ISSN 1000-9825, CODEN RUXUEW
*Journal of Software*, Vol.21, No.5, May 2010, pp.991−1006
doi: 10.3724/SP.J.1001.2010.03746

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

# 数据库服务——安全与隐私保护*

田秀霞[1,2], 王晓玲[3+], 高 明[2], 周傲英[2,3]

[1](上海电力学院 计算机与信息工程学院,上海 200090)

[2](复旦大学 上海市智能信息处理重点实验室,上海 200433)

[3](华东师范大学 上海市高可信计算重点实验室,上海 200062)

## Database as a Service—Security and Privacy Preserving

TIAN Xiu-Xia[1,2], WANG Xiao-Ling[3+], GAO Ming[2], ZHOU Ao-Ying[2,3]

[1](Computer and Information Engineering College, Shanghai University of Electric Power, Shanghai 200090, China)

[2](Shanghai Key Laboratory of Intelligent Information, Fudan University, Shanghai 200433, China)

[3](Shanghai Key Laboratory of Trustworthy Computing, East China Normal University, Shanghai 200062, China)

+ Corresponding author: E-mail: xlwang@sei.ecnu.edu.cn

**Tian XX, Wang XL, Gao M, Zhou AY. Database as a services—Security and privacy preserving. *Journal of Software*, 2010,21(5):991−1006.** http://www.jos.org.cn/1000-9825/3746.htm

**Abstract**: This paper gives a summary of the secure and privacy preserving in database as a service (DaaS) from five primary techniques such as data confidentiality, data integrity, data completeness, query privacy preserving and access control policy. Data confidentiality is analyzed from the encrypted-based and division-based aspects; Data integrity and data completeness focus on the signature-based, challenge-response and probability-based aspects; Query privacy preserving and access control policy are analyzed mainly from exist problems. Finally, this paper gives the future research directions, existing problems and challenges of DaaS in the security and privacy preserving.

**Key words**: DaaS (database as a service); data confidentiality; data integrity; data completeness; access control policy

**摘 要**: 主要从数据的机密性、数据的完整性、数据的完备性、查询隐私保护以及访问控制策略这 5 个关键技术,综述国际上在数据库服务——安全与隐私保护方面的研究进展.数据的机密性主要从基于加密和基于数据分布展开分析;数据的完整性和完备性主要从基于签名、基于挑战-响应和基于概率的方法展开分析;查询隐私保护和访问控制策略主要从目前存在的问题展开分析.最后展望了数据库服务——安全与隐私保护领域未来的研究方向、存在的问题及面临的挑战.

**关键词**: 数据库服务;数据的机密性;数据的完整性;数据的完备性;访问控制策略

* Supported by the National Natural Science Foundation of China under Grant Nos.60673137, 60773075, 60925008 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant No.2008AA01Z1470967 (国家高技术研究发展计划(863)), the Shanghai Leading Academic Discipline Project of China under Grant No.B412 (上海市重点学科建设项目); the Research Fund for Excellent Youth Scholars of Shanghai Higher Education of China under Grant No.Z-2006-52 (上海市高校选拔优青教师科研专项基金)

Received 2009-05-11; Revised 2009-08-12; Accepted 2009-10-10

中图法分类号: TP311 文献标识码: A

随着网络技术和通信技术的日益成熟以及网络通信带宽的不断增加,越来越多客户信息、交易信息、医疗信息等涉及到个人隐私的数据以电子化的方式被存储和管理.为了避免昂贵的管理和聘请专业管理人员的费用,越来越多的企业寻求一种能够提供基本的硬件基础设施以外的数据管理服务,如提供容量计划、DBMS (database management system)、访问控制管理等以减少自身维护、管理带来的大量开销.数据库服务作为一种新的数据管理模式满足了企业需求,并可以提供与本地数据库一样的数据管理服务.

数据库服务虽然可以为客户提供必要的硬件、软件维护等,但是由于越来越多的数据信息涉及到个人的隐私,如一个人是否患有不希望公开的传染病或癌症,而且企业间的竞争以及数据库隐私数据窃取促使企业选择具有安全和隐私保护能力的数据库管理技术.因此,本文从数据库服务需要解决的如下 4 个方面的安全、隐私保护关键问题以及一个用来进一步提高数据库服务可用性的访问控制安全机制展开分析:

(1) 数据库的内容往往涉及企业的隐秘信息,因此,数据库服务需要有完善的数据安全机制来保证数据库的内容不会泄露(数据的机密性(data confidentiality)).

(2) 服务提供者不一定可信,不可能像企业维护自己的数据库一样可信,因此数据库服务需要保证数据库的内容不会被破坏(数据的完整性(data integrity)).

(3) 服务提供者不能随意删除或添加自己的任意的数据,因此,数据库服务需要保证服务提供者提供的数据是正确的,返回客户的结果是完备的(数据的完备性(data completeness)).

(4) 服务提供者不能察觉客户查询相应数据的目的,客户能够从 $N$ 个数据元素中检索第 $i$ 个元素而不被服务提供者发现客户对第 $i$ 个元素感兴趣(查询隐私保护(query privacy preserving)).

(5) 数据库服务在保证上述 4 个方面的安全与隐私的同时,也需要保证数据库的可用性,否则这样的数据库服务没有实际应用价值,因此,在数据库服务的可用性方面也提出了挑战:除了通过建立某个属性的索引信息保证数据库的可用性以外,如何开发数据库服务中的安全、有效的访问控制技术,使得在满足上述安全的情况下不同的用户具有不同的访问特权(访问控制策略(access control policy).

本文首先引入数据库服务的基本概念,并概括数据库服务的基本框架结构以及安全与隐私保护涉及的数据的机密性、数据的完整性、数据的完备性、查询隐私保护在数据库服务框架中的基本概念,然后展开这几个方面的实现技术,随后介绍增强数据库服务可用性的安全控制策略,最后提出数据库服务中目前存在的研究问题及其挑战,总结全文并展望其未来的研究方向.

## 1 数据库服务

### 1.1 数据库服务的概念

文献[1]提出了数据库服务的概念.该技术受到学术界和工业界的广泛关注.到目前为止,网络上已存在很多数据库服务,如 MySQLHosting(Adhost.com),IBM Data Center Outsourcing Services(www-1.ibm.com/services)等.

我们认为数据库服务(database as a service,简称 DaaS),也称作数据库外包(database outsourcing),就是指企业(数据拥有者)将自身的数据库创建、访问、维护、升级、管理等任务委托给专门的可以提供这些功能的第三方(数据库服务提供者)管理.采用 DaaS 一方面可以减轻企业购买昂贵的软件、硬件、处理软件升级、雇佣数据库管理和专业维护人员耗费的负担,另一方面,企业也可以将有限的资源集中在自身具有核心竞争力的业务上,同时,提供 DaaS 的专业企业也可以通过取得大量该业务的订单,对不同的企业提供类似的服务来减小开支,取得规模经济,获得利润.

### 1.2 数据库服务框架

我们用面向服务(service oriented architecture,简称 SOA)的观点用来刻画数据库服务的架构,如图 1 所示.架构中主要有 3 个角色(数据拥有者、数据库服务提供者和数据请求者)和 3 类数据(数据源、查询与结果、密钥).

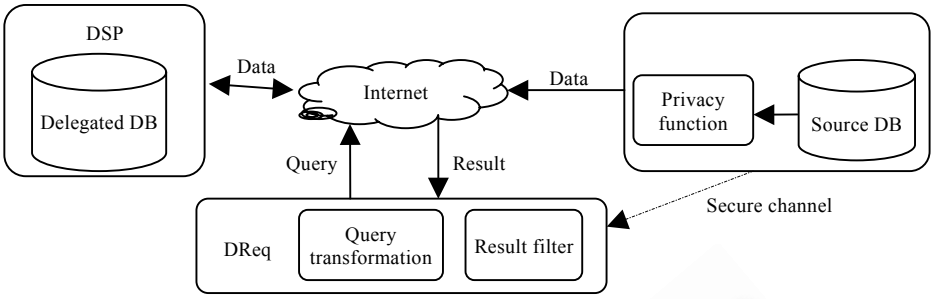



Fig.1 Architecture of DaaS

图 1 DaaS 的框架结构

我们首先介绍 3 个角色:数据库服务提供者(DaaS provider,简称 DSP)、数据拥有者(data owner,简称 DO)、数据请求者(data requestor,简称 DReq).

