协议开始前, 服务端将所需的参数发送给用户, 包括可支持查询的最大查询长度 l_p, 同态加密参数 N,p。

步骤 1: 客户端获取到用户输入的关键词, 将其填充至长度为 l_p 后, 逐字符加密, 得到密文 $\overline{P}=\left(\overline{p_1},\overline{p_2},\cdots,\overline{p_{l_p}}\right)$ 和使用的通配符 $\overline{W}=\left(\overline{w_0},\overline{w_1},\ldots,\overline{w_{l_p-1}}\right)$, 并将其发送给服务端。

步骤 2: 服务端接收到查询模式串 $\overline{P}$ 和 $\overline{W}$ 后, 对数据库中每条记录的键值 $\overline{K_i}(0\leqslant i\leqslant n-1)$ 都执行 $\overline{m_i}=\mathrm{match}\left(\overline{K_i},\overline{P},\overline{W}\right)$。

步骤 3: 进行密文的提取。步骤 2 中匹配结果 $\overline{m_i}$ 是 0 或 1 的密文, 将其与该条记录对应的值 V_i 相乘, 得到 $\overline{r_i}=\mathrm{fetch}\left(\overline{m_i},\overline{V_i}\right)=\overline{m_i}\times\overline{V_i}$。其中每条记录的 V_i 也是逐字符加密的向量值, 当不匹配时, $\overline{r_i}=\overline{0}$, 客户端无法获得任何信息; 而当匹配时, $\overline{r_i}=\overline{V_i}$, 即为用户查询的结果。服务端将所有 $\overline{r_i}$ 发送给客户端。

步骤 4: 客户端解密, 所有非 0 的内容即为查询到的数据。

这一协议可以通过简单的改动实现对多关键词的查询。当客户端所需的查询中具有多个关键词时, 对每一个关键词进行一次步骤 2, 获取单个关键词的结果, 随后使用“and”的逻辑来对多个结果进行连接, 最终生成 1 个匹配结果并正常进入步骤 3 中。

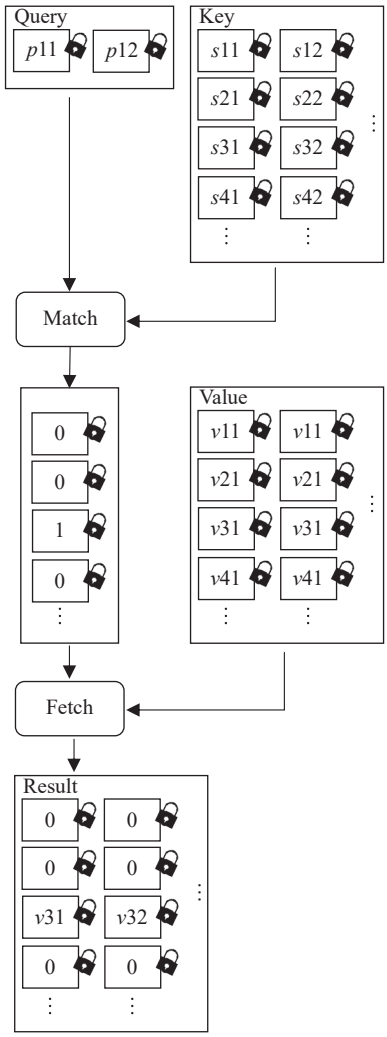

图 3　PIR 协议基本流程

Fig. 3　Process of the PIR protocol

对于 $\overline{m_i}$ 而言, 匹配成功的密文为 1, 否则为 0, 所以要实现“and”逻辑连接仅需将多个结果进行相乘即可。

3.2　正确性和安全性分析

协议的正确性得益于密文匹配机制, 协议的正确性较为明显, 可以通过简单的演绎推理得证。

协议的安全性。对于客户端而言, 客户端的查询内容经过同态加密算法进行加密, 具备机密性, 且每次查询时查询内容的长度也相同, 攻击者或服务器无法从中获取任何有效信息。在整个协议中, 所有的操作均在密文上完成, 包括密文匹配、密文提取等, 充分保证了查询的隐私。同态加密的机密性和非确定性保证了对于每一条具体的数据记录, 其匹配结果对于服务端而言都是不可区分的, 无法从这一中间结果中推论出任何结论。进一步地, 服务端最终返回的查询结果包含一整列的数据, 所以协议不会对服务端暴露查询内容所在的位置, 能够做到访问模式(Access patterns)隐私。

