软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2021,32(10):3266−3282 [doi: 10.13328/j.cnki.jos.006086]

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

医疗大数据隐私保护多关键词范围搜索方案*

张明武[1,2,3], 黄嘉骏[1,3], 韩 亮[4]

[1](湖北工业大学 计算机学院,湖北 武汉 430068)

[2](密码科学技术国家重点实验室,北京 100878)

[3](桂林电子科技大学 计算机与信息安全学院,广西 桂林 541004)

[4](Department of Computer Science and Electrical Engineering, University of Missouri-Kansas City, Kansas 64110, USA)

通讯作者: 张明武, E-mail: csmwzhang@gmail.com

摘 要: 随着医疗信息系统的急速发展,基于医疗云的信息系统将大量电子健康记录(EHRs)存储在医疗云系统中,利用医疗云强大的存储能力和计算能力对 EHRs 数据进行安全与统一的管理.尽管传统加密机制可以保证医疗数据在半诚实云服务器中的机密性,但对加密后的 EHRs 数据执行安全、快速、有效的范围搜索,仍是一个有待解决的关键问题.提出一种支持多关键词范围搜索的可搜索加密方案:利用向量积保持加密机制实现复杂查询结构的可搜索加密,可支持连接关键词查询、范围查询以及通配符的查询;通过随机化构建搜索索引和搜索陷门,实现搜索模式隐藏,达到搜索语句的隐私保护;采用矩阵哈达马积缩小所需密钥矩阵的维度.理论分析和实验结果表明:该方案在达到医疗数据隐私保证的同时,对用户的检索策略也进行了有效的隐私性保护,有效提高了检索效率,降低了创建索引及陷门所用时间,实现了多用户多文件下医疗数据的范围搜索能力.

关键词: 隐私保护;搜索加密;非对称向量积加密;哈达马积;医疗云

中图法分类号: TP309

中文引用格式: 张明武,黄嘉骏,韩亮.医疗大数据隐私保护多关键词范围搜索方案.软件学报,2021,32(10):3266−3282. http://www.jos.org.cn/1000-9825/6086.htm

英文引用格式: Zhang MW, Huang JJ, Harn L. Range-based multi-keyword searchable scheme with privacy protection in e-healthcare cloud systems. Ruan Jian Xue Bao/Journal of Software, 2021,32(10):3266−3282 (in Chinese). http://www.jos.org.cn/1000-9825/6086.htm

Range-based Multi-keyword Searchable Scheme with Privacy Protection in e-Healthcare Cloud Systems

ZHANG Ming-Wu[1,2,3], HUANG Jia-Jun[1,3], HARN Lein[4]

[1](School of Computer Science, Hubei University of Technology, Wuhan 430068, China)

[2](State Key Laboratory of Cryptology, Beijing, 100878, China)

[3](School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, China)

[4](Department of Computer Science and Electrical Engineering, University of Missouri-Kansas City, Kansas 64110, USA)

Abstract: With the rapid development of medical information systems, the information system based on medical clouds stores massive electronic health records (EHRs) in medical cloud systems and employs the powerful storage and computing capacity of medical clouds to manage EHRs in a safe and unified manner. Although the traditional encryption mechanism can protect the privacy of medical data in semi-honest cloud servers, it is still an open problem to perform safe and efficient range-based search for the encrypted EHRs. To address

* 基金项目: 国家自然科学基金(62072134, U2001205); 密码科学与技术国家重点实验室开放课题; 广西自然科学基金(2019JJD170020)

Foundation item: National Natural Science Foundation of China (62072134, U2001205); The Open Research Project of State Key Laboratory of Cryptology of China; Natural Science Foundation of Guangxi Province of China (2019JJD170020)

收稿时间: 2019-03-22; 修改时间: 2019-10-13; 采用时间: 2020-05-13

this problem, in this work, a range-based multi-keyword searchable scheme is proposed. It can implement searchable encryption of complex query structures with scalar-product preserving encryption and support the query of connection keywords, ranges, and wildcard characters. Furthermore, the indexes and trapdoors are created in a random manner to hide the search mode and protect the privacy of search statements. The Hadamard product is adopted to reduce the dimension of the required key matrix. Theoretical analysis and experimental results show that the scheme can efficiently protect the privacy users' search strategy while guaranteeing the privacy of medical data. This scheme improves the retrieval efficiency and reduces the time in index and trapdoor creation, achieving the range-based search of medical data in multi-user and multi-file medical environments.

Key words: privacy protection; searchable encryption; asymmetric scalar-product preserving encryption; Hadamard product; medical clouds

近年来,纸质病历在医疗系统中由于不方便储存和管理正在逐步被淘汰,电子健康记录(electronic health records,简称 EHRs)应用越来越普及.对许多医院和医疗系统而言,如何实现 EHRs 的有效管理、存储和高效检索,是有待解决的主要问题之一.医疗云系统以其强大的存储能力和计算能力,为 EHRs 计算和存储提供了有效的解决方案.将大量 EHRs 数据储存在云端,可极大地减少医疗中心维护、计算和存储的开销.然而,医疗云属于第三方商业公司外包服务,存储在云端的医疗数据的隐私和安全风险可能会导致巨大的经济损失和社会信誉问题.因此,研究实现敏感健康数据隐私保护条件下的安全存储与检索,是目前重要的研究内容之一.

将 EHRs 数据加密后上传到云端,当用户需要提取数据时再将其下载后解密,似乎能够满足对于病人和医生对 EHRs 在云服务下隐私保护的需求.然而,EHRs 数据加密后不再具有原有的可直接检索和访问特性,当医疗机构需要检索医疗数据时,无法直接在密文中分辨出所需的数据.在数据模型较小的情况下,用户可以将所有密文数据下载至本地,先解密后再在明文中搜索自己想要的数据.随着云端数据规模的急剧扩大,医疗大数据也将以指数级的速度增长,这种浪费大量时间开销与通信带宽的做法显然已不能满足实际需求,也不利于精准的范围搜索.因此,如何做到在医疗大数据中既确保 EHRs 数据隐私,又能对加密后的海量 EHRs 数据进行快速且精准的多关键词搜索,同时确保用户检索谓词的隐私,是目前迫切需要解决的问题.

可搜索加密(searchable encryption,简称 SE)是一种能够使得用户对加密文件中某些关键词进行查询的密码技术,不仅可以在密文上进行搜索,而且还能够保护数据在半诚实医疗云中的安全性.Golle 等人[1]提出了可搜索加密机制,将文件与其对应关键词绑定,可实现密文搜索,但不支持范围搜索;文献[2,3]通过对加密搜索文件构建关键词索引的方式提高搜索效率,但不能实现范围搜索.文献[4]基于 Simhash 的降维思想,将文档关键词进行 *n*-gram 处理并得到 Simhash 指纹以实现模糊搜索.

考虑到医疗文件搜索特性,需对加密后的医疗文件实现范围查询.文献[5]利用非对称向量积保持加密构建关键词索引,并在生成索引向量时引入属性树的概念,缩短构造索引向量的大小,实现在精确搜索的同时,具有比基于双线性对的可搜索加密更好的搜索效率.由于引入了属性树,使得该方案只具有限制的范围精确搜索.

对称可搜索加密会依赖于安全信道,为了克服该问题,Boneh 等人[6]首次提出了可搜索公钥加密:数据拥有者建立文件关键字索引,使用接收者公钥加密文件和关键字索引,并上传到云服务器;用户用自己的私钥生成待检索关键字的陷门并发送给云,云服务器将加密的关键字索引与陷门进行匹配,将搜索结果返回给用户.文献[7]提出了支持多关键词范围搜索的公钥可搜索加密,将关键词放入合数阶群双线性运算中,但搜索速度缓慢,效率较低.

根据医疗信息系统多用户搜索实际情况,以高效进行密文搜索的同时满足对多关键词搜索,以及关键词内的范围精确搜索需求,本文提出一种多关键词范围可搜索加密方案.主要贡献如下:

(1) 基于范围的多关键词精确搜索:引入非对称向量积保持加密机制构建关键词索引,支持具有复杂查询结构的连接查询功能,支持基于范围的多关键词精确搜索;

(2) 降低系统初始化开销:通常可搜索加密方案的关键词集合较大,所创建的文件关键词的向量维度也很大,因此在构建索引和陷门时,使用的可逆矩阵也会很大,这导致系统开销急剧增长.本文在构建索引和陷门时,将文件关键词向量切分成长度相同的向量段,并将向量段拼接成对应矩阵,使得所需可逆矩阵减小,极

大地缩短了系统时间.在本文实验中,当关键词总量n=1000时,方案所用时间花销是文献[8]中方案的1/30左右.同时,本方案对关键词量n的剧烈增长有着较强的抵抗性,符合大数据环境下,关键词总量较大的电子医疗系统的要求;

(3) 降低构建索引和陷门时间:构建索引和陷门时采用文档向量和矩阵乘法运算生成,矩阵过大直接导致计算时间增多.本文将关键词向量通过分段构建成关键词矩阵,将关键词矩阵与设计的可逆矩阵做哈达码积,使得构建索引和陷门所需时间大为降低.

1 相关工作

Song 等人[9]首次提出了可搜索加密的概念,采用将文档文件划分为若干词组,对词组进行加密.在搜索阶段,服务器需要扫描整个密文词组进行匹配,但不能提供范围搜索.随后,Curtmola 等人[10]构造了加密的哈希表索引,表中包含关键词陷门和关键词的文档标识集合.文献[11,12]提出了关键词排序搜索方案,通过对相关度进行保序加密,以实现对搜索结果的精确排序.