(1) 数据库服务提供者(DSP)

DSP 是指一个专业的提供 DaaS 的企业,维护企业的数据库(图 1 中委托管理的数据库(delegated DB),并能正确地进行数据库的复制、备份等数据管理任务.但是 DSP 不一定能够保证数据的机密性,并且也可能是数据库的攻击者.所以为了防止 DSP 对委托的数据库中数据的未授权访问,他从 DO 处接收的是经过保护用户隐私方式(如图 1 中 DO 处的隐私数据处理模块(privacy function))处理过的数据.DSP 可以根据 DO 提供的辅助信息(索引信息)有效地响应数据请求者的查询,但是不能查看查询结果.

(2) 数据拥有者(DO)

DO 是实际上拥有自身用户数据(图 1 中数据源(source DB))的企业,产生用户数据并将数据以保护用户隐私的方式委托给 DSP.为了加快对委托的数据库的查询效率,DO 需要提供一些辅助手段,如针对某些字段建立保护隐私的索引或采用保护隐私的访问控制授权等,以增强委托数据库的可用性.

(3) 数据请求者(DReq)

DReq 是指可以将用户的查询转换成数据库服务器可识别的查询(如通过图 1 中 DReq 处的查询转换(query transform)实现),将数据库服务器返回的保护隐私的查询结果经过处理(如通过图 1 中 DReq 处的结果过滤(result filter)实现),方便用户进行查询后处理的前端.DReq 具有一定的计算和存储能力,如 DReq 可以是计算机,也可以是手机或无线 PDA 等.

图 1 中涉及的 3 类数据传送操作是:DO 和 DSP 之间的数据传送,DReq 和 DSP 之间的查询和结果返回,DO 和 DReq 之间的密钥分发和验证结构传送.具体如下:

(1) DO 和 DSP 之间的数据传送

DO 和 DSP 之间的数据传送是指在传统数据库的基础上增加的 DO 和 DSP 之间的交互.在 DO 和 DSP 之间传送数据时,数据必须以某种保护数据隐私的方式(如加密数据)传送,因为企业传送的数据涉及到企业员工或财务隐私信息.

(2) DReq 和 DSP 之间的查询和结果返回

DReq 和 DSP 之间的查询和结果返回是指 DReq 可以向 DSP 提交查询,提交的查询与客户-服务器模式类似,不同的是,DReq 的查询需要经过查询转换,转换成 DSP 可以识别的相关属性的隐私保护形式.DSP 接受查询并在加密的数据库上执行该查询,然后返回查询结果.

(3) DO 和 DReq 之间的密钥分发和验证结构传送

DO 和 DReq 之间的密钥分发和验证结构传送主要是为了使 DReq 能够验证 DSP 正确并且完备地返回 DO 希望返回的数据,DO 将验证的密钥或验证结构以一定的安全方式(如图 1 中虚线表示的安全信道(secure channel))传送给 DReq.

目前的研究[1,2]也对 DaaS 架构进行了一些探索.文献[2]提出了 3 种模式:统一客户模式(unified client model)、多查询者模式(multi-querier model)和多数据拥有者模式(multi-owner model).统一客户模式是指每一个委托的数据库仅仅有 1 个客户(上述数据拥有者和数据请求者归一)使用,该客户创建、维护和查询数据等.多查询者模式是指与图 1 匹配的模式,有两种类型的客户(分别是图 1 中的 DO 和 DReq),DO 添加、删除和修改数据库记录,而多个数据请求者可以查询数据库.多数据拥有者模型是指可以有多个拥有不同安全原则的数据库拥有者创建数据库.实际上,后面两个模型的共同点是都可以存在多个查询请求者.

### 1.3 安全和隐私保护

DaaS 是一种不同于以往本地数据库管理的管理模式,为了保证 DaaS 的安全与隐私保护,图 1 框架中的每一部分可能分别对应于数据的机密性、数据的完整性、数据的完备性以及查询隐私保护中一个或多个.而且由于数据库管理模式的不同,数据的机密性、数据的完整性、数据的完备性以及查询隐私保护的概念也不同于本地数据库管理模式,具体如下:

(1) 数据的机密性

数据的机密性由图 1 中 DO 处的隐私数据处理模块(如,加密)实现,是指 DO 在将其数据委托给 DSP 之前需要对委托的数据进行隐私保护处理,经过处理的数据可以保证数据库的内容在没有授权的情况下不能被访问,即使是 DSP 也不能访问,或者即使可以访问也因为不知道秘密信息而不知道确切的数据.DaaS 提供的数据的机密性主要包括两层含义:一是保护数据不被未授权的 DReq 访问;二是保护数据不被不可信的 DSP 访问.只有当这两种情况下的机密性都被保证时才可以保证企业机密的信息不会被泄露.

(2) 数据的完整性

数据的完整性由图 1 中 DO 处的隐私数据处理模块(如,签名)实现,是指 DO 需要提供额外的机制来保证 DSP 对 DReq 提出的查询的返回的结果是完整的,即返回的查询结果是真实的原始数据,并且没有任何篡改.实际上,数据的完整性也存在两层含义:一是保证数据来源的真实性,确实是取自 DO 的数据,这个完整性也称作真实性(authenticity):二是保护数据不被未授权地修改,这是通常意义上的完整性.

(3) 数据的完备性

数据的完备性由图 1 中的 DSP 和 DO 共同实现,是指 DO 需要提供额外的机制(如验证结构)来保证 DSP 对 DReq 提出的查询的返回结果是完备的,也就是说,查询在整个数据库上能够正确执行,并返回所有满足查询条件的元组,DSP 不能任意地向委托的数据库中增加元组或者删除数据库中的已有元组.保证查询结果的完备性,就是查询结果应该是未经删减过的数据库拥有者实际委托给 DSP 的原始数据(内容和元组个数相同).

(4) 查询隐私保护

查询隐私保护也称作隐私信息检索,通过图 1 中 DReq 中的查询转换/结果过滤模块实现,是指 DO 的数据库在委托给 DSP 后,为了保护 DReq 的查询意图,DReq 需要提供保护请求者隐私的查询,仅仅通过这个查询,DSP 不能分析请求者的查询目的,从而也不能分析 DReq 的行为模式.

目前,DaaS 的安全与隐私保护问题是 DaaS 模式成为实际的应用之前必须解决的问题.第 2 节~第 4 节将分别从保证数据的机密性、数据的完整性和完备性和查询隐私保护的实现技术进行分析.

## 2 数据的机密性

数据的机密性主要采用基于数据加密的方法和基于数据分布的方法实现.下面分别介绍两种实现方法并分析其在保证数据的机密性时的共同点和不同点.

### 2.1 基于数据加密的方法

由于 DSP 不一定完全可信,目前采用的保证 DaaS 中数据的机密性的方法主要是在 DO 处对需要委托的数据进行加密操作而在 DReq 处进行结果过滤和解密处理.主要的加密算法为对称算法(如 DES,3DES)和非对称算法(如 RSA)[3].对称算法具有较快的加密速度,一般选择用来进行数据加密,非对称算法则用来加密秘密密钥.

数据库加密的粒度有 4 种:表、字段、元组和元组属性值.为了提高加密的安全性和灵活性,目前主要采用对元组和元组属性值进行加密的方法.加密的数据库虽然保证了数据的机密性,但是加密后的数据给查询工作造成了技术实现上的困难,如不能实现模糊匹配查询、多表查询不必要的伪连接、解密工作量增加、查询响应时间开销增大等.因此,如果选择一个不好的加密方法来加密数据,数据库数据的可用性就大大降低,这样会给 DSP 和 DReq 带来很大的额外开销,从而导致 DaaS 模式的优势受到很大的影响.因此,在 DaaS 的应用中,我们需要选择一种能够有效支持数据操作(如范围查询、聚集查询等)的加密算法.