同时, 客户端也无法获取到除查询内容外的信

息,所以对于数据拥有者,协议能保证数据的隐私性。

3.3 批处理优化

在上述协议中,每个字符单独作为一个密文加密,对于数据库中的每一条记录,都需要进行匹配计算,同时,在传输结果时,每条记录也都对应一个密文需要传输,即,服务端计算复杂度和整体传输量都与记录条数成线性关系,在数据库数据量较大时,协议性能较差。本文使用同态加密中的SIMD技术进行批处理优化,即,可以将多个明文打包在同一个密文的多个slot中,随后将密文作为若干明文的向量,进行批量计算,BFV同态加密方案中最大批大小(即slot数量)为 N,例如32 768,可以提供较高的并发度,从而显著降低服务端计算延时和协议传输量。

数据库中记录的条数通常较多,所以本文在进行批处理时以每条记录作为主要并行点,将多条记录的相同字符都打包在同一密文中。例如 $(k_{11},k_{12},v_1),(k_{21},k_{22},v_2),(k_{31},k_{32},v_3)$,这3条记录的键含有2个字符,值包含1个字符,则压缩后服务端会按照字符分别存储为3个密文 $ct_1=\overline{(k_{11},k_{21},k_{31})}$,$ct_2=\overline{(k_{12},k_{22},k_{32})}$,$ct_3=\overline{(v_1,v_2,v_3)}$。使用批处理优化时,服务端存储的数据通过这种方式压缩后存储,而对于客户端,在生成请求时,同样需要用这种方式,使用相同的字符填充所有slot,随后加密并发送给服务端。在密文匹配和密文提取阶段,依然保持原有的操作,但每次操作时的单个密文实际上都对应着多条记录批量执行。

以单个字符作为最小单位,将多条记录相同位置的字符放在同一密文中,这样的方式还可以带来极大的可扩展性和灵活性,较大的批大小 N 可以使得计算与传输效率对记录条数的增长不敏感。由于各字符的独立性,可以灵活支持不同的最大查询长度 l_p,也可以提供较高的并发性,在云环境下,通过横向扩展机器,使用分布式并行计算,进一步优化服务端延时。

3.4 复杂度分析

假设数据库具有 n 条记录,每条记录的键长度为 l_s,最大查询长度为 l_p,共有 C 个查询关键词。当进行精确查询时,客户端在上传阶段上传 C 个密文,而在下行阶段下载 n 个密文,当进行模糊查询时,上传阶段需要上传 $C*(1+n)$ 个密文,下行阶段下载 n 个密文,两种模式下查询服务端均需要执行 $n*C$ 次match,$C-1$ 次条件间逻辑运算和 n 次fetch。当进行批处理优化后,批大小为 N,模糊查询所需要的上传量缩减为 $2C$ 个密文,两种查询模式的下行均缩减 $\lceil n/N\rceil$ 个密文,服务端运算缩减为 $\lceil n/N\rceil*C$ 次match、$C-1$ 次条件间逻辑运算和 $\lceil n/N\rceil$ 次fetch。

4 实验评估

4.1 实验环境和参数设置

本方案的实验代码使用C++编程语言,在同态加密库OpenFHE[37](v0.9.2)上实现。所有实验都是在Linux虚拟环境中,i6700HQ CPU @2.6 GHz和16 GB内存的笔记本电脑上进行。使用OpenFHE库提供的BFV作为全同态加密方案,选择的参数满足对明文空间、乘法深度和128位安全等级的要求,单个字符是16位,可以适配大部分的字符集,例如ASCII、Unicode(UCS-2)等,所以设置明文模数 $p=65\,537$,在精确查询和模糊查询中,本方案PIR协议的乘法深度均在16~20之间,故设置 $N=32\,768$。

4.2 精确查询

精确查询的实验部分包括本方案和CKP在键值分别为32位和64位下,对于单个关键词精确查询的表现。其中数据库具有随机生成的 $n=32\,768$ 条记录,值长度均为256位。在关于CKP的评估中,参数设置类同于本方案,其中的特殊参数,海明权重(k)依照其在[15, table5]中表现最好的情况设置,分别为16和32,其实验结果见表2。

在整体服务端延时方面,本方案具有显著的提升,这一提升得益于使用数论原理实现的高效的等值比较函数和批处理的优化。面对数据量较大的数据库,本方案可以在单次运行下批量处理32 768条记录,故其均摊服务端延时具有显著的优势。对于CKP,虽然其通过特殊编码的方式,乘法深度较低,但

表2 精确查询主要指标对比

Table 2 Key metrics comparison of precise query

Key/bit	Scheme	Multiplicative depth	Multiplication	Computational cost/s	Communication cost/kB
32	CKP	5	491 520	1 451.69	322
	This paper	18	34	5.682	12 200
64	CKP	6	507 904	3 010.58	322
	This paper	19	68	9.871	12 200