Li 等人[13]提出了关键词模糊匹配的搜索方案,一定程度上实现了模糊范围搜索.文献[4]基于 Simhash 的降维思想,将文档关键词做 n-gram 处理并得到 Simhash 指纹来实现模糊搜索.但模糊搜索方案中只考虑到关键词字符上的模糊处理,实际应用中存在大量同义词现象,不能输入关键词的所有同义词来查询所需文档,导致搜索结果准确度较低.

为了实现精准的范围查询,Cao 等人[14]提出为每个文档创建文档向量,并利用向量空间模型和安全 KNN(K-nearest neighbour)思想,实现多关键词的排序搜索.但是利用 KNN 实现范围查询需要很多次的重复迭代,导致效率较低.文献[15–17]引入了布隆过滤器以减少存储空间,但对模糊集合中每个关键词都需要用多个哈希函数来将其插入到布隆过滤器中,因此会增加计算开销.文献[18,19]实现了基于对称的可搜索加密,不适用于医疗搜索模型下的多用户环境.

文献[6]提出的基于公钥的可搜索加密方案和文献[20]中提出的基于属性的可搜索加密方案可以扩展实现非对称密钥下的搜索加密,但该方案不能实现范围搜索.Xu 等人[21]构造出第一个可以抵抗关键词猜测攻击并支持模糊关键字搜索的加密方案.Ma 等人[22]为移动医疗系统设计了一种无证书的可搜索加密,以解决搜索加密系统的证书托管问题.为了提高搜索陷门的隐私性,Wang 等人[23]设计了支持多关键字的无需安全信道的可搜索公钥加密.Chen 等人[24]提出了支持关键字搜索的双服务器加密方案.上述方案不支持范围搜索能力.文献[25–28]将关键词和文件映射到计算代价较高的合数阶群上并作双线性运算,搜索效率较低.文献[29]提出了一种可以抵抗敏感信息泄露的加密,以解决解密密钥或陷门在可能被泄露情况下的安全性,但该方案不能提供搜索功能[30].文献[31]提出一种医疗云中实现动态搜索能力的可搜索加密,但其基于对称的可搜索加密,实际应用中对密钥管理带来一定的复杂性.文献[32]提出一种可验证的基于词典的可搜索加密方案,能够验证搜索结果的完备性.文献[33]提出一种达到前向安全的轻量级的可搜索公钥加密方案,支持对工业物联网场景的安全数据搜索应用.

2 预备知识

2.1 符号说明

本文中,n 表示医疗数据文件中的关键词数,m 表示多关键词搜索索引数($m \leqslant n$).文中将使用小写符号 a、b 等表示标量,大写符号 I、Q 等表示向量,使用空心大写字符 $\mathbb{A}$、$\mathbb{B}$ 表示矩阵.表 1 对本文中主要出现的符号进行了描述.

Table 1 Terms and notations

表 1 主要符号说明

符号	说明	符号	说明
F	电子健康记录文件 EHRs	TD_S	向量 $\vec{S}$ 对应的搜索陷门
CT	加密的 EHRs 文件	E_{key}, D_{key}	对称加密/解密算法
KEY	加密 EHRs 文件的对称密钥	$SK=\{s,P,M_1,M_2\}$	HC 整体私钥
$\mathcal{A},\mathcal{M}$	矩阵	$\mathcal{P}$	划分二进制矩阵
$W=(w_1,w_2,\ldots,w_n)$	EHRs 文件提取关键词集合	$\vec{Q}$	所有位置为 1 的参考向量
$\hat{I}'=(\hat{I}'_1,\hat{I}'_2,...,\hat{I}'_m)$	EHRs 文件加密的搜索索引	$h=n/l$	划分向量段数
$\vec{S}$	用户搜索向量	$\mathcal{A}*\mathcal{B}$	矩阵 $\mathcal{A}$ 和 $\mathcal{B}$ 的哈达马积

2.2 哈达马积

设两个 $m\times n$ 的矩阵 $\mathcal{A}=[a_{ij}]$与 $\mathcal{B}=[b_{ij}]$(其中,$i\in[1,m]$,$j\in[1,n]$),两个矩阵的哈达马积(Hadamard product)记为 $\mathcal{A}*\mathcal{B}$,哈达马积的元素定义为两个矩阵对应元素的乘积:$(\mathcal{A}*\mathcal{B})_{ij}=a_{ij}b_{ij}$,即 $\mathcal{A}*\mathcal{B}=|a_{ij}b_{ij}|_{m\times n}$.值得注意的是,只有大小完全相同的矩阵才能进行哈达马积运算.

2.3 非对称向量积保持加密

在非对称的向量积保持加密方案(asymmetric scalar-product preserving encryption,简称 APSE)[17]中,设 E_Q 和 E_T 分别为查询矢量和属性矢量的加密算法.设 I'_i 为 $\vec{I}_i$ 的加密矢量输出,Q'是 $\vec{Q}$ 的加密矢量输出,即:

$$I'_i=E_T(\vec{I}_i,k)=\mathcal{M}^T\vec{I}_i,Q'=E_Q(\vec{Q},k)=\mathcal{M}^{-1}\vec{Q},$$

其中,$\mathcal{M}$ 是 $n\times n$ 的可逆矩阵.该方案能够保持 $\vec{I}_i$ 和 $\vec{Q}$ 的内积运算:

$$I_i'^T\cdot Q'=\vec{I}_i^T\mathcal{M}\mathcal{M}^{-1}\vec{Q}=\vec{I}_i^T\cdot\vec{Q}.$$

显然,ASPE 方案可以实现查询矢量 $\vec{Q}$ 和属性矢量 $\vec{I}_i$ 的内积.

3 问题描述

3.1 系统模型

本节中给出面向医疗云的范围可搜索加密方案系统模型,如图 1 所示.系统中存在 3 个参与角色:医疗中心 HC(health center)、医疗云服务器 CS(cloud server)和数据使用者 DU(data user).

(1) 医疗中心 HC:HC 作为完全可信的机构,拥有所有 EHRs 文件,负责对系统进行初始化并产生系统公开参数,选取密钥和加密算法,生成自身私钥.同时,HC 负责建立关键词索引,将文件的关键词索引加密,并将生成的文件密文和索引密文上传给医疗云服务器 CS.医疗中心 HC 提供验证数据使用者 DU 的注册信息,并将密钥发送给授权用户 DU.值得注意的是,系统初始化过程只做 1 次;

(2) 医疗云服务器 CS:负责数据处理,包括数据存储、计算、搜索以及重加密密文的生成与发送.在本文模型中,EHRs 被外包存储到医疗云服务器 CS 中,要求它是一种半诚实的云服务器模型,即“诚实且好奇(honest-but-curious)”的.虽然云服务器 CS 会如实执行用户的查询请求,返回正确的检索结果,但云服务器为了获得用户敏感数据去收集信息,并窥探用户查询的内容以及加密 EHRs 文件的各种信息;

(3) 数据用户 DU:根据自身查询请求生成搜索矢量,构建搜索陷门并向云服务器进行检索请求.

系统工作流程分为 6 个步骤.

1. 步骤 1:系统初始化、用户注册与关键词提取.

可信 HC 在初始化阶段随机生成一个 $l\times h$ 比特位的二进制划分矩阵 $\mathcal{P}=\{1,0\}_{l\times h}$,选取两个 $l\times l$ 的可逆矩阵 $\mathcal{M}_1$ 和 $\mathcal{M}_2$ 以及一个 n 位全部值为 1 的二进制参考向量 $\vec{Q}$,称为参考查询矢量.其中,l 是一个正整数且满足 $l|n$,h 表示划分段数 $h=n/l$.其中,私钥是一个三元组 $SK=\{\mathcal{P},\boldsymbol{M}_1,\boldsymbol{M}_2\}$.

为了保证密文的可搜索性,可信 HC 在初始化阶段从所有 EHRs 文件 $F=\{f_1,f_2,\ldots,f_m\}$中提取关键词集合:

$$W=(w_1,w_2,\ldots,w_n).$$

2. 步骤 2:EHRs 文件加密.

设医疗中心 HC 拥有 m 个医疗数据文件 HER,$F=\{f_1,f_2,\ldots,f_m\}$.为了减小数据存储以及使用开销,HC 选取对称加密方案$\varepsilon(E,D)$(如 AES 算法)对外包的 EHRs 文件加密成 $CT=(ct_1,ct_2,\ldots,ct_m)$.

3. 步骤 3:索引生成.

医疗中心 HC 根据提取的关键词集合 $W=(w_1,w_2,\ldots,w_n)$,为每个文件 f_i 加密密文的 ct_i 构建安全搜索索引 $\hat{I}_i'$,最后以结构 $(\hat{I}_i'\|ct_i)$ 的形式上传到云服务器 CS 中.

4. 步骤 4:陷门的生成.

4.1 用户授权:数据使用者 DU 在对医疗云上 EHRs 文件进行检索前,需要预先在医疗中心 HC 处注册.HC 根据 DU 注册信息决定是否授权.若 DU 得到授权,HC 将私钥 $SK=\{KEY,P,M_1,M_2\}$通过安全信道发送给 DU;

4.2 陷门生成:若 DU 得到搜索授权,DU 根据公开的关键词集合 $W=(w_1,w_2,\ldots,w_n)$构建搜索向量 $\vec{S}$,随后由私钥 $SK=\{P,M_1,M_2\}$和搜索向量 $\vec{S}$ 生成搜索陷门 TD_S,并发送给医疗云服务器 CS.