例如,根据选择的加密算法,表 1 在 DSP 处存储的对应的加密关系为

$$Employee^s(etuple, eid^s, ename^s salary^s, addr^s did^s).$$

其中,元素 *etuple* 是通过选择的加密算法存储的关系 *Employe* 中的一个元组的对应的密文(加密字符串).每一个属性(如 $eid^s$)是对应于属性 *eid* 的索引(保护隐私的索引、桶分区索引等)[4,5],被用来在 DSP 处加快 DReq 查询的快速执行.

**Table 1** Employee Relation

**表 1** 职工关系表

| eid | ename | salary | addr | did |
|---|---|---|---|---|
| 1810 | ZhangLi | 8k | PuDong | 10 |
| 1811 | WangMing | 10k | YangPu | 20 |
| 1812 | HuangMei | 6k | HongKou | 50 |
| 1813 | LiLi | 20k | XuHui | 30 |

对数据库数据加密的研究开始于密钥管理[6],后来对数据库数据加密的技术发展成由 Song 等人[7]提出的基于加密的文本串的关键字搜索.该方法可以成功地应用在邮件服务器上,对加密的邮件进行分类,也可以应用于关系表中实现关键字搜索.最初提出的加密查询处理就是将整个加密的数据库发送给 DReq,在这种情况下,DSP 不提供查询引擎服务并且查询在 DReq 端进行.这一方法引起的问题就是由于带宽有限使得数据传输耗费巨大,并且解密和查询处理整个数据库都需要在 DReq 端进行,从而使得具有较弱计算和处理能力的请求者设备(如手机和无线 PDA)不能使用 DaaS 提供的服务.NetDB2 为网络上的 DaaS[1],NetDB2 解决了在 DaaS 中的两个重要的挑战,就是数据隐私和性能问题.NetDB2 使用加密:软件层次的加密(采用 BLOWFISH 和 RSA 算法)或硬件层次的加密(采用硬件实现的 DES 算法)保证了隐私并第一次直接解决和评估了 DaaS 的性能问题(分别从数据库加密的不同粒度如字段、行或页进行性能测试并评估),这对 DaaS 的成功是非常关键的.

为了提高对加密后数据的有效查询,文献[8]提出了一项桶分区技术,并利用一种基于代数的方法对加密的字段进行查询重写,其主要思想就是通过将明文域分成多个分区并给它们分别指定唯一的 id 桶编号,然后将明文值映射到所在分区对应的唯一桶编号即密文值.文献[9]则提出了一个比分区技术好,用数学定义良好的保持顺序和距离的加密函数,但是提出的计算结构只是对某些查询类型有效.文献[10]中提出了一种为加密查询处理提供确定性的桶尺寸优化技术的算法.因此,如果具有一个较好的过滤机制,从 DSP 处检索数据的通信消耗就会大为减少,因而查询响应时间就会减少.但是,由于过滤过程的质量严格依赖于泄露给 DSP 的信息,因此这些方法存在隐私性能降低的问题.为了执行范围查询,这种桶分区技术对桶的 id 映射函数采用了文献[11]中提出的保持顺序的加密函数,因此使范围查询可以得到很好的支持.随后,文献[12,13]则指出,利用保持顺序的加密函数会降低安全性,即增加数据库信息泄露的危险性,而且由于用户保存的密钥太多,也会在一定程度上导致密钥的泄露.为了在 DaaS 环境中支持更多的查询,文献[14]提出了用数学表达式处理 SQL 查询聚合函数以处理带有嵌套的复杂查询,并为提出的查询类型提供了查询策略.文献[15]提出了使用同态加密方法支持安全的聚合查询,即一个在聚合字段上的操作可以通过计算服务器端某个聚合字段的聚合,并在客户端解密而获得.

### 2.2 基于数据分布的方法

鉴于上述加密方法对查询性能的影响,研究者从数据分布的角度探索了另外一种秘密共享的方法,以实现数据的机密性[16,17].该类方案中基于数据分布的思想来源于孙子定理的密钥分散管理思想[18].设 $D$ 是一个需要保护的秘密数据,$D \in U$,$U$ 是一个有效的需要保护的数据集合,并且设 $D_1,\ldots,D_n \in V$ 是秘密数据 $D$ 的某种拆分,$V$

是子数据集合,则$(k,n)$门限方案的特点就是:知道 $k$ 个或更多个 $D_i$,很容易计算秘密数据 $D$,使得 $H(D\mid D_{i_1},...,D_{i_t})=0$;但是,知道 $k$–1 或更少个 $D_i$,计算秘密数据 $D$ 是不确定的,使得 $H(D\mid D_{i_1},...,D_{i_t})=H(D)$,因为此时集合 $U$ 中的所有元素都有可能是秘密数据 $D$,其中 $H(\cdot)$是熵函数.满足这两个特点的$(D_1,D_2,\ldots,D_n)$称为$(k,n)$门限方案(threshold scheme).

Shamir 利用孙子定理提出了第一个没有依赖密钥的$(k,n)$门限方案[19].利用 Shamir 密钥共享方法计算秘密数据的有效性特点,Emekci 等人在文献[16,17]中采用基于数据分布的方法安全地处理需要委托的隐私数据.他们将密钥共享的理论应用于秘密数据字段值,并将得到的 $n$ 个(如 $n$=3)数据拆分分别存放在不同的 DSP 处,解决了 DaaS 中的单个 DSP 不安全和查询隐私保护问题.文献[16]中开发的中间件 Abacus 主要应用于数据仓库,并采用了耗费低的哈希函数来隐藏数据,能够实现选择、交集和连接等查询操作.后来,Emekci 等人把这个工作扩展到适用于任何类型的数据库,并将 DSP 的集合与 Shamir 的密钥共享算法应用于 DO 需要保护的隐私数据[17].

如图 2 所示,假设一个 DO 使用 Shamir 密钥共享理论将其数据库委托给专业的 DSP(如 DSP 1,DSP 2,DSP 3)进行管理和维护,同时不泄露数据内容给任何 DSP $i$,$1\le i\le 3$.假设 DO 的数据库中有如表 1 所示的单个的 Employee 表,为了保护 salary 字段,DO 在有限域 $F_p$($p$ 是一个大素数)上为每个 salary 字段值选择了一个次数为 1 的多项式,多项式的常量项是对应的 salary 值.图 2 显示了 DO 在将其数据库委托给 3 个提供者时,为了保护 salary 字段而存放在 3 个 DSP 处的各个 salary 字段值的数据拆分.

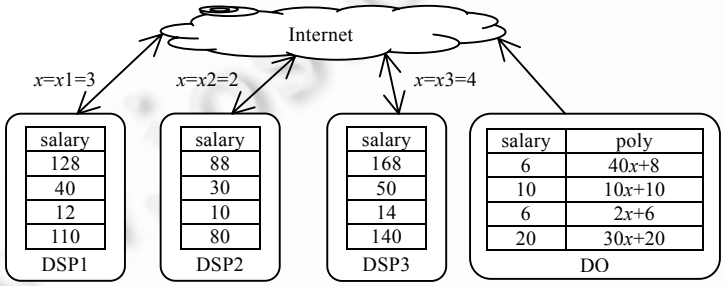



Fig.2 Application of secret share theory in DaaS

图 2 秘密共享理论在 DaaS 中的应用

### 2.3 两种方法的比较

首先,两种方法都可以保证委托数据的机密性,以有效预防 DSP 即使泄露存储的数据也无法泄密.基于加密的方法依赖用户密钥的安全性,而基于分布方式则依赖 $n$ 个 DSP 中少于 $k$ 个人是不可信的,一旦用户密钥泄露或者多于 $k$ 个 DSP 共谋,就会造成隐私数据的泄露.

其次,保护隐私数据的原理不同.基于加密的方法主要采用各种加、解密函数实现,而基于数据分布的方法则采用 Shamir 密钥共享理论实现.