是其每个字符在具有相同海明权重的编码时长度均显著增加, 并且每个比特都需要一个单独的密文, 这导致其在密文比较中需要的乘法次数为 $n*(k-1)$ 次。而在本文的方案中, 其所需的乘法次数仅为 $\lceil n/N \rceil * (\lceil \log(p-1) \rceil * l_p + l_p - 1) + 1$ 。除此之外, 由于 CKP 使用了特殊的压缩方式, 在服务端还需要进行额外的展开操作, 这也造成了额外的服务端时间开销和客户端压缩开销。在总通信量方面, CKP 的压缩方式有效降低了通信量, 而本方案使用了较为通用的批处理方式进行压缩, 其传输结构和服务端后续处理时所需的结构一致, 具有更好的适用性, 且通信量依然控制在较小的范畴内。

综合而言, 本方案显著降低了作为主要瓶颈的服务端计算延时, 使得对上万条记录的数据库进行单次查询的端到端实际时间缩减至 10 s 内。

4.3　模糊查询

本方案在模糊查询以及多关键词查询时的表现如表 3 所示, 其中数据库是随机生成的 $n = 32768$ 条记录, 值长度均为 256 位。

表 3　模糊查询时的表现

Table 3　Performance of fuzzy query

Query parameter	Multiplication	Computational cost/s
$l_s = 2, l_p = 2, C = 1$	34	5 593
$l_s = 2, l_p = 2, C = 2$	68	13 526
$l_s = 4, l_p = 2, C = 1$	102	16 492
$l_s = 4, l_p = 2, C = 2$	204	31 845
$l_s = 8, l_p = 4, C = 1$	340	59 137

实验中分别选取了不同的数据库键长度和最大查询长度, 当 $l_s = 4$、$l_p = 2$、$C = 2$ 时, 可以支持最多两个关键词, 查询长度为 2 字符的模糊查询; 当 $l_s = 8$、$l_p = 4$、$C = 1$ 时, 可以支持数据库键长度为 8 字符、查询长度为 4 字符的模糊查询, 并满足覆盖大部分常见使用场景的需求。当乘法深度相近, 同态加密参数相同时, 运行时间主要取决于同态乘法的执行次数, 乘法次数为 $((\log(p-1)+1)*(l_s - l_p + 1)*l_p - 1)*C + C - 1$ 。

当查询参数较小, 例如单关键词、2 字符长度的查询时, 可以提供秒级别的端到端查询速度; 而当查询参数较大时, 依然能提供不到 1 min 的查询速度。另外, 实验是在资源受限的个人设备上进行的, 在计算资源更为丰富的云端时可以提供更高效的计算, 亦可以利用本方案较好的并发能力, 将若干次匹配运算分布在不同的节点进行计算, 从而显著提升效率, 达到接近于明文查询的效果。

5　结　论

本文基于同态加密, 提出了一种基于费马小定理的密文比较算法以及密文匹配机制, 可以支持在加密字符串上使用带有通配符的加密模式串进行匹配。这一密文匹配机制可以在多种不同的场景下发挥作用, 例如应用于隐私信息检索, 或结合倒排索引的思想应用于密文全文检索中。基于此密文匹配机制, 本文设计了支持模糊查询的高效无交互 PIR 协议, 显著提升了隐私信息检索的效率, 并扩展了多种检索方式的功能, 极大地提升了协议的实用性。该协议具有以下优秀的特性:

(1)在安全性方面, 本协议能够保障用户查询隐私不被泄漏, 包含用户查询信息、用户访问模式, 并同时保障数据隐私。

(2)在功能方面, 本方案首次实现了一个可以支持模糊查询和多关键词查询的隐私信息检索协议, 实现了近乎与当前明文检索相同的检索能力。模糊查询中支持对带有通配符的密文模式串进行匹配, 且单个查询中可以包含多个查询关键词。

(3)在效率方面, 得益于高效的密文匹配机制以及批处理优化, 本方案的单次查询传输量较低(约为 10 MB), 且服务端计算延时显著优于当下主流的方案[15]。此外, 协议运行仅需在云服务器与客户端之间进行一轮传输, 除生成查询和解密结果外, 主要计算负载均在服务端执行, 很好地利用了云服务的存储和计算资源。

(4)在本方案中, 对上万条记录进行的单次查询仅需要秒级别的延时, 同时本方案的批处理优化还具有高度的并行性和可扩展性, 可以将每个字符的匹配任务进行单独的拆分, 在资源丰富的云环境下可以充分进行分布式计算, 在应对海量数据的同时, 提供快速的响应。强大的查询功能叠加查询效率上的优化, 使得本协议能够在保护隐私的同时, 获得趋近于明文查询的体验。

参考文献:

[1] 孙寒玉, 顾春华, 万锋, 等. 一种基于 OpenStack 的云应用开发框架 [J]. 华东理工大学学报 (自然科学版), 2015, 41(2): 272-276.