5. 步骤 5:陷门的匹配.

CS 收到搜索陷门 TD_S 后,将收到的搜索陷门 TD_S 与 EHRs 密文的安全索引 $\hat{I}_i'$ 做搜索匹配运算,将匹配的 EHRs 密文发送回 DU.

6. 步骤 6:搜索确认和解密.

数据用户 DU 收到匹配 EHRs 密文之后,通过解密算法 $D_{key_i}(f_i)$ 得到所要搜索的 EHRs 文件 f_i.

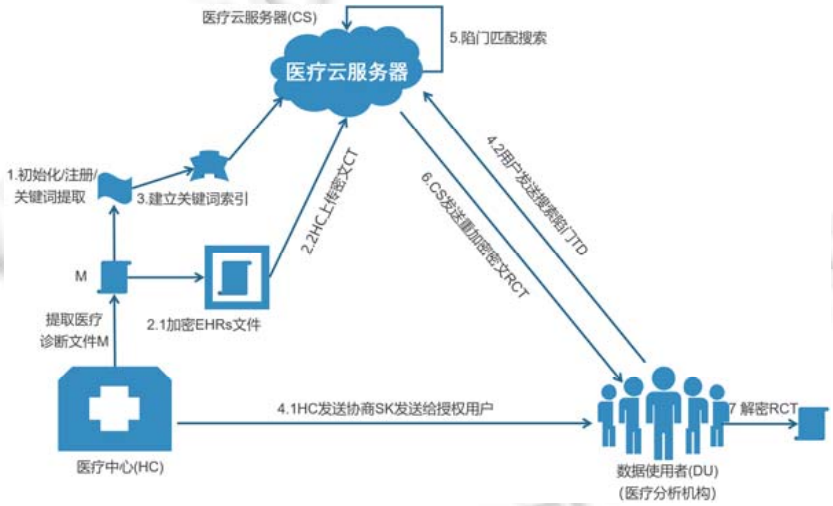

Fig.1 System model

图 1 系统模型

3.2 安全模型与设计目标

对于医疗系统,在安全需求中,我们主要考虑半诚实的医疗云服务器.

- 半诚实医疗云服务器.

选择外包的云服务器大多是半诚实的,这类云服务器会“诚实地”按照协议中的各项流程执行协议,也会按照索引标识符搜索对应的安全索引,完成授权用户的搜索请求后,也会如实返回结果.同时,这类云服务器又是“好奇的”,它会在执行协议过程中利用陷门、密文以及收集的其他信息窥探用户查询内容以及加密的 EHRs 文件信息,例如用户在云服务器上查询的具体内容以及用户提交的查询请求等.

本文方案可以达到如下目标.

(1) 基于范围的多关键词精确搜索:该方案能够针对多种关键词进行范围精确搜索,进行高效率搜索,能够在支持具有复杂查询结构的连接查询的同时,能够进行支持通配符的查询;

(2) 查询隐私安全:云服务器除了获取搜索结果的密文记录信息,不能得到其他有关 EHRs 属性以及用户关键词查询的信息;

(3) 搜索模式隐藏:搜索模式隐藏要求,云服务器在执行一个搜索请求不是由“相同的关键词构造”时,不能获得用户搜索请求有关的任何有用信息,即云服务器无法区分两个不同的用户搜索请求;

(4) 高效性:在相同关键词数量的情况下,方案具有较高的效率,适应于医疗大数据环境隐私保护下的范围搜索功能.

4 具体方案

4.1 属性矢量和查询矢量表示

本系统中,EHRs 中的每条记录均具有预先设定的属性(例如病人姓名、年龄、性别、病种、病因等),部分属性还都具有其子属性,这些子属性共同构成该 EHRs 的关键词集.为了满足 EHRs 密文的可搜索性,我们对这些属性编码成二进制矢量,以进行搜索匹配运算.

我们采用独热编码技术(one-hot coding),使用 N 位状态寄存器来对 N 个状态进行编码,每个状态都有其独立的寄存器位,并且在任意时候只有 1 位有效.

我们通过一个例子来详细说明本文属性矢量构建方式以及如何转化成属性矩阵.假定 EHRs 总共有 4 个属性(年龄、性别、疾病、地区),其中,年龄有 100 个属性值(1,2,...,100),性别有两个属性值(男,女),疾病假定有 3 个属性值(高血压,高血糖,高血脂),地区假定有 5 个属性值(北京,上海,武汉,深圳,成都).那么对于该 EHRs 而言,共有 110 个关键词.对于一位记录为(47,女,高血糖,北京)患者的 EHRs 属性,可由图 2 所示的二进制属性矢量进行转换和表示.

当某医疗组织 DU 希望查询“高血压在武汉 40 多岁人群”中的患病情况时,则该构造生成查询请求:

“年龄=4* AND 疾病=高血压 AND 地区=武汉”.

查询请求中对于性别并不关心,该属性用通配符“*”代替,查询中的范围查询“40 多岁”(40~49 岁之间)利用通配符“4*”来构造.上述查询的查询向量和查询矩阵转换构造如图 3 所示.

Fig.2 Index vector and martix

图 2 索引矢量和索引矩阵

Fig.3 Search vector and martix

图 3 查询矢量与查询矩阵

本文第 2 节所述 APSE 方案,实现了查询矢量 $\vec{Q}$ 和属性矢量的内积.为实现更快的系统构建以及匹配效率,将原有 APSE 方案由保证两个向量的内积拓展到两个矩阵的哈达码积,其正确性在第 4.8 节中进行详细论证.

本文实现查询矩阵构造方式如图 4 所示,其范围搜索的方式如图 5 所示.为实现范围搜索,只需在构建查询向量时对在范围内的属性位置 1,查询矩阵由差分矩阵处理后构建查询矩阵该属性值置 0.因而与含有该属性的索引矩阵的哈达码积为 0,即能够实现对所有需要属性的范围搜索功能.

(1010011100 1000110010 .... 1010110011 1000110001)

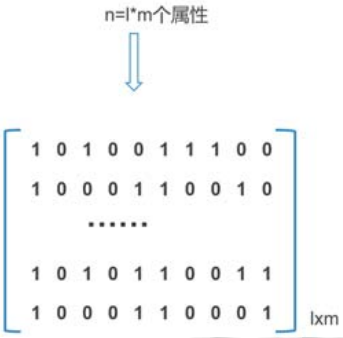

Fig.4 Construction of keyword search matrix

图 4 查询关键词矩阵构造

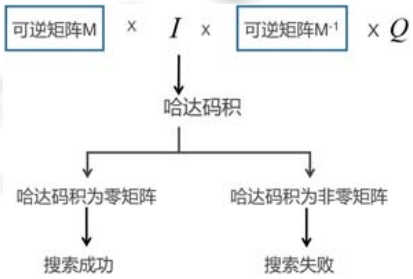

Fig.5 Match for range search procedure

图 5 范围搜索匹配

在图 3 所述实例中,对于 46 岁女患者而言,年龄 46 属性值为 1,性别属性值为 01.按照所构建的查询矩阵年龄 40~50 范围内属性值均为 0,性别属性值均为 0,最后,哈达码积计算得到的矩阵也为 0 矩阵,即搜索成功.

若年龄为 39 岁的患者,其年龄对应 39 属性值为 1,最终查询矩阵在年龄 39 上的属性值也为 1,哈达码积计算得到的矩阵在对应年龄 39 属性上的值不为 0,即哈达码积结果为非零矩阵.因此该病例并不会被本文算法搜索得到,即该病例搜索失败.对于该例子而言,只要属性符合我们的查询请求“(年龄=4*) AND (疾病=高血压) AND (地区=武汉)”的病例均会被成功搜索得到,即实现了范围搜索能力.

设云医疗系统中共有 m 个文件和 n 个关键词,接下来我们给出具体构造方案.

4.2 系统初始化Setup

可信健康中心 HC 对文件 F 中提炼关键词总集(如年龄、地区、病种等),记为 $W=(w_1,w_2,\ldots,w_n)$,生成 $l\times h$ 位二进制矩阵 $P=\{1,0\}_{l\times h}$,选取两个 $l\times l$ 的可逆矩阵 M_1 和 M_2 以及一个 n 位初始值为 1 的二进制参考向量 $\vec{Q}$,称为参考查询矢量.其中,l 是一个正整数且满足 $l|n,h$ 表示划分段数,$h=n/l$.

同时,对于医疗数据 EHRs 文件 $F=(f_1,f_2,\ldots,f_m)$,HC 选择加密文件 F 的对称加密算法.

最后,HC 公开参数 $pp=\{n,\vec{Q},W,\varepsilon(E,D)\}$.

保存 HC 的系统私钥 $SK=\{KEY,P,\boldsymbol{M}_1,\boldsymbol{M}_2\}$.

4.3 EHRs文件加密Enc

医疗中心 HC 对所拥有的 EHRs 文件集合 $F=(f_1,f_2,\ldots,f_m)$,利用对称密钥 $KEY=(key_1,key_2,\ldots,key_m)$调用对称加密算法,将文件加密成 $CT=(ct_1,ct_2,\ldots,ct_m)$.

4.4 索引生成$Index_{SK}(F)$

HC 对所有 EHRs 文件 F 提炼关键词总集 $W=(w_1,w_2,\ldots,w_n)$.不失一般性,对第 k 个病人的 EHRs 文件 f_k,调用算法 1 生成关键词索引矩阵 $\hat{\boldsymbol{I}}_k$.