最后,基于加密的方法虽然保证了数据的机密性,但是加密的数据库却使得执行一个查询所需要进行的加、密解密数据的计算复杂性增加了查询响应时间,增加了 DReq 进行查询转换和过滤服务器返回的查询结果的负担,并且不能实现模糊匹配查询、多表查询不必要的伪连接等.而基于数据分布的方法一方面解决了大量加解密带来的查询响应时间性能问题,另一方面可以有效地实现连接、范围、精确匹配等查询,并且为了提高基于数据分布的工作效率,也可以采用构造某一域的保持顺序的多项式集合的方法,特别是利用秘密数据值的单调递增函数决定多项式的系数.该方法可以实现只从 DSP 处检索需要的数据或者一个小的超集的方法,因而它可以进一步减小查询处理过程中的计算和通信消耗.

## 3 数据的完整性和数据的完备性

数据的完整性确保一个查询的结果确实来自真实的 DO,而数据的完备性则确保一个查询在目标域上正确

地执行并返回所有满足查询条件的数据.虽然数据的完整性和数据的完备性概念有所不同,但是他们都需要DO 提供额外的机制来保证,而且有些机制既然以实现数据的完整性也可以实现数据的完备性.到目前为止,实现数据的完整性或完备性的机制主要有 3 种:基于数字签名(digital signature)的方法、基于挑战-响应(challenge-respond)的方法和基于概率(probability)的方法.

### 3.1 基于数字签名的方法

目前,大多数方案都采用基于数字签名的方法来实现数据的完整性和数据的完备性.数字签名,简称签名,是一种基于对称密码或非对称密码(公钥密码机制)的算法[3].大部分签名算法都是基于非对称密码机制实现的.常见的数字签名方法如 RSA,DSA 等.签名具有以下几个特征:签名是可信的、不可伪造的、不可重用的;签名的文件是不可篡改的;签名是不可抵赖的.数据的完整性正是利用数字签名的这些特征来确保查询结果来自真实的原始数据,没有任何篡改.

(1) 基于简单签名的方法

检查 DSP 为查询返回的数据(元组)是否属于 DO 委托的原始数据库的数据,最简单的方法就是由 DO 对需要委托的数据库中的每个元组进行签名.DO 在将数据库委托给 DSP 的同时,将每个元组对应的签名也发送给 DSP.为使 DReq 能够验证元组的真实来源,DO 需要通过安全的通信渠道将其有效公钥发送给 DReq.当 DReq 向 DSP 提交一个查询后,DSP 除了返回要求的查询结果以外,还需要返回所有结果元组对应的签名,这样 DReq 就可以通过验证返回元组的签名,决定是否接受返回的元组.如果签名验证不成功,就说明元组被篡改过,DReq 可以拒绝接受这些元组.

比如,为了保证元组的完整性,DO 可以在需要委托的表中增加一个签名字段用来存放每个元组对应的签名,或者计算出的元组签名单独存放.DO 可以通过下面的公式来计算表 1 中每个元组 $t_i$ 的签名 $s_i$:

$$s_i = S(H(t_i), key),\ 1 \le i \le 4.$$

$S$ 是一个签名算法(如 RSA),$key$ 是 DO 的密钥, $H$ 是一个单向哈希函数,可以避免重放攻击.单向哈希函数具有如下的特征:它能把一个变长字符串 $V$ 作为输入并把它转换成一个定长二进制序列 $H(V)$;求它的逆是非常困难的,也就是说,已知 $V'$,攻击者要找到一个 $V$ 使得 $H(V)=V'$ 在计算上是不可能的[3].

(2) 基于 Merkle 哈希树的方法

Merkle 哈希树[20]是一种高效的用于数据认证的树结构.它与简单签名方法不同,不需要为每个元组都计算一个签名.Merkle 哈希树是一棵二叉树,其每个叶子节点对应一个元组,存储这个元组经过单向哈希函数变换后的哈希值(指纹);其每个中间节点存储一个由其孩子节点值连接后经过单向哈希函数变换后的哈希值;直到按照中间节点的计算方法计算出根节点的哈希值为止.最后,只需要对二叉树的根节点值签名,就可以为树中的所有叶节点(元组)分别提供完整性认证.从 Merkle 哈希树的构建过程可知,Merkle 哈希树只需要计算一系列的哈希值而不是签名.由于单向哈希函数的计算代价比签名的计算代价要小很多,所以 Merkle 哈希树方法和简单签名方法在效率上有很大的改进.

假设表 1 中的 4 个元组分别表示为 $t_1,t_2,t_3,t_4$,其对应的 Merkle 哈希树和 Merkle 签名 $s$ 构建如图 3 所示.$MH$ 是一个单向哈希函数,$H_0,\ldots,H_6$ 是对应树中各个节点的哈希值,$\|$表示字符串连接.对 $H_0$ 进行签名,该签名就是整个数据库的签名,也称为 Merkle 签名.

在使用 Merkle 签名验证返回结果的完整性时,当 DReq 提交一个查询以后,DSP 除了返回 DReq 的结果元组之外,还需要返回认证路径(authentication path)和 Merkle 签名.DReq 利用结果元组和认证路径,在本地计算出 Merkle 哈希树根节点的哈希值.然后使用 DO 的公钥验证返回的 Merkle 签名,验证 Merkle 签名经过公钥解密后是否与本地计算出的根节点的哈希值相等,如果相等,则相信返回的元组没有被篡改.

实际上,简单签名方法只是针对每个元组进行签名,当 DSP 删除一些元组时,DReq 并不能验证这样的查询结果的完备性.文献[21,22]利用聚合签名方法签名每一个元组,元组具有排序序列中两个相邻元组的信息,这样可以通过检查聚合签名是连续的来确保一个简单查询结果的完备性.实际上这种方法只能有效处理连接操作的一个子集,因为只有当连接的结果形成原始数据的一个连续的区域时才能保证完整性和完备性.聚合签名方

法虽然可以很好地处理 DSP 返回查询结果的完整性和完备性的检查问题,但是这种方法的缺陷在于为每个元组维护一个签名会带来大量的性能损失,如占用空间大、计算重复耗时和维护困难,每次更新、插入或者删除元组 DO 都需要重新计算并传输签名.基于 Merkle 哈希树的签名方法[23]虽然只需要计算一系列的哈希值和一次签名,节省了大量的验证和传输代价,但是 Merkle 哈希树最好构建在排好序的数据元组上,这样可以与简单签名方法一样,通过返回包含结果区间的两个边界元组来保证查询结果的完备性.然而仍然存在某些类型的查询,DReq 端的网络和 CPU 负载非常大,而且在某些极端的例子中,负载可能与本地处理这些查询的负载一样高,这样就不能体现采用 DaaS 模式的优势.为了保证数据库数据的最新性,还需要一个辅助的系统以安全及时的方式传送最新的根签名给每个 DReq.文献[24,25]中提出了一种不需要额外的系统就可以自然解决数据更新问题的方法.该方法只需要根据当前的时间变化插入数据元组.