[2] 荣喜丰. 云计算网络环境下的信息安全研究 [J]. 网络安全技术与应用, 2021(7): 83-84.

[3] 张小青, 张舒黎, 雷术梅, 等. 私有信息检索技术分析对比研究 [J]. 通信技术, 2023, 56(2): 198-206.

[4] CHOR B, KUSHILEVITZ E, GOLDREICH O, *et al*. Private information retrieval[J]. Journal of the ACM (JACM), 1998, 45(6): 965-981.

[5] CHOR B, GILBOA N, NAOR M. Private information retrieval by keywords[J]. Cryptology ePrint Archive, 1998, 1998: 3.

[6] SUN H, JAFAR S A. The capacity of symmetric private information retrieval[J]. IEEE Transactions on Information Theory, 2019, 65(1): 322-329.

[7] ACAR A, AKSU H, ULUAGAC A S, *et al*. A survey on homomorphic encryption schemes: Theory and implementation[J]. ACM Computing Surveys (Csur), 2018, 51(4): 1-35.

[8] AGUILAR M C, BARRIER J, FOUSSE L, *et al*. XPIR : Private information retrieval for everyone[J/OL]. (2015-01-02) [2015-09-04]. https://ia.cr/2014/1025.

[9] ANGEL S, CHEN H, LAINE K, *et al*. PIR with compressed queries and amortized query processing[C]//2018 IEEE Symposium on Security and Privacy (SP). [s.l.]: IEEE, 2018: 962-979.

[10] MUGHEES M H, CHEN H, REN L. OnionPIR: Response efficient single-server PIR[C]//Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2021: 2292-2306.

[11] CORRIGAN-GIBBS H, HENZINGER A, KOGAN D. Single-server private information retrieval with sublinear amortized time[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Cham, Switzerland: Springer International Publishing, 2022: 3-33.

[12] CHEN H, HUANG Z, LAINE K, *et al*. Labeled PSI from fully homomorphic encryption with malicious security[C] // Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2018: 1223-1237.

[13] ALI A, LEPOINT T, PATEL S, *et al*. Communication–computation trade-offs in PIR[C]//30th USENIX Security Symposium 2021. [s.l.]: [s.n.], 2021: 1811-1828.

[14] WU Z, ZHANG D, LI Y, *et al*. PSKPIR: Symmetric keyword private information retrieval based on psi with payload[J]. Cryptology ePrint Archive, 2023, 2023: 1631.

[15] MAHDAVI R A, KERSCHBAUM F. Constant-weight PIR: Single-round keyword PIR via constant-weight equality operators[J/OL]. (2022-02-15) [2022-02-16] https://doi.org/10.48550/arXiv.2202.07569.

[16] CHEON J H, KIM M, KIM M. Optimized search-and-compute circuits and their application to query evaluation on encrypted data[J]. IEEE Transactions on Information Forensics and Security, 2016, 11(1): 188-199.

[17] KIM M, LEE H T, LING S, *et al*. On the efficiency of FHE-based private queries[J]. IEEE Transactions on Dependable and Secure Computing, 2018, 15(2): 357-363.

[18] TAN B H M, LEE H T, WANG H, *et al*. Efficient private comparison queries over encrypted databases using fully homomorphic encryption with finite fields[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 18(6): 2861-2874.

[19] CHEON J H, KIM M, KIM M. Search-and-compute on encrypted data[C]//Financial Cryptography and Data Security: FC 2015 International Workshops. [s.l.]: Springer Berlin Heidelberg, 2015: 142-159.

[20] HU C, HAN L. Efficient wildcard search over encrypted data[J]. International Journal of Information Security, 2016, 15(5): 539-547.

[21] FABER S, JARECKI S, KRAWCZYK H, *et al*. Rich queries on encrypted data: Beyond exact matches[C]//Computer Security--ESORICS 2015: 20th European Symposium on Research in Computer Security. Vienna. Austria: Springer International Publishing, 2015: 123-145.

[22] ZHAO F, NISHIDE T. Searchable symmetric encryption supporting queries with multiple-character wildcards[C] // CHEN J, PIURI V, SU C, *et al*. Network and System Security. Cham, Switzerland: Springer International Publishing, 2016: 266-282.