将关键词总集每 l 个划分为一段,即:

$$W_1=(w_1,w_2,\ldots,w_l),W_2=(w_{l+1},w_{l+2},\ldots,w_{2l}),\ldots,W_h=(w_{h+1},w_{h+2},\ldots,w_{h+n}).$$

最后生成 $h=n/l$ 段,关键词总集可以表示为 $\boldsymbol{W}=(W_1,W_2,\ldots,W_h)$.

算法 1. 索引生成算法.

输入:关键词总集 $W=(w_1,w_2,\ldots,w_n)$,

EHRs 文件 $F=(f_1,f_2,\ldots,f_m)$,

HC 私钥中的$\{P,M_1,M_2\}$;

输出:文件索引 $\hat{\boldsymbol{I}}'$.

BEGIN:

FOR EACH $x\in[1,n]$, $w_x\in W$;

1. 对关键词总集 W 进行划分,生成矩阵 $\boldsymbol{W}=(W_1,W_2,\ldots,W_h)$;
2. 对于 $k\in[1,m],f_k\in F$;

以 $\boldsymbol{W}$ 和 f_k 关键字生成关键字矩阵:

FOR EACH $j\in[1,h]$;

FOR EACH $i\in[1,l]$;

IF 关键词存在;

该位置 1;

ELSE

该位置 0;

生成矩阵 $\hat{\boldsymbol{I}}_k$;

随机选取 $\boldsymbol{A}=|\omega_{i,j}|_{l\times h}$;

计算 $\boldsymbol{I}'_k=\boldsymbol{A}*\hat{\boldsymbol{I}}_k$;

划分矩阵 $\boldsymbol{P}$ 划分 $\hat{\boldsymbol{I}}_k$ 为 $\boldsymbol{I}'_{k,a}[i,j]$ 和 $\boldsymbol{I}'_{k,b}[i,j]$:

对 $\boldsymbol{P}$ 的第 i 行第 j 列 $\boldsymbol{P}[i,j]$;

IF $\boldsymbol{P}[i,j]=1$;

$\boldsymbol{I}'_{k,a}[i,j]+\boldsymbol{I}'_{k,b}[i,j]=\boldsymbol{I}'_k[i,j]$;

ELSE $\boldsymbol{P}[i,j]=0$;

$\boldsymbol{I}'_{k,a}[i,j]=\boldsymbol{I}'_{k,b}[i,j]=\boldsymbol{I}'_k[i,j]$;

END FOR;

END FOR;

根据$\{\boldsymbol{M}_1,\boldsymbol{M}_2\}$计算: $\hat{I}'_k=\{\boldsymbol{M}_1\boldsymbol{I}'_{k,a},\boldsymbol{M}_2\boldsymbol{I}'_{k,b}\}$

END;

对文件 f_k,用 $\boldsymbol{I}'_k[i,j]$ 表示第 i 行第 j 列关键字是否存在($i\in[1,l],j\in[1,h]$):若 $\boldsymbol{I}'_k[i,j]=1$,则对应位置的关键字存在;若 $\boldsymbol{I}'_k[i,j]=0$,则对应的第 i 行第 j 列位置的关键字不存在.将第 k 个 EHRs 文件具有的关键词属性通过该方

式表示成为 $l\times h$ 的(0,1)矩阵 $\hat{\boldsymbol{I}}_k$.

HC 选取随机矩阵 $\boldsymbol{A}=[\omega_{i,j}]_{i=1,\ldots,l,j=1,\ldots,h}$,计算 $\boldsymbol{A}$ 和索引矩阵 $\hat{\boldsymbol{I}}_k$ 的哈达马积,即 $\boldsymbol{I}'_k=\boldsymbol{A}*\hat{\boldsymbol{I}}_k$,其中,$\omega_{i,j}\neq 0$.令 $\boldsymbol{P}[i,j]$ 表示二进制矩阵 $\boldsymbol{P}$ 的第 i 行第 j 列,对于($i\in[1,l]$,$j\in[1,h]$):若 $\boldsymbol{P}[i,j]=1$,则将 $\boldsymbol{I}'_k[i,j]$ 随机划分成 $\boldsymbol{I}'_{k,a}[i,j]$ 和 $\boldsymbol{I}'_{k,b}[i,j]$,且满足 $\boldsymbol{I}'_{k,a}[i,j]+\boldsymbol{I}'_{k,b}[i,j]=\boldsymbol{I}'_k[i,j]$;若 $\boldsymbol{P}[i,j]=0$,则置 $\boldsymbol{I}'_{k,a}[i,j]=\boldsymbol{I}'_{k,b}[i,j]=\boldsymbol{I}'_k[i,j]$ 构造方法如图 6 所示.

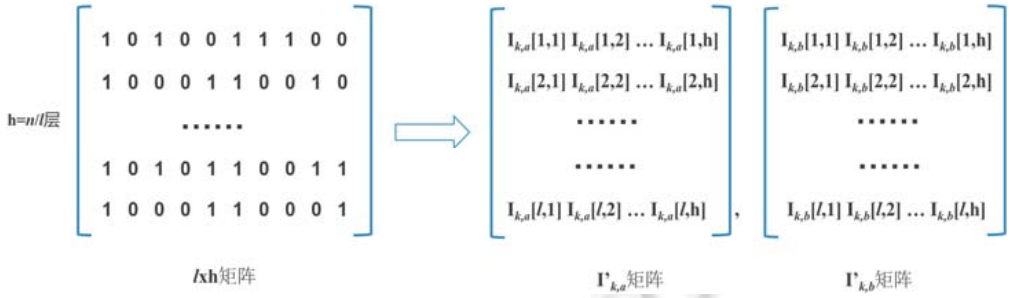

Fig.6 Division of index matrix

图 6 索引矩阵的划分

利用$\{\boldsymbol{M}_2\}$生成加密索引矩阵 $\hat{\boldsymbol{I}}'_k:\hat{I}'_k=\{\boldsymbol{M}_1\boldsymbol{I}'_{k,a},\boldsymbol{M}_2\boldsymbol{I}'_{k,b}\}$.

最后,将 $(\hat{I}'\|CT)$ 上传至云服务器 CS.

4.5 搜索陷门生成

4.5.1 搜索向量构造

数据使用者 DU 要搜索和访问云服务器 CS 上加密 EHRs 文件时,根据关键词信息,在构建搜索向量 $\vec{S}$ 时,对于包含的关键词,将其置 1,构建差分搜索向量 $\vec{S}'$ 在该位置 0.对本方案,匹配算法 4 能够搜索到所需文件.

例如,搜索年龄在 20~49 岁、患有高血压的男性患者的病历,如图 7 所示.医疗组织由该方法构建搜索向量,在文件索引和搜索陷门匹配阶段,以该例具体分析如何实现范围搜索.

Fig.7 Construction instance: search vector with “male, hypertension, age between 20 and 49”

图 7 构造实例:查询向量“年龄在 20~49 岁、患有高血压的男性病人”

4.5.2 陷门 $\text{Trapdoor}_{SK}(S)$生成

DU 向医疗中心 HC 请求搜索授权,医疗中心 HC 根据授权规则决定是否授权.数据使用者 DU 获得了授权后,HC 通过安全信道将密钥 SK 发送给 DU,DU 执行以下步骤获得搜索陷门 TD_S(如算法 2 所示).

1. DU 根据搜索向量 $\vec{S}$, 利用参考向量 $\vec{Q}$ 计算搜索差分向量 $\vec{S}'=\vec{Q}-\vec{S}$;
2. DU 将 $\vec{S}'$ 进行划分,即 $\vec{S}'_1=(s'_1,s'_2,...,s'_l),\vec{S}'_2=(s'_{l+1},...,s'_{2l}),...,\vec{S}'_{n/l}$ 共 h 段,搜索差分向量表示为
$$\boldsymbol{S}=(\vec{S}'_1,\vec{S}'_2,...,\vec{S}'_h)^T.$$
同样方式将 $\boldsymbol{S}$ 排列为 $l\times h$ 的矩阵 $\hat{\boldsymbol{S}}$;
3. 用 $\hat{\boldsymbol{S}}[i,j]$ 表示第 i 行第 j 列是否为本次需要搜索的关键词($i\in[1,l],j\in[1,h]$):若 $\hat{\boldsymbol{S}}[i,j]=1$, 则对应第 i 行第 j 列位置的关键字是不需要搜索的;若 $\hat{\boldsymbol{S}}[i,j]=0$, 则对应的第 i 行第 j 列位置的关键字是需要搜索的. 最后输出差分搜索矩阵 $\hat{\boldsymbol{S}}$;
4. DU 选取随机矩阵 $\boldsymbol{B}=|\tau_{i,j}|_{l\times h}$,计算矩阵 $\boldsymbol{B}$ 与差分搜索矩阵 $\hat{\boldsymbol{S}}$ 的哈达马积,即 $\hat{\boldsymbol{S}}'=\boldsymbol{B}*\hat{\boldsymbol{S}}$. 这里,$\tau_{i,j}\neq 0$.随后,根据收到的密钥 SK 构造搜索陷门.令 $\boldsymbol{P}[i,j]$表示二进制矩阵 $\boldsymbol{P}$ 的第 i 行第 j 列,对($i\in[1,l],j\in[1,h]$):若 $\boldsymbol{P}[i,j]=1$,将 $\hat{\boldsymbol{S}}[i,j]$ 划分为 $\hat{\boldsymbol{S}}_a[i,j]$ 和 $\hat{\boldsymbol{S}}_b[i,j]$, 且满足 $\hat{\boldsymbol{S}}[i,j]=\hat{\boldsymbol{S}}_a[i,j]=\hat{\boldsymbol{S}}_b[i,j]$; 若 $\boldsymbol{P}[i,j]=0$,则将 $\hat{\boldsymbol{S}}[i,j]$ 置为 $\hat{\boldsymbol{S}}[i,j]=\hat{\boldsymbol{S}}_a[i,j]+\hat{\boldsymbol{S}}_b[i,j]$. 结合$\{\boldsymbol{M}_1,\boldsymbol{M}_2\}$生成陷门 TD_S, $TD_S=\{\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}_a,\boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}_b\}$;
5. DU 将搜索陷门 TD_S 发送给云服务器 CS 请求搜索.