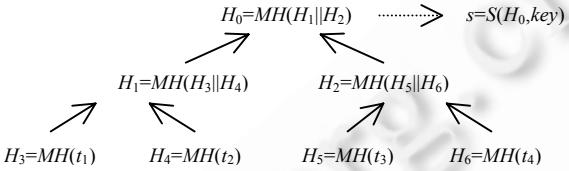



Fig.3 Merkle Hash tree and Merkle signature

图 3 Merkle 哈希树与 Merkle 签名

### 3.2 基于挑战-响应模式的方法

在一个网络、非集中以及恶劣的环境中,DSP 是否能够正确执行查询遭到质疑,因为一个懒惰或有恶意的 DaaS 服务提供者可能会为了避免因为执行查询支付 CPU 和存储费用而选择返回一个不准确或完全不正确的查询结果.为了解决这个问题,文献[26]提出了一种对任意查询,确保查询正确执行的解决方法.该方法是建立在挑战-响应协议上的运行时查询证明机制.在密码学中,挑战-响应协议经常用来证明一个公钥密码方案的安全性[3],如一个公钥密码方案是否具有选择密文安全性的证明等,需要挑战者和攻击者之间的不断交互.在运行查询证明机制中,对每一批查询,DSP 被挑战提供查询执行证明,然后,这作为接受实际的查询结果是否正确的先决条件在 DReq 端被检查,以确定是否接受查询返回的结果.这种方法要求 DReq 首先将数据元组分为 $n$ 份($d_1$,…, $d_n$),要与 DO 委托数据库时对数据库元组的分组一致(可以根据时间等).当他提交查询 $Q$ 时,需要首先在本地计算出 $CT=H(Q(d_i))$,$i\in[1,n]$,作为挑战值与查询一起发送给 DSP,DSP 在响应查询并返回查询结果的同时,需要返回 $i$ 值作为挑战值的响应值.因此提交查询的 DReq 就可以通过检查 DSP 是否返回正确的响应值来检查 DSP 是否正确地执行了 DReq 提交的查询.该方法提出的目的是为了实现只有当 DSP 在所有的数据元组上执行过查询后才能得到正确的 $i$ 值.但是,当 DReq 只是发送正确的挑战值给 DSP 时,DSP 仍然可以拒绝执行部分数据元组上的查询,因为只要所有需要的挑战值对应的响应 $i$ 值都已找到,就不需要再执行其他数据元组上的查询了.

### 3.3 基于概率的方法

基于概率的方法到目前为止主要有两种:完整性审计方法[24]和双重加密方法[27].

(1) 完整性审计方法

完整性审计方法就是通过在需要委托的数据库中插入少量的伪造元组来验证查询返回数据元组的完备性.对一个提交的查询,存在一定的概率使得插入的伪造元组和原始数据元组一起返回,因而可以通过有效地分析返回结果中的伪造元组实现数据的完备性验证.这种方法中对于完整性的认证包括数据认证(验证元组是否被篡改)和元组认证(检验元组是伪造元组还是真实元组)实现.数据认证则与前面描述的简单签名方法类似,但只是在表中增加一个验证列(元组数据连接后的哈希值或指纹).如下的公式表示验证列 $a_h$ 的生成:

$$a_h = H(tid \| a_1 \| a_2 \| ... \| a_n).$$

其中,$tid$ 是关键字,$a_i$,$1\le i\le n$,表示元组中相应字段的值.根据上述公式,表 1 中第一元组的验证列的生成可表示为

$$a_h = H(1810 \| ZhangLi \| 8 \| PuDong \| 10).$$

单向哈希函数 $H$ 的使用,使得攻击者如果想从加密的元组中计算出一个有效的指纹 $a_h$ 在计算上是不可能的,因为加密阻止了攻击者知道相应字段的明文值,而它才是真正需要被输入到单向哈希函数中的数据.元组认证则通过如下公式来实现:

$$a_h = \begin{cases} H(tid \| a_1 \| a_2 \| ... \| a_n), & \text{真实元组} \\ H(tid \| a_1 \| a_2 \| ... \| a_n)+1, & \text{伪造元组} \end{cases}.$$

验证列如果等于 $H(tid \| a_1 \| a_2 \| ... \| a_n)$,则 DReq 知道这是个真实的元组;如果等于 $H(tid \| a_1 \| a_2 \| ... \| a_n)+1$,那么他知道这是个伪造元组,否则是一个攻击者恶意插入的元组.

伪造元组生成方式的不同对 DReq 的计算和存储要求也有所不同.如果只是简单地生成一些随机的伪造元组,并将这些元组存储在 DReq 端会造成其额外的存储开销,以及每次查询时高昂的检查代价.文献[24]提出了使用确定性函数(比如线形函数等)来生成伪造元组的方法,确定性函数产生的数据是被加密过的,DSP 不能区分插入的伪造元组数据和加密数据库中的任何其他元组.因此,数据库拥有者只需要向 DReq 发送函数定义而不是所有的伪造元组.这样一方面避免了 DReq 存储大量的伪造元组,另一方面提高了检查返回结果的完整性时的性能.使用伪造数据来进行完整性和完备性验证的方法相对于已有的工作而言,具有 DSP 透明性,DSP 不需要提供除了数据库后台以外的任何其他数据结构以及处理逻辑来支持这种方法,而且这种方法可以通过简单扩展来支持很多复杂的数据库查询,比如连接和更新操作.

(2) 双重加密方法

Wang 等人在文献[27]中提出了一种低耗费的可证明安全性的双重加密的方法来实现数据元组的完整性认证,可以应用于手机或无线 PDA 设备中,克服了文献[24,25]中往数据库中插入与真实数据元组具有相似分布的伪造数据元组的缺点,不需要构造伪造数据元组,就可以实现数据元组的完整性认证.文中假设 DO 需要委托的数据库 $T$ 在 DSP 中对应的数据库 $T_s$ 由两部分构成,如图 4 所示.一部分数据元组由 $T$ 中的所有原始数据元组用主密钥 $k$ 加密形成,另一部分则由 $T$ 的百分之 $r$ 部分用辅助密钥 $k'$ 加密形成,其中 $r$ 是一个重复因子.这样,DSP 处的数据库 $T_s$ 就包括两种采用不同加密方法得到的数据元组,其实也可以把 $T_s$ 看作是由 3 部分构成的,如图 4 所示,其中,I,II(与 III 是同一部分元组)两部分是用同一个主密钥 $k$ 加密的数据元组,III(与 II 是同一部分元组)是用辅助密钥 $k'$ 加密的数据元组.

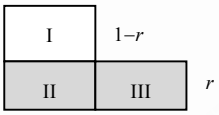



Fig.4 Component of delegated database $T_s$

图 4 委托数据库 $T_s$ 的组成部分

为了保证 DReq 查询结果的完整性,他们在每个元组中添加了一个称作 *dual* 的信息,这个 *dual* 信息主要通过 DO 和 DReq 共享的密钥和单向哈希函数生成.比如,假设 DO 和 DReq 之间有一个共享密钥 *key*,那么对任何元组 $t$ 的 *dual* 信息 $t_{dual}$ 的计算如下,其中 $H$ 是单向哈希函数:

$$t_{dual} = \begin{cases} H(key,t), & t \in \text{I} \\ H(key,t)+1, & t \in \text{II} \\ H(key,t)+2, & t \in \text{III} \end{cases}.$$

这样,根据单向函数的性质,可以很容易地通过检查 $t_{dual}$ 信息来判断元组 $t$ 是不是一个有效的元组,如果是有效的元组,那么可以通过上式知道元组是通过什么方式加密得到的,然后解密元组并访问元组.

### 3.4 3种方法的比较

3 种方法都用来实现数据的完整性和数据的完备性,但是它们分别采用了不同的原理并且在实现数据的完整性和完备性的保证上存在一定的差异.

首先,基于签名的方法采用签名原理既可以实现数据的完整性也可以实现数据的完备性,不过在实现数据的完备性时需要附加一个验证结构;基于挑战-响应的方法采用密码学中的挑战-响应协议来实现数据的完整性,但是不能实现数据的完备性;基于概率的方法则是采用插入一定数量的伪造元组或不同加密的重复元组的方法实现数据的完整性和完备性,但是采用不同加密的重复元组的方法不能实现数据的完备性.

其次,基于签名的方法(如聚合签名和 Merkle 签名)在实现数据的完备性时需要建立在有序的数据元组上,因而给系统带来较大的时间空间开销.虽然基于挑战-响应的方法解决了基于前面的方法建立在有序的数据元组上的问题,但是这样的方案也不能检测所有的恶意攻击,如,当恶意的 DSP 检索了完备的结果但是为了商业利益只返回部分查询结果,而且挑战-响应方法需要通过修改 DBMS 核心以实现查询执行证明.为了解决基于数据签名的方法和基于挑战-响应方法带来的问题,基于概率的方法来确保 DSP 返回查询结果的正确性和完备性.它不需要修改 DBMS 核心,除了查询处理以外,不需要数据库引擎执行任何特殊的函数就可以实现数据库的正确性和完备性检测,其检测只取决于 DBMS 返回的结果.