[23] 伍祈应, 马建峰, 李辉, 等. 支持用户撤销的多关键字密文查询方案 [J]. 通信学报, 2017, 38(8): 183-193.

[24] CHEN J, HE K, DENG L, *et al*. EliMFS: Achieving efficient, leakage-resilient, and multi-keyword fuzzy search on encrypted cloud data[J]. IEEE Transactions on Services Computing, 2017, 13(6): 1072-1085.

[25] 黄保华, 赵统, 雷素梅, 等. 基于语义的模糊关键词排序可搜索加密方案 [J]. 广西大学学报 (自然科学版), 2023, 48(5): 1167-1180.

[26] 张凯鑫, 杨晨, 李顺东. 字符串匹配的保密计算 [J]. 密码学报, 2022, 9(4): 619-632.

[27] 王缵, 丁晓锋, 周潘, 等. 移动边缘计算中基于位置信息的安全 skyline 查询处理方法 [J]. 中国科学:信息科学, 2021, 51(10): 1721-1737.

[28] BRAKERSKI Z, GENTRY C, VAIKUNTANATHAN V. (Leveled) fully homomorphic encryption without bootstrapping[J]. ACM Transactions on Computation Theory, 2014, 6(3): 1-36.

[29] BRAKERSKI Z. Fully Homomorphic encryption without modulus switching from classical gapSVP[C]//SAFAVI-NAINI R, CANETTI R. Advances in Cryptology – CRYPTO. Berlin, Heidelberg: Springer Berlin Heigdelberg, 2012: 868-886.

[30] FAN J, VERCAUTEREN F. Somewhat practical fully homomorphic encryption[J]. Iacr Cryptology Eprint Archive, 2012, 2012: 144.

[31] CHEON J H, KIM A, KIM M, *et al*. Homomorphic encryption for arithmetic of approximate numbers[C]//Advances in Cryptology–ASIACRYPT 2017: 23rd International Conference on the Theory and Applications of Cryptology and Information Security. Hong Kong, China: Springer International Publishing, 2017: 409-437.

[32] LYUBASHEVSKY V, PEIKERT C, REGEV O. On ideal lattices and learning with errors over rings[C]//Advances in Cryptology–EUROCRYPT 2010: 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques. Riviera, French: Springer Berlin Heidelberg, 2010: 1-23.

[33] SMART N P, VERCAUTEREN F. Fully homomorphic SIMD operations[J]. Designs, Codes and Cryptography, 2014, 71(1): 57-81.

[34] GENTRY C, SAHAI A, WATERS B. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based[C]//Advances in Cryptology–CRYPTO 2013: 33rd Annual Cryptology Conference. Santa Barbara, CA, USA: Springer Berlin Heidelberg, 2013: 75-92.

[35] CHILLOTTI I, GAMA N, GEORGIEVA M, *et al*. TFHE: fast fully homomorphic encryption over the torus[J]. Journal of Cryptology, 2020, 33(1): 34-91.

[36] REGEV O. On lattices, learning with errors, random linear codes, and cryptography[J]. Journal of the ACM, 2009, 56(6): 1-40.

[37] Al BADAWI A, BATES J, BERGAMASCHI F, *et al*. Openfhe: Open-source fully homomorphic encryption library[C]//Proceedings of the 10th Workshop on Encrypted Computing & Applied Homomorphic Cryptography. [s.l.]:[s.n.], 2022: 53-63.

An Efficient Private Information Retrieval Protocol Based on Homomorphic Encryption Supporting Wildcard Query

YAN Zidong, XU Xian

(School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, China)

Abstract: Private information retrieval techniques can protect user query privacy while conducting data retrieval. However, the existing privacy information retrieval protocols are still not efficient enough and have relatively weak functions. Based on homomorphic encryption and number theory principles, an efficient ciphertext matching mechanism is proposed, and a non-interactive symmetric private data retrieval protocol is designed to support fuzzy queries. From the perspectives of efficiency and functionality, the efficiency of privacy information retrieval is significantly improved, and the functionality of the protocol is expanded. Specifically, in the proposed scheme, a single query on tens of thousands of records only requires a delay of seconds, superior to the most efficient keyword-based privacy information retrieval schemes currently available in terms of server-side latency. Additionally, this scheme allows for fuzzy queries with wildcards and multi-keyword queries, enabling users to obtain an experience similar to plaintext queries while protecting user and data privacy.

Key words: private information retrieval**;** homomorphic encryption**;** ciphertext**;** wildcard query**;** cloud service

（责任编辑：张欣）