算法 2. 陷门生成算法.

输入:搜索向量 $\vec{S}$, 参考向量 $\vec{Q}$, HC 私钥中的$\{P,M_1,M_2\}$;

输出:搜索陷门 TD_S.

开始:

1. 计算搜索差分向量 $\vec{S}'=\vec{Q}-\vec{S}$;
2. 对于搜索差分向量 $\vec{S}'$:

   将 $\vec{S}'$ 构建成矩阵 $\boldsymbol{S}=(\vec{s}'_1,\vec{s}'_2,...,\vec{s}'_h)^T$

   得到差分搜索矩阵 $\hat{\boldsymbol{S}}$;

   FOR EACH $j\in[1,h]$;

   FOR EACH $i\in[1,l]$;

   选取随机矩阵 $\boldsymbol{B}=|\tau_{i,j}|_{l\times h}$;

   计算 $\boldsymbol{S}'=\boldsymbol{B}*\hat{\boldsymbol{S}}$;

   划分矩阵 $\boldsymbol{P}$ 将矩阵 $\hat{\boldsymbol{S}}$ 划分为 $\hat{\boldsymbol{S}}_a[i,j]$ 和 $\hat{\boldsymbol{S}}_b[i,j]$:

   对于 $\boldsymbol{P}$ 的第 i 行第 j 列 $\boldsymbol{P}[i,j]$;

   IF $\boldsymbol{P}[i,j]=1$;

   $\hat{\boldsymbol{S}}[i,j]=\hat{\boldsymbol{S}}_a[i,j]=\hat{\boldsymbol{S}}_b[i,j]$;

   ELSE $\boldsymbol{P}[i,j]=0$;

   $\hat{\boldsymbol{S}}[i,j]=\hat{\boldsymbol{S}}_a[i,j]+\hat{\boldsymbol{S}}_b[i,j]$;

   END FOR;

   END FOR;
3. 计算: $TD_S=\{\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}_a,\boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}_b\}$;

END

4.6 索引与陷门匹配 *Query*($TD_S,\hat{\boldsymbol{I}}'$)

云服务器 CS 执行匹配查询,以索引矩阵 $\hat{\boldsymbol{I}}'$ 和搜索陷门 TD_S 为输入,计算:

$$\hat{I}'*TD_S=\{\boldsymbol{M}_1\hat{\boldsymbol{I}}'_a,\boldsymbol{M}_2\hat{\boldsymbol{I}}'_b\}*\{\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}_a,\boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}_b\}=(\boldsymbol{M}_1\hat{\boldsymbol{I}}'_a)*(\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}_a)+(\boldsymbol{M}_2\hat{\boldsymbol{I}}'_b)*(\boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}_b)=\hat{\boldsymbol{I}}'_a*\hat{\boldsymbol{S}}_a+\hat{\boldsymbol{I}}'_b*\hat{\boldsymbol{S}}_b,$$

其中,“*”表示两个矩阵的哈达马积.

当且仅当计算所得的矩阵为 0 矩阵时,云服务器 CS 才将对应密文 CT'发送给 DU.

注解 1:

(1) 根据本方案可知:对不需要的搜索关键词,其搜索差分矩阵在该位置 1,一旦某文件在该关键词上的值也为 1 时,该文件的索引和此次搜索陷门进行匹配计算的矩阵不为 0 矩阵,对于包含不需要关键词的文件将不会被检索;

(2) 对于包含需要的关键词、不含不需检索关键词的文件,可由通配符描述要搜索文件时,根据独热编码原理,将通配符代表位置的关键词均置 1,即搜索差分矩阵在这些关键词的值均为 0.即:若某文件索引该位置值为 1,该文件的索引和此次搜索陷门计算的矩阵也为 0 矩阵,即该文件仍然会被搜索得到;

(3) 由于构建的文件索引和搜索陷门中包含文件的所有属性,只需将搜索向量中需要搜索的关键词置 1,就能实现范围搜索.

4.7 密文解密

DU 收到匹配的密文 ct_i 后,使用对称解密算法$\varepsilon(E,D)$和相应的解密密钥 key_i 将 ct_i 解密成所需要的 EHRs 文件 f_i.

5 方案正确性

为描述检索匹配算法的正确性,用一个实例来验证陷门匹配的正确性.假定 l=3,h=4,私钥中二进制矩阵:

$$\boldsymbol{P}=\begin{bmatrix}1&0&0&1\\0&0&1&1\\1&0&1&0\end{bmatrix}.$$

对于某个需要搜索的文件 f_k,其关键词向量假定为 $\vec{I}$ =[101001010011];

用户搜索向量为 $\vec{S}$ =[101001010111];

那么,搜索差分向量 $\vec{Q}-\vec{S}$ =[010110101000];

可以得到, $\vec{I}\cdot(\vec{Q}-\vec{S})$ 向量内积为 0,此用户可能成功检索到文件 f.

根据本方案, $I'=A*\hat{I}$, 建立的关键词矩阵为

$$\boldsymbol{I}'=\begin{bmatrix}\omega_{1,1}&0&\omega_{1,3}&0\\0&\omega_{2,1}&0&\omega_{2,4}\\0&0&\omega_{3,3}&\omega_{3,4}\end{bmatrix},$$

则以私钥 $\boldsymbol{P}$ 通过索引生成算法 $Index_{SK}(F)$划分出的 $\boldsymbol{I}'_a$、$\boldsymbol{I}'_b$ 的索引矩阵为

$$\boldsymbol{I}'_a=\begin{bmatrix}a_1&0&\omega_{1,3}&a_4\\0&\omega_{2,1}&a_7&a_8\\a_9&0&a_{11}&\omega_{3,4}\end{bmatrix},\boldsymbol{I}'_b=\begin{bmatrix}\omega_{1,1}-a_1&0&\omega_{1,3}&-a_4\\0&\omega_{2,1}&-a_7&\omega_{2,4}-a_8\\-a_9&0&\omega_{3,3}-a_{11}&\omega_{3,4}\end{bmatrix}.$$

以$\{\boldsymbol{M}_1,\boldsymbol{M}_2\}$生成加密的索引矩阵为 $\hat{I}'=\{\boldsymbol{M}_1\boldsymbol{I}'_a,\boldsymbol{M}_2\boldsymbol{I}'_b\}$.

用户 DU 使用搜索向量 $\vec{S}$ 请求搜索时,首先调用陷门生成算法 $Trapdoor_{SK}(S)$划分出来搜索陷门 $\hat{\boldsymbol{S}}_a$、$\hat{\boldsymbol{S}}_b$ 为

$$\hat{\boldsymbol{S}}_a=\begin{bmatrix}0&b_2&b_3&\tau_{1,4}\\b_5&b_6&\tau_{2,3}&0\\\tau_{3,1}&b_9&0&b_{12}\end{bmatrix},\hat{\boldsymbol{S}}_b=\begin{bmatrix}0&\tau_{1,2}-b_2&-b_3&\tau_{1,4}\\\tau_{2,1}-b_5&-b_6&\tau_{2,3}&0\\\tau_{3,1}&\tau_{3,2}-b_9&0&-b_{12}\end{bmatrix}.$$

以密钥$(\boldsymbol{M}_1,\boldsymbol{M}_2)$生成陷门 TD_S,并将 TD_S 发送给云服务器 CS.

云服务器以陷门 $TD_S=\{\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}_a,\boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}_b\}$ 和搜索索引 $\hat{I}'=\{\boldsymbol{M}_1^T\hat{\boldsymbol{I}}'_a,\boldsymbol{M}_2^T\hat{\boldsymbol{I}}'_b\}$, 通过匹配算法判断:

$$\hat{I}'*TD_S=\{\boldsymbol{M}_1\hat{\boldsymbol{I}}'_a,\boldsymbol{M}_2\hat{\boldsymbol{I}}'_b\}*\{\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}_a,\boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}_b\}=(\boldsymbol{M}_1\hat{\boldsymbol{I}}'_a)*(\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}_a)+(\boldsymbol{M}_2\hat{\boldsymbol{I}}'_b)*(\boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}_b)=\hat{\boldsymbol{I}}'_a*\hat{\boldsymbol{S}}_a+\hat{\boldsymbol{I}}'_b*\hat{\boldsymbol{S}}_b.$$

对文件 f 的$(\hat{\boldsymbol{I}}'_a,\hat{\boldsymbol{I}}'_b)$以及用户搜索陷门的$(\hat{\boldsymbol{S}}_a,\hat{\boldsymbol{S}}_b)$执行关键词搜索运算,算法输出结果如下:

$$\hat{\boldsymbol{I}}'_a * \hat{\boldsymbol{S}}_a = \begin{bmatrix} 0 & 0 & b_3\omega_{1,3} & a_4\tau_{1,4} \\ 0 & b_6\omega_{2,1} & a_7\tau_{2,3} & 0 \\ a_9\tau_{3,1} & 0 & 0 & b_{12}\omega_{3,4} \end{bmatrix}, \hat{\boldsymbol{I}}'_b * \hat{\boldsymbol{S}}_b = \begin{bmatrix} 0 & 0 & -b_3\omega_{1,3} & -a_4\tau_{1,4} \\ 0 & -b_6\omega_{2,1} & -a_7\tau_{2,3} & 0 \\ -a_9\tau_{3,1} & 0 & 0 & -b_{12}\omega_{3,4} \end{bmatrix}.$$

显然, $\hat{\boldsymbol{I}}'_a * \hat{\boldsymbol{S}}_a + \hat{\boldsymbol{I}}'_b * \hat{\boldsymbol{S}}_b$ 值为 0 矩阵,则检索成功.