最后,若一个查询结果经验证满足上述数据的完整性和完备性,则其也满足文献[2,22]中所说的查询认证(query authentication)或文献[26,28]中的查询执行保证(query execution assurance)或文献[21,29]中的查询正确性和完备性的要求.如果同时满足数据的完整性和完备性要求,就说明 DSP 提供的查询结果和用户希望从 DO 获得的结果完全一致(内容和数量都相同),DSP 没有随意添加、删除或修改数据,是一个可信的 DSP.但是他们都没有考虑查询刷新的问题,也就是保证查询结果来自最新更新的数据库.文献[30]中曾经提过查询刷新的问题,但是并没有给出相应的解决方案.文献[25]利用文献[24]中提出的在需要委托的数据库中插入、删除伪造元组的方法实现了查询刷新的问题,使得查询结果总是能够正确地反映当前 DSP 处数据库的最新更新状态,而不是旧的数据库状态.同时,文献[25]还通过在基于认证的数据结构中添加时间戳的方式提供刷新保证.

## 4 查询隐私保护

查询隐私保护,也称作隐私信息检索,一直是多年来的研究主题.这个问题首先在这样的上下文环境中提出:一个访问 DSP 处数据的 DReq 不向 DSP 泄露他确切的个人兴趣[31,32].如一个分析家想从被某一企业设置为有效的投资家数据库中检索信息,但是他并不想把自己未来的意图暴露给这家企业.这个问题最初的意图是根据 DSP 处的数据在 DSP 处是可用的来解决的,特别是 DReq 不想把他们的查询泄露给 DSP,因为这些查询信息可能会泄露他们关于行为模式的隐私.

如果一个数据库被模拟为一个长度为 $N$ 的字符串,并被保存在远程服务器上,DReq 想根据位置信息 $i$ 检索信息 $x_i$,但是并不泄露任何关于 $i$ 的信息给 DSP[31−33],一个对 DReq 来说不好的解决方法就是检索整个数据库,但如果只有 1 个服务器[34],这个方法就被认为是最好的方法.在这个领域的理论研究已经有很长时间的历史,并且大多数解决方法都是依赖于多个服务器的可用性[35]或多个 DSP[36],其中涉及的通信复杂性最好达到 $O(n^{1/5.25})$.文献[37]提出通信复杂性可以减少到 $O(n^{1/32582658})$,并且显示通信复杂性在关于 Mersenne 素数密度[38]合理的数字理论假设的前提下可以达到 $n^{o(1)}$.然而如果想获得线性通信复杂性,那么可以在多个 DSP 处提供同一个数据库的多个备份,通过实验知道提供 $k$ 个 DSP 可以使通信复杂性上限减少到 $O\left(N^{\frac{\log\log k}{k\log k}}\right)$[37].

确保客户查询隐私的隐私信息检索概念也被扩展到数据隐私的领域,这被称作对称隐私信息检索.Emekci 等人与 Sion 等人在文献[17,39]的工作中挑战了隐私信息检索和对称隐私信息检索的计算实际性,通过广泛的实验和评估说明了使用隐私信息检索协议检索信息要比传送整个数据库到 DReq 端以保护 DReq 隐私的速度慢几个阶.因此,他们对隐私信息检索协议在实际中的可用性提出怀疑,并陈述了需要找到替代的方法来解决非常实际的 DaaS 问题.同时,文献[40]中也显示,需要在一个相当高的时间复杂度内使用加密结果实现一个隐私保护的问题.比如,对一个合成的数据:包括一个站点的 10 个文档和另一个站点的 100 个文档使用基于加密方法的耗费估计是 2 小时的计算和 3GB 的数据传输.同样地,对于一个实际的包括大约 100 万条医疗记录的数据集,基于加密的方法大约需要耗费 4 小时的计算和 8GB 的数据传输.文献[17]中提出的基于数据分布的方法,就是将数据分发在多个服务器上而不是用数据加密的方法获得数据安全和客户查询的隐私,并且通过实验证明了数据分布在保护数据安全和客户查询隐私方面的高效性.文献[6,36]从如何抵抗强攻击者的角度描述了隐私信息检索.文献[36]首先提出了当服务器响应失败或有意、无意回答不正确时,如何抵抗这类攻击的概念和解决方案.实际上,为了减少通信复杂性而使用多个服务器存放信息的拷贝,这也是造成安全性脆弱的原因之一,同时

在信息理论上使得数据库操作难以寻找数据库内容的确切位置.文献[40]中的方案增强了隐私信息检索系统针对强攻者的抵抗,并引入了混合隐私信息检索的概念.它不仅在信息理论上,而且进一步在计算上保证了服务器的联合不能攻击用户的隐私.

## 5 访问控制策略

为了有效查询加密的数据,在基于数据加密部分曾经指出,与加密数据一起存放的还有索引信息[4,5],DBMS 可以使用这些索引信息选择查询中返回的数据元组.虽然这在一定程度上可以减少返回到 DReq 端的非请求的元组的个数,但是它们都是假设 DReq 可以访问全部查询结果[8,10,11].这个假设并不适合真实的现实世界应用,不同的用户应该具有不同的访问特权.如通过选择性地加密数据以便于 DReq 只解密他们被授权访问的数据[41−48].实际上,选择性的加密数据早在基于 XML 的发布[49,50]中就已经得到应用.

### 5.1 访问控制策略

访问矩阵是一个表示访问控制策略的概念模型,指定了每一个主体 $s$ 对每个对象 $o$ 拥有的访问权利.矩阵中每个主体占一行,每个对象占一列.每个单元格 $A[s,o]$ 指定行中的主体对列中的对象的访问授权,如果 $A[s,o]=1$,则表示主体 $s$ 可以访问对象 $o$;否则,$A[s,o]=0$.访问控制的任务就是保证只有矩阵授权的操作被执行.假设一个访问矩阵 $A$,访问控制列表 $ACL_i$ 表示相应于第 $i$ 列的向量(阅读元组 $t_i$ 的主体的集合),能力列表 $CAP_j$ 表示相应于第 $j$ 行的向量(用户 $u_j$ 可以阅读的对象的集合).假设系统中有 3 个用户 A,B,C,他们需要根据不同的授权阅读表 1 中的元组 $t_1,t_2,t_3,t_4$,如图 5 所示的访问矩阵就是表示实现不同授权的例子.

| | $t_1$ | $t_2$ | $t_3$ | $t_4$ |
|---|---|---|---|---|
| A | 0 | 1 | 1 | 0 |
| B | 1 | 0 | 1 | 1 |
| C | 0 | 0 | 1 | 1 |

Fig.5 Access control matrix

图 5 访问控制矩阵

### 5.2 访问控制策略的增强

图 5 表示的访问控制策略虽然可以放在传统的客户-服务器模式的服务器端,以控制用户对对象的访问授权,但是通过密码学增强的访问控制策略却不能委托给提供数据库服务的 DSP,因为 DO 和 DReq 不信任服务器能够保证委托的数据库内容和访问控制策略的机密性.因此,DO 不得不参与到访问控制增强中,除非数据本身能够实现选择访问.可以采用不同的方法来增强访问矩阵中的访问授权,如使用单个数据密钥加密数据元组的方法、不同密钥加密不同数据元组的方法、密钥推导的方法.

(1) 单个数据密钥加密数据方法

单个数据密钥加密数据元组的方法就是对每个数据元组都使用同一个数据密钥来加密,数据密钥随后使用可以访问数据元组的用户的加密密钥(私钥)加密.在这样的访问控制策略下,单个数据密钥的备份的数量可以达到系统中用户的数量.对大型系统来说,允许对数据的细粒度访问(如在单个元组层次或属性层次的加密),这个数量可能远远超过需要保护数据元组本身的尺寸.所以当访问控制策略变化时,需要更新加密的数据密钥,并重新分发给系统中的所有用户.因此,这样的访问控制策略只适合策略长时间固定不变的数据库系统,对于用户变化较频繁和需要更新的数据库系统来说是不适合的.