6 安全性分析

本方案的应用场景为面向医疗云中可搜索加密,我们主要考虑半诚实医疗云服务器.对于外包半诚实云服务器,云服务器会“诚实地”按照协议的各项流程执行协议,也会按照索引标识符找到对应的安全索引,完成授权用户搜索请求后,会如实地返回结果.同时,这类云服务器又是“好奇的”,通过陷门以及收集信息去窥探用户查询的内容以及加密 EHRs 文件的各种信息.

6.1 搜索模式的隐藏

方案在半诚实医疗云中可以实现搜索模式的隐藏:给定搜索向量 $\vec{S}$, 数据拥有者生成两次搜索陷门 TD_S 以及 TD'_S, 云服务器无法区分 TD_S 和 TD'_S, 证明云服务器在获得“两个搜索是由相同的关键词生成”的前提下而无法获取的任何知识,即实现了搜索模式的隐藏,证明过程如下.

1. 数据拥有者通过搜索向量 $\vec{S}$ 计算差分向量 $\vec{S}' = \vec{Q} - \vec{S}$, 将 $\vec{S}'$ 排列成为一个 $l \times h$ 的矩阵 $\boldsymbol{S}$;
2. 数据拥有者选取两个随机矩阵 $\boldsymbol{B}=[\tau_{i,j}]_{l\times h}$ 以及 $\boldsymbol{B}' = [\tau'_{i,j}]_{l\times h}$, 计算 $\boldsymbol{B}$ 与 $\boldsymbol{S}$ 以及 $\boldsymbol{B}'$与 $\boldsymbol{S}$ 的哈达马积:
$$\hat{\boldsymbol{S}} = \boldsymbol{B} * \boldsymbol{S}, \hat{\boldsymbol{S}}' = \boldsymbol{B}' * \boldsymbol{S};$$
3. 数据拥有者使用划分矩阵 $\boldsymbol{P}$ 对 $\hat{\boldsymbol{S}}$ 和 $\hat{\boldsymbol{S}}'$ 进行划分,生成 $\hat{\boldsymbol{S}}_a[i, j]$ 和 $\hat{\boldsymbol{S}}_b[i, j]$ 以及 $\hat{\boldsymbol{S}}'_a[i, j]$ 和 $\hat{\boldsymbol{S}}'_b[i, j]$. 最后生成搜索陷门 $TD(S) = (\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}_a, \boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}_b)$ 以及 $TD'(S) = (\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}'_a, \boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}'_b)$.

对于云服务器而言,对 $TD(S)$以及 $TD'(S)$执行匹配算法:

$$\hat{I}' * TD = (\boldsymbol{M}_1\hat{\boldsymbol{I}}'_a, \boldsymbol{M}_2\hat{\boldsymbol{I}}'_b) * (\boldsymbol{M}_1^{-1}\hat{\boldsymbol{S}}_a, \boldsymbol{M}_2^{-1}\hat{\boldsymbol{S}}_b).$$

由方案正确性分析可知:$TD(S)$与 $TD'(S)$在执行匹配运算后具有相同的查询结果,即匹配文件运算中 $\hat{I}' * TD$ 为 0 矩阵,云服务器无法从匹配算法中获得任何有用的区分信息.若云服务器仍能区分不同检索向量的陷门 $TD(S)$和 $TD'(S)$,则云服务器不仅能区分 $\boldsymbol{B}$ 和 $\boldsymbol{B}'$,同时能够区分划分矩阵 $\boldsymbol{P}$ 对 $\hat{\boldsymbol{S}}$ 和 $\hat{\boldsymbol{S}}'$ 的随机划分.而方案中,矩阵 $\boldsymbol{B}$ 和 $\boldsymbol{B}'$均是随机选取的,矩阵 $\boldsymbol{B}$ 每一位置上元素的取值范围为[1,u],而查分矩阵中值为 1 的数量为 v,云服务器区分 $\boldsymbol{B}$ 和 $\boldsymbol{B}'$的概率为

$$p_1 = \frac{1}{u^v}.$$

假定划分矩阵 $\boldsymbol{P}$ 中值为 1 的个数为η,则服务器区分划分矩阵 $\boldsymbol{P}$ 对 $\hat{\boldsymbol{S}}$ 和 $\hat{\boldsymbol{S}}'$ 随机划分的概率为

$$p_2 = \frac{1}{2^\eta}.$$

最后,服务器区分 $TD(S)$和 $TD'(S)$的概率为

$$P = p_1 p_2 = \frac{1}{u^v 2^\eta}.$$

可以得到,云服务器能够区分 $TD(S)$和 $TD'(S)$取决于随机矩阵 $\boldsymbol{B}$ 中每个位置的取值范围以及差分矩阵和划分矩阵中 1 的个数.对于一个关键词总数为 n 的搜索系统而言,服务器直接猜测两次搜索是否来自同一个查询成功的概率为 $P'=1/(2^n)$.

在本方案中,我们假定随机矩阵 $\boldsymbol{B}$ 每个位置上的取值范围为[1,2],而我们需要搜索关键词总数一半的 EHRs 文件,则 $p_1=1/(2^{n/2})$.由于划分矩阵是随机生成的[0,1]矩阵,随机选取的值为 0 或 1 的概率相同,其中,为 1 的个数 $\eta \approx n/2$,即约为关键词总数 n 的一半,$p_2=1/(2^{n/2})$.在这种最小假设下,服务器成功区分的概率为 $P=p_1p_2=1/(2^n)$.可以得到:在最小假设下,本文方案中服务器成功区分的概率 P 与服务器直接猜测的概率 P'相同.

一般而言,作为范围搜索,一次搜索请求中需搜索的关键词个数远远小于不需要搜索关键词的个数.因此,

差分矩阵中 1 的个数远大于 0 的个数.同时,随机矩阵的取值范围也远远大于[1,2].在本文方案中,服务器成功区分的概率 $P<P'$.因此,云服务器不能区分 $TD(S)$和 $TD'(S)$,因此,该方案能够有效实现搜索模式的隐藏.

6.2 查询隐私安全

查询隐私安全要求云服务器 CS 无法根据自身构造的陷门来获取存储在云服务器端医疗 EHRs 密文的属性信息,能够抵抗已知明文攻击.

- 陷门与搜索索引间匹配运算过程中的已知明文攻击安全性

假定攻击者向 EHRs 数据库中插入一组攻击者已知的明文 I 并得到对应陷门.在不引入随机数的情况下,对于任何矢量 $\vec{I}_i \in I$,攻击者不知道加密值 $\boldsymbol{I}'_{k,a}[i,j]$ 和 $\boldsymbol{I}'_{k,b}[i,j]$. 本文模型中,攻击者并不知道拆分二进制矩阵 $\boldsymbol{P}$,将 $\boldsymbol{I}'_{k,a}[i,j]$ 和 $\boldsymbol{I}'_{k,b}[i,j]$ 模型划分为两个随机的 $l\times h$ 大小的矩阵,以求解 $\hat{\boldsymbol{I}}'_k=\{\boldsymbol{M}_1\boldsymbol{I}'_{k,a},\boldsymbol{M}_2\boldsymbol{I}'_{k,b}\}$. 其中,$(\boldsymbol{M}_1,\boldsymbol{M}_2)$是两个 $l\times l$ 的可逆矩阵.解方程个数为 $2l\times h$,而 $\boldsymbol{I}'_{k,a}[i,j]$ 和 $\boldsymbol{I}'_{k,b}[i,j]$ 中有 $2l\times h$ 个未知数,在$(\boldsymbol{M}_1,\boldsymbol{M}_2)$中有 $2l^2$ 个未知数.本方案 l 取值满足 $l>h$,即 $2l^2$ 大于 $2l\times h$,攻击者无法求解出置换矩阵,因此在搜索过程中可以抵御已知明文攻击,保证了查询隐私的安全性.

7 性能分析

目前,主要的可搜索加密 SE 方案大致分为两大类:基于公钥 PKC 的 SE 方案和基于对称密钥 SKC 的 SE 方案.在能够支持关键词连接查询的方案中,文献[28]是基于 PKC 的方案,文献[5]和文献[8]是同样使用了非对称向量积保持加密 SKC 的方案.由于双线性的方案效率较低,本节中我们不再考虑双线性对构造陷门和关键词匹配的基于 PKC 方案(文献[28]),仅与同样使用了 ASPE 方案的文献[5]和文献[8]作性能比较.

对于关键词总量 n 的可搜索加密系统,用 t_a 表示一次点加运算所用时间,t_m 表示一次点乘所用时间,T_d 表示生成一个长度为 d 的二进制向量所用时间,$T_{i\times j}$ 表示生成大小为 $i\times j$ 的矩阵所用时间,$\alpha_{i\times j}$ 表示大小为 $i\times j$ 的矩阵做转置所用时间,$\beta_{i\times j}$ 大小为 $i\times j$ 的矩阵做逆所用时间.方案计算性能的理论分析见表 2.

Table 2 Comparison of time complexity (n: number of keywords)

表 2 方案时间复杂度(n:关键词数)

阶段	MCKS-I[5]	YLC18[8]	本文方案
初始化	$T_n+2T_{n\times n}$	$T_n+2h\cdot T_{l\times l}$	$T_n+2T_{l\times l}$
索引生成	$2n^2(t_a+t_m)+2\alpha_{n\times n}$	$2h^2\cdot l(t_a+t_m)+2h\cdot\alpha_{l\times l}$	$2h\cdot l(t_a+t_m)+2\alpha_{l\times l}$
陷门生成	$2n^2(t_a+t_m)+2\beta_{n\times n}$	$2h^2\cdot l(t_a+t_m)+2h\cdot\beta_{l\times l}$	$2h\cdot l(t_a+t_m)+2\beta_{l\times l}$
查询检索	$2n(t_a+t_m)+t_a$	$2h\cdot l(t_a+t_m)+h\cdot t_a$	$2h\cdot l(t_a+t_m)+t_a$

在实际测试中,运行在 Window 7 上 64bit 台式机进行,处理器为 Intel(R) G3240,CPU 主频率为 3.10GHz,4G 内存 RAM.实验实现基于 Java 语言,版本为 1.80.非对称向量积保持加密方案实验仿真基于 JAMA 矩阵运算库编写.

本方案实验中,l 的取值会对实验结果产生影响,其主要影响私钥$(\boldsymbol{P},\boldsymbol{M}_1,\boldsymbol{M}_2)$的生成以及矩阵运算时矩阵的大小.当 l 取值增大时,生成私钥$(\boldsymbol{P},\boldsymbol{M}_1,\boldsymbol{M}_2)$以及矩阵的哈达马积运算的计算开销就会增大.随着关键词数量 n 的增长,ASPE 方案构建矩阵大小也会增大,导致文献[5,8]和我们的方案所需时间也相应增加.

7.1 系统初始化性能

在 MCKS-II 方案[5]初始化过程中,需产生两个 $n\times n$ 的可逆矩阵$(\boldsymbol{M}_1,\boldsymbol{M}_2)$及所需 n 位二进制矢量 S,同时需预计算 $\boldsymbol{M}_1^T$、$\boldsymbol{M}_2^T$、$\boldsymbol{M}_1^{-1}$ 和 $\boldsymbol{M}_2^{-1}$.在初始化阶段,主要运算开销是矩阵的生成、计算矩阵转置和矩阵求逆,时间复杂度为 $O(n^3)$.文献[8]在初始化过程中构造 h 个可逆矩阵$(\boldsymbol{M}_1,\boldsymbol{M}_2)$,大小为 $l\times l$,产生 n 位二进制向量 S,同时需预计算 $\boldsymbol{M}_1^T$ 和 $\boldsymbol{M}_2^T$.本文方案在初始化阶段需构造可逆矩阵$(\boldsymbol{M}_1,\boldsymbol{M}_2)$,大小为 $l\times l$,产生一个 $l\times h$ 大小的二进制矩阵 $\boldsymbol{P}$.本文方案和文献[5]的 MCKS_II 方案在初始化时间上大大优于文献[8]的方案,在文件属性 n=1000 时,MCKS_II 所用时间比稍显更优(见表 3).

Table 3 Time of initialization phase (ms)

表 3 初始化时间 (毫秒)

关键词数	MCKS-II[5]	YLC18[8]	本文方案
250	44.142	330.279	32.209
500	48.413	418.825	39.465
750	51.988	487.392	40.756
1 000	60.940	570.445	42.549

MCKS-II 初始化效率与树形图的构建有关,假定文献[5]树形图有 v 层,每层树的节点有 n_0 个子节点,其属性值长度 $n_1=1+(v-1)n_0$.EHRs 属性值分别取值 $n=(750,1000,1250,1500)$,文献[5]方案中子节点个数 $n_0=10$,树层数 $v=4$.那么属性值长度 $n_1=31$,可逆矩阵(M_1,M_2)大小为 $n_1\times n_1$.MCKS-II 方案中树的最大容量 $n=1000$,在 $n=750$ 后,MCKS-II 方案相比本文方案在初始化时间上更优.但考虑到现实场景,并非所有属性值都能有 10 个子属性(如性别属性只有 2 个子属性),因此文献[5]中 MCKS-II 方案所假定 $n_1=31$ 理想化的树是不完全存在的.由文献[5]的性能分析得知,将 MCKS-I 方案中置属性值 $n=302$.根据树形结构转换 $n_1=56$,初始化时间为 44.142ms.文献[5]构造的 $n=302$ 树的初始化时间比本文方案在 $n=500$ 时所需时间更长.

后续实验中,根据文献[5]在性能分析中所提 MCKS-I 方案中属性值 $n=302$,根据树形结构转换 $n_1=56$,根据 $n_1=56n/302$ 转换比例来对 MCKS-II 方案进行测试实验.

考虑到非对称向量积保持加密方案的安全性,其中,l 大小应大于 h,在属性值 n 的取值上,保证索引安全的属性值有 $n=(750,1000)$.在属性值 $n=750$ 时,本方案时间为 40.756ms,文献[8]初始化时间为 487.392ms,本文方案比文献[8]方案优化约 12 倍.随着属性值 n 增加到 $n=1000$,本文方案比文献[8]快约 15 倍.随着属性值 n 的增大,本文方案在系统初始化效率上会比文献[5]和文献[8]都要高.

7.2 索引生成性能分析

为加密 EHRs 文件关键词,需先构造关键词集合 $W=(w_1,w_2,\ldots,w_n)$,总体 EHRs 文件总共具有 n 个属性值.根据 APSE 安全需求,属性值 n 需取较大值,实验测试中取 $n=(750,1000,1250,1500)$.

在生成 Index 索引的过程中,将每个 EHRs 文件映射出的 n 个属性值进行加密处理.该阶段的运行时间主要由以下两方面决定:数据库中记录条数(行数)以及索引矩阵大小(索引矢量长度).我们假定数据库中具有 1 000 个文件,每个文件的关键词属性值为 n.通过 Java 做生成 Index 索引仿真实验,生成 Index 索引时间见表 4.

Table 4 Time of building the index (ms)

表 4 建立索引所需时间 (毫秒)

关键词数	MCKS-II[5]	YLC18[8]	本文方案
750	174.293	152.433	92.885
1 000	213.672	206.729	105.887
1 250	239.631	230.767	128.561
1 500	278.324	270.335	144.281

实验分析可知:MCKS-II 方案与本文方案相对文献[8]方案在加密索引方面有明显的效率提升,且本文方案在生成 Index 索引时比 MCKS-II 效率更高.当属性值 $n=1500$ 时,处理 1 000 个文件索引时文献[8]方案时间为 270.335ms,MCKS-II 方案时间为 278.324ms,本文方案为 144.281ms.当属性值 n 增大时,本文方案在生成索引效率上比文献[5]和文献[8]方案更优.

7.3 陷门生成性能

生成陷门阶段将用户搜索向量 n 个属性值加密处理,需预计算 $\boldsymbol{M}_1^{-1}$ 和 $\boldsymbol{M}_2^{-1}$ 的时间.生成陷门 TD 的时间可见表 5.测试结果显示:本文方案在生成陷门 TD 时效率最高,其次是 YLC18 方案.

Table 5 Time of creating the trapdoor (ms)

表 5 生成陷门时间 (毫秒)

关键词数	MCKS-II[5]	YLC18[8]	本文方案
750	15.789	11.712	3.665
1 000	26.673	21.634	4.610
1 250	42.054	27.127	6.089
1 500	58.469	36.482	7.402

7.4 查询性能

表 6 列出了文件数量为 1 000 时方案的查询性能.在查询方面:本文方案在计算效率上较 MCKS-II 方案相同;文献[8]方案由于需要多组矩阵进行运算,求和时需将所有分块后的结果再统计到一起,因而,相比于本文方案以及 MCKS-II 方案会显得稍慢.但是,这一点对查询时间的影响不大,甚至不如程序运行两次时产生的误差大.

Table 6 Time of keyword search (ms)

表 6 关键词搜索时间 (毫秒)

关键词数	MCKS-II[5]	YLC18[8]	本文方案
750	57.457	61.483	56.049
1 000	65.467	68.115	64.301
1 250	69.376	72.457	70.083
1 500	77.934	78.328	75.878

将整个系统执行和关键词搜索分成 4 个阶段:初始化、建立搜索索引、生成陷门、查询搜索.分别在这 4 个阶段与文献[5]中的 MCKS-II 方案以及文献[8]中的方案进行对比,实验结果显示,本文方案在运行效率上得到了较好的提升.由于使用更小的可逆分块矩阵作为密钥,使得方案对关键词数的剧烈增长有着较好的抵抗性,符合大数据环境下存在大量关键词数且需要提供良好计算性能的电子医疗系统隐私保护范围搜索.