(2) 不同密钥加密不同数据元组的方法

使用不同密钥加密不同数据元组的方法就是对每个元组都采用一个不同的密钥进行加密,这将导致密钥的数量和数据库中的数据元组一样多.在这样的访问控制策略下,为了访问对应的多个元组,每个用户不得不同时维护多个密钥(维护他能够访问的所有元组的密钥).然而,存在大量的密钥会造成密钥管理的困难,不适合现实世界中的数据管理.

(3) 密钥推导方法

密钥推导方法就是给定一个密钥和一系列公共有效的信息推导另一个密钥,以增强选择数据加密的访问控制.密钥推导方法主要基于层次结构或有向无环图(directed acyclic graph,简称 DAG)实现[51].文献[42]提出了一个基于用户集合之间偏序关系的密钥生成方案.它不是采用基于广播加密方法[43],而是采用基于 Diffie-Hellman 的密钥生成方案给偏序关系形成的 V-图中的节点分配公钥和私钥.它要求每个用户集合只需存放 1 个密钥.给定用户集合 $A$ 的密钥和一些公开的信息,可以计算分配给用户集合 $B$ 的密钥当且仅当集合 $B$ 在包含次序中比 $A$ 低(在 V-图中,$B$ 是 $A$ 的孩子).而文献[46,47]则提出使用另一种密钥推导方法,它首先引入了两个函数:用户密钥分配函数 $\phi$ 和资源分配函数 $\lambda$.$(\phi,\lambda)$根据访问矩阵 $A$ 是完备的(complete),如果每个用户根据 $A$ 可以解密他能访问的所有元组,表示为$(\phi,\lambda)\Rightarrow A$.$(\phi,\lambda)$根据访问矩阵 $A$ 是合理的(sound),如果没有用户可以根据 $A$ 解密他不能访问的元组,表示为$(\phi,\lambda)\Leftarrow A$.如果满足上述两个条件则说$(\phi,\lambda)$正确地增强 $A$,表示为$(\phi,\lambda)\Leftrightarrow A$.用户层次 $UH=(2^{|U|},\leq)$指的是,层次的域是用户集合 $U$ 的幂集,$\leq$是子集包含的偏序关系.在图形上,UH 层次可以表示为一个图(用户层次图),为 $2^{|U|}$中的每个元素建立一个顶点,存在一个连接顶点 $v_1$ 和顶点 $v_2$ 的路径当且仅当 $v_2\leq v_1$,一个图的顶点因此相应于一个用户的集合(或用户组).用户层次图中的每个顶点 $v$ 都与一个密钥 $k_v$ 关联,图中的一个路径 $v_1\rightarrow v_2\rightarrow\ldots\rightarrow v_n$ 代表一个密钥推导路径,意味着从与顶点 $v_1$ 关联的密钥开始,沿着路径,可能推导出路径上所有顶点 $v_i,i=2,\ldots,n$ 的密钥.

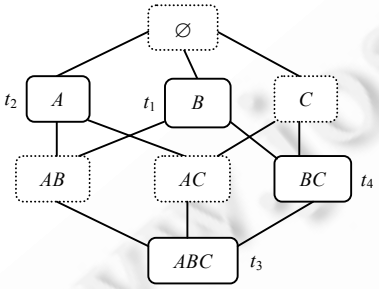



Fig.6 User hierarchy graph corresponding to Fig.5

图 6 相应于图 5 的用户层次图

增强图 5 中访问控制矩阵的用户层次图如图 6 所示.用户集合 $U=\{A,B,C\}$,$\{\varnothing,A,B,C,AB,AC,BC,ABC\}$为其对应的幂集,分别表示在图 6 中.实线框中的用户集合(如,$BC$)代表访问控制矩阵中对应元组($t_4$,相应的用户集合旁边的标识)的访问控制列表,虚线框中的用户集合则是用户集合的幂集中除了访问控制矩阵中对应元组的用户集合以外的用户集合.图 6 表示的 UH 图中存在路径 $B\rightarrow BC\rightarrow ABC$,则意味着密钥 $k_{ABC}$ 可以从密钥 $k_{BC}$ 推导出,$k_{BC}$ 可以从 $k_B$ 推导出.根据 UH 的定义,$\phi(u)=k_{v_i}, v_i=\{u\}, t\in\lambda(k_v), acl_t=v$,因此,只要 $u\in v$,每个用户就可以推导所有密钥 $k_v$.

基于树层次的密钥推导函数比 DAG 图上的密钥推导函数方便,它们能够更好地支持动态的 DaaS 场景.DSP 处的数据库中的访问控制策略可能会发生改变,如由于添加、删除用户或添加、删除元组引起的策略变化.DAG 密钥推导方法基于复杂的数学理论(模指数)以至于对自动环境的支持是复杂的和无效的.相比而言,树层次方法则利用了简单的哈希函数.为了避免 DAG 层次的缺点,文献[46]中实现了一种将基于 DAG 的 UH 转换为基于树的用户层次图(tree based user hierarchy,简称 TUH)的算法,但是同时也引入了使得每个用户的密钥环包含不再只有 1 个密钥的缺点.文献[47]则是在设置好增强访问控制的访问控制策略后,为了正确地访问和管理委托的数据库,DReq,DO 和 DSP 不得不存储一些额外的称作元数据的信息,如授权元数据、描述元数据和密钥管理元数据,这样,DReq 端和 DSP 就可以使用这些元数据解释和执行 SQL 语句,并正确地管理存储的数据.

虽然上述 3 种方法都是根据为元组创建密钥的方法来实现 DaaS 场景中的访问控制增强.但是使用单个数据密钥加密数据的方法和使用不同密钥加密不同元组的方法来实现密码学增强的访问控制在现实世界中是不适用的.而密钥推导方法则可以有效地应用于 DaaS 场景中,实现对加密数据的选择访问,使得拥有不同密钥的用户可以访问不同的数据元组.而且密钥推导方法和元数据的结合使用可以更加有效地应用在自动的 DaaS 场景中,实现有效的加密数据的选择访问.文献[45]中提出了一种新颖的双层加密的方法,能够实现访问控制增强和策略变化时的有效演化.它在数据上施加两层加密,内层由用户加密以提供对数据的初始保护,外层则由 DSP 加密以反映策略的变化.这样,两层机制的结合提供了一个 DaaS 场景中有效的增强的访问控制策略.文献[52]也提出了一个可以有效的分发加密数据的密钥密码学增强的访问控制策略.但是,由于 DaaS 场景的特殊性,用来

设计增强访问控制策略的公共信息同样不能被攻击者利用,否则可能泄露系统增强的安全授权策略.为了解决这个问题,文献[48]中提出了通过在存放公开信息的公开目录中增加一个保护层的方法,需要在新的加密层中遵循与密钥推导路径逆序方式的推导路径.该方法可以安全、有效地发布公开信息,并只允许授权访问数据的用户根据密钥推导路径推导密钥.

## 6 存在的问题和挑战

前面总结的各种研究成果虽然从不同方面解决了DaaS中的安全和隐私保护问题,但是,如何将各个安全解决方案有机地融合为一个整体,建立一个健壮而且有效的服务,以安全和隐私保护的方式去管理数据,是一个需要解决的问题.也就是说,DSP 需要同时保证数据是安全的(机密性),查询结果也是完整的(完整性),并且查询可以被安全有效地、正确地执行(查询隐私保护和完备性),同时尽可能好地提高数据的可用性(访问控制策略).

随着可信计算[53]在国内外研究和实现的展开,如果将作为服务的数据库建立在一个可信的硬件操作平台上,则可以在一定程度上避免对所有数据库数据执行加密操作,从而可以有效地提高数据执行效率,并支持实现各种数据库操作.目前不少文献已经关注可信硬件在DaaS模型中的应用[52,54].

归纳起来,在DaaS场景中,在假设硬件基础设施和必要的维护有效的情况下,主要存在以下问题和挑战:

(1) 查询处理

如何安全、有效地实现各种查询,如精确匹配、范围查询、聚集和连接查询.到目前为止,只有在基于数据分布的方式中有效地实现了精确匹配查询.虽然很多文献描述了对范围查询的实现,但大多数是基于一个桶分区和保持顺序加密函数的结合来实现范围查询,而且主要是针对数字字段的范围查询.为了实现聚集操作,大部分文献都集中在使用同态加密函数来实现,目前已经实现了 SUM,AVG,COUNT,MAX,MIN,但是没有实现比较操作.实现连接查询的文献则比较少,目前主要是基于概率的方法和基于数据分布的方法实现连接查询.

(2) 更新处理

目前提出的文献大多数都是在静态的DaaS场景中实现,为了实现动态的DaaS场景,有的文献中提出建立动态的索引结构方法,也有的提出使用基于密钥推导的密码学增强的访问控制策略方法来实现.有的文献则提出寻求一种懒惰更新与递增更新的方法来实现更新.其实不管是什么方式,主要是保证数据库能够进行有效的更新,最好不需要在每次更新数据时都需要重新加密数据库或更新整个认证结构,以保证更新可以有效地实现.其实在动态的DaaS场景中还有一个大多被忽略的问题,就是如何实现查询刷新问题,使得查询结果可以反映数据库的最新变化.

(3) 隐私和公共数据管理

目前大多数文献都在考虑如何保护委托的数据库中用户的敏感信息,很少考虑一旦数据被存放在 DSP 处,一个用户可能不仅想查询他自己的私有信息,而且想查询DSP提供的公有数据信息.因此,DSP可以提供附加的值:不仅存放数据请求者的私有数据还提供与大型公有数据存储的无缝访问和整合.比如,一个数据请求者委托的数据库可能包含其朋友的私有数据的集合,如电话号码、地址等,而DSP可能还为饭店提供服务,包括饭店及其地址的数据库,数据请求者可以利用数据库请求查询一个离他的朋友家比较近的饭店,但是并不泄露他的朋友的任何隐私信息.

(4) 访问控制

由于DaaS模式的特殊性,到目前为止对DaaS模式中访问控制的研究有限,主要是集中在如何由不可信的DSP通过再次加密的方式增强访问控制并依赖DSP实现访问控制策略的变化,但是由于加密方法的多次使用导致系统的工作效率不高.访问控制机制是控制用户执行合法授权的第一步,如果能够将访问控制和保证数据安全和隐私的方式结合起来将使得DaaS的可用性进一步增强.今后的研究工作需要关注如何设计DaaS模式中灵活、有效的访问控制机制.

## 7 总结与展望

本文回顾了最近几年来国际上在 DaaS 这一新兴数据库领域的主要研究成果,综述了 DaaS 面临的严峻的安全和隐私保护问题,主要从以下 5 个关键技术展开:数据的机密性、查询隐私保护、数据的完整性、数据库的完备性以及访问控制策略.最后总结了 DaaS 作为一个新兴的数据库研究领域目前存在的问题以及挑战,并归纳总结了 DaaS 环境中目前研究的主要问题以及挑战.相信随着 DaaS 中安全问题的解决以及不同安全保护机制的整合,DaaS 将成为今后企业或个人部署数据库应用程序的主要选择.

**References**:


[1] Hacigümüs H, Mehrotra S, Iyer B. Providing database as a service. In: Proc. of the Int'l Conf. on Data Engineering. Washington: IEEE Computer Society Press, 2002. 29−38.

[2] Mykletun E, Narasimha M, Tsudik G. Authentication and integrity in outsourced databases. ACM Trans. on Storage, 2006,2(2): 107−138.

[3] Schneier B, Wrote; Wu SZ, Zhu SX, Zhang WZ, Trans. Applied Cryptography—Protocols, Algorithms, and Source Code in C (Second Edition). Beijing: China Machine Press, 2006 (in Chinese).

[4] Shmueli E, Waisenberg R, Elovici Y, Gudes E. Designing secure indexes for encrypted databases. In: Proc. of the IFIP Conf. on Database and Applications Security. LNCS 3654, Heidelberg, Berlin: Springer-Verlag, 2005. 54−68.

[5] Yang ZQ, Zhong S, Wright RN. Privacy-Preserving queries on encrypted data. In: Proc. of the 11th European Symp. on Research in Computer Security. LNCS 4189, Heidelberg, Berlin: Springer-Verlag, 2006. 479−495.

[6] Davida GI, Wells DL, Kam JB. A database encryption system with subkeys. ACM Trans. on Database Systems, 1981,6(2):312−328. [doi: 10.1145/319566.319580]

[7] Song DX, Wagner D, Perrig A. Practical techniques for searches on encrypted data. In: Proc. of 2000 IEEE Symp. on Research in Security and Privacy. Washington: IEEE Computer Society Press, 2000. 44−55.

[8] Hacigümüs H, Iyer B, Mehrotra S, Li C. Executing SQL over encrypted data in the database service provider model. In: Proc. of the ACM SIGMOD Conf. New York: ACM Press, 2002. 216−227.

[9] Özsoyoglu G, Singer DA, Chung SS. Anti-Tamper databases: Querying encrypted databases. In: Proc. of the 17th Annual IFIP WG 11.3 Working Conf. on Database Applications and Security. Cleveland: Case Western Reserve University, 2003. 133−146.

[10] Hore B, Mehrotra S, Tsudik G. A privacy-preserving index for range queries. In: Nascimento MA, Özsu MT, Kossmann D, Miller, Blakeley RJJA, Schiefer KB, eds. Proc. of the 13th Int'l Conf. on Very Large Data Bases. New York: ACM Press, 2004. 720−731.

[11] Agrawal R, Kiernan J, Srikant R, Xu YR. Order preserving encryption for numeric data. In: Proc. of the ACM SIGMOD Conf. New York: ACM Press, 2004. 563−574.

[12] Kantarcioglu M, Clifton C. Security issues in querying encrypted data. In: Proc. of the IFIP Conf. on Database and Applications Security. LNCS 3654, Heidelberg, Berlin: Springer-Verlag, 2005. 325−337.

[13] Li J, Omiecinski ER. Efficiency, security trade-off in supporting range queries on encrypted databases. In: Proc. of the IFIP Conf. on Database and Applications Security. LNCS 3654, Heidelberg, Berlin: Springer-Verlag, 2005. 69−83.

[14] Chung SS, Ozsoygolu G. Anti-Tamper databases: Processing aggregate queries over encrypted databases. In: Proc. of the 22nd Int'l Conf. on Data Engineering Workshops. LNCS 4127, Heidelberg, Berlin: Springer-Verlag, 2006. 89−103.

[15] Ge TJ, Zdonik SB. Answering aggregation queries in a secure system model. In: Koch C, Gehrke J, Garofalakis MN, Srivastava D, Aberer K, Deshpande A, Florescu D, Chan CY, Ganti V, Kanne CC, Klas W, Neuhold EJ, eds. Proc. of the 33rd Int'l Conf. on Very Large Data Bases. New York: ACM Press, 2007. 519−530.

[16] Emekci F, Agrawal D, ElAbbadi AE. Abacus: A distributed middleware for privacy preserving data sharing across private data warehouses. In: Proc. of ACM/IFIP/USENIX the 6th Int'l Middleware Conf. LNCS 3790, Heidelberg, Berlin: Springer-Verlag, 2005. 21−41.

[17] Emekci F, Agrawal D, Abbadi AE, Gulbeden A. Privacy preserving query processing using third parties. In: Barga RS, Zhou XF, eds. Proc. of the Int'l Conf. on Data Engineering. Washington: IEEE Computer Society Press, 2006. 27.

[18] Cao ZF. The Public Cryptography. Harbin: Heilongjiang Education Press, 1993. 158−195 (in Chinese).

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

[20] Merkle RC. A certified digital signature. In: Proc. of the 9th Annual Int'l Cryptology Conf. on Advances in Cryptology. LNCS 435,

Heidelberg, Berlin: Springer-Verlag, 1989. 218−238.

[21] Pang H, Jain A, Ramamritham K, Tan KL. Verifying completeness of relational query results in data publishing. In: Ozcan F, ed. Proc. of the ACM SIGMOD Conf. New York: ACM Press, 2005. 407−418.

[22] Narasimha M, Tsudik G. Authentication