8 结束语

对称可搜索加密通过对称密钥可以快速地对文件进行加密,同时利用构造索引及陷门来对密文文件进行隐私保护下基于关键词的搜索.其优点在于加解密速度快,搜索表达灵活,适用于具有海量数据的大型文件(如文献和电子病历等)搜索环境.为提高效率的同时还能够保障灵活的范围搜索能力,以满足大数据医疗云中对医疗数据隐私保护下的搜索需求,本文提出了一种支持多关键词范围搜索的可搜索加密方案.该方案支持范围查询和通配符模式检索,利用文件索引矩阵和搜索陷门矩阵替代以往的一维向量,同时利用哈达马积运算简化查询匹配时间,极大地提高了搜索效率.实验结果表明:本文所提出的方案有效地缩短了系统初始化、索引构造以及陷门生成的时间,针对关键词数剧烈增长有着较好的抵抗性,满足了大数据环境下关键词数多的电子医疗系统的要求.然而在实际应用中,用户建立的搜索向量中包含大量少用或不用的关键词.降低未涉及关键词导致的系统复杂度问题,是我们下一阶段研究的目标.

References:

[1] Golle P, Staddon J, Waters B. Secure conjunctive keyword search over encrypted data. In: Proc. of the Int'l Conf. on Applied Cryptography and Network Security (ACNS 2004). LNCS 3089, Springer-Verlag, 2004. 31−45.

[2] Goh EJ. Secure indexes. ICAR cryptology ePrint archive, 2003/216, 2003.

[3] Chang YC, Mitzenmacher M. Privacy preserving keyword searches on remote encrypted data. In: Proc. of the Int'l Conf. on Applied Cryptography and Network Security. 2005. 442−455.

[4] Yang Y, Yang SL, Ke M. Rank fuzzy keyword search based on Simhash over encrypted cloud data. Chinese Journal of Computers, 2017,40(2):431−444 (in Chinese with English abstract).

[5] Zhang LL, Zhang YQ, Liu XF, Quan HY. Efficient conjunctive keyword search over encrypted electronic medical records. Ruan Jian Xue Bao/Journal of Software, 2016,27(6):1577−1591 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5005.htm [doi: 10.13328/j.cnki.jos.005005]

[6] Boneh D, Di Crescenzo G, Ostrovsky R, *et al*. Public key encryption with keyword search. In: Proc. of the EURORYPT 2004. 2004. 506−522.

[7] Li M, Yu S, Cao N, Lou WJ. Authorized private keyword search over encrypted data in cloud computing. In: Proc. of the IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Minneapolis, 2011. 383−392.

[8] Yang Y, Liu J, Cai S, Yang S. Fast multi-keyword semantic ranked search in cloud computing. Chinese Journal Computers, 2018, 41(6):1346−1359 (in Chinese with English abstract).

[9] Song DX, Wagner D, Perrig A. Practical techniques for searches on encrypted data. In: Proc. of the IEEE Symp. on Security and Privacy. 2000. 44−55.

[10] Curtmola R, Garay J, Kamara S, *et al*. Searchable symmetric encryption: Improved definitions and efficient constructions. Journal of Computer Security, 2011,19(5):895−934.

[11] Wang C, Cao N, RenK, *et al*. Enabling secure and efficient center keyword search over out sourced cloud data. IEEE Trans. on Paralleled Distributed Systems, 2012,23(8):1467− 1479.

[12] Wang C, Cao N, Li J, *et al*. Secure ranked keyword search over encrypted cloud data. In: Proc. of the IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Genova, 2010. 253−262.

[13] Li J, Wang Q, Wang C, *et al*. Fuzzy keyword search over encrypted data in cloud computing. In: Proc. of the IEEE INFOCOM. San Diego, 2010. 1−5.

[14] Cao N, Wang C, Li M, *et al*. Privacy-preserving multi-keyword ranked search over encrypted cloud data. IEEE Trans. on Parallel and Distributed Systems, 2014,25(1):829−837.

[15] Li JG, Tian XX, Zhou AY. Privacy preserving fuzzy keyword search in database as a service paradigm. Chinese Journal of Computers, 2016,39(2):414−428 (in Chinese with English abstract).

[16] Yu CM, Chen CY, Chao HC. Privacy-preserving multikey-word similarity search over outsourced cloud data. IEEE Systems Journal, 2015,99:1−10.

[17] Yang C, Zhang W, Xu J, *et al*. A fast privacy-preserving multi-keyword search scheme on cloud data. In: Proc. of the Int'l Conf. on Cloud and Service Computing. Shanghai, 2013. 104−110.

[18] Fu Z, Sun X, Linge N, *et al*. Achieving effective cloud search services: Multi-keyword ranked search over encrypted cloud data supporting synonym query. IEEE Trans. on Consumer Electronics, 2014,60(1):164−172.

[19] Fu Z, Wu X, Guan C, *et al*. Toward efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement. IEEE Trans. on Information Forensics and Security. 2016,11(12):2706−2716.

[20] Li S, Xu MZ. Attribute-based public encryption with keyword search. Chinese Journal of Computers, 2014,37(5):1017−1024 (in Chinese with English abstract).

[21] Xu P, Jin H, Wu QH, *et al*. Public-key encryption with fuzzy keyword search: A provably secure scheme under keyword guessing attack. IEEE Trans. on Computers, 2013,62(11):2266−2277.

[22] Ma M, Hed, Khanmk, *et al*. Certificate less searchable public key encryption scheme for mobile health care system. Computers & Electrical Engineering, 2017,43:1−9.

[23] Wang T, Au MH, Wu W. An efficient secure channel free searchable encryption scheme with multiple keywords. In: Proc. of the Int'l Conf. on Network and System Security. 2016. 251−265.

[24] Chen RM, Mu Y, Yang GM, *et al*. Dual-server public-key encryption with keyword search for secure cloud storage. IEEE Trans. on Information Forensics and Security, 2016,11(4):789−798.

[25] Jiang P, Mu Y, Guo FC, *et al*. Public key encryption with authorized keyword search. In: Proc. of the Australasian Conf. on Information Security and Privacy. 2016. 170−186.

[26] Chen RM, Mu Y, Yang GM, *et al*. Server-aided public key encryption with keyword search. IEEE Trans. on Information Forensics and Security, 2016,11(12):2833−2842.

[27] Li JG, Lin XN, Zhang YC, *et al*. KSF-OABE: Outsourced attribute-based encryption with keyword search function for cloud storage. IEEE Trans. on Services Computing, 2016. [doi: 10.1109/TSC.2016.2542813]

[28] Wong KK, Cheung DW, Kao B, Mamoulis N. Secure *k*-NN computation on encrypted databases. In: Proc. of the 35th SIGMOD Int'l Conf. on Management of Data. New York: ACM, 2009. 139−152.

[29] Zhang MW, Chen MW, He DB Yang B. An efficient leakage-resilient and CCA2-secure PKE system. Chinese Journal of Computers, 2016,39(3):482−502 (in Chinese with English abstract).

[30] Huang JJ. Privacy-preserving range-based multi-keyword search scheme in medical clouds. [MS. Thesis]. Wuhan: Hubei University of Technology, 2019 (in Chinese with English abstract).

[31] Li H, Yang Y, Dai Y, Yu S, Xiang Y. Achieving secure and efficient dynamic searchable symmetric encryption over medical cloud data. IEEE Trans. on Cloud Computing, 2017. [doi: 10.1109/TCC.2017.2769645]

[32] Wang SP, Liu LJ, Zhang YL. Verifiable dictionary-based searchable encryption scheme. Ruan Jian Xue Bao/Journal of Software, 2016,27(5):1301−1308 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4912.htm [doi: 10.13328/j.cnki.jos.004912]

[33] Chen B, Wu L, Kumar N, Choo KKR, He D. Lightweight searchable public-key encryption with forward privacy over IIoT outsourced data. IEEE Trans. on Emerging Topics in Computing, 2019. [doi: 10.1109/TETC.2019.2921113]

附中文参考文献:

[4] 杨旸,杨书略,柯闽.加密云数据下基于 Simhash 的模糊排序搜索方案.计算机学报,2017,40(2):431−444.

[5] 张丽丽,张玉清,刘雪峰,全韩彧.对加密电子医疗记录有效的连接关键词的搜索.软件学报,2016,27(6):1577−1591. http://www.jos.org.cn/1000-9825/5005.htm [doi: 10.13328/j.cnki.jos.005005]

[8] 杨旸,刘佳,蔡圣暐,杨书略.云计算中保护数据隐私的快速多关键词语义排序搜索方案.计算机学报,2018,41(6):1346−1359.

[15] 李晋国,田秀霞,周傲英.面向 DaaS 保护隐私的模糊关键字查询.计算机学报,2016,39(2):414−428.

[20] 李双,徐茂智.基于属性的可搜索加密方案.计算机学报,2014,37(5):1017−1024.

[29] 张明武,陈泌文,何德彪,杨波.高效弹性泄漏下 CCA2 安全的公钥加密体.计算机学报,2016,39(3):492−502.

[30] 黄嘉骏.医疗云中隐私保护多关键词范围搜索方案[硕士学位论文].武汉:湖北工业大学,2019.

[32] 王尚平,刘利军,张亚玲.可验证的基于词典的可搜索加密方案.软件学报,2016,27(5):1301−1308. http://www.jos.org.cn/1000-9825/4912.htm [doi: 10.13328/j.cnki.jos.004912]

张明武(1972－),男,博士,教授,CCF 高级会员,博士生导师,主要研究领域为应用密码学,信息安全与隐私保护技术.

韩亮(1955－),男,博士,教授,博士生导师,主要研究领域为信息安全.

黄嘉骏(1995－),男,硕士,主要研究领域为信息安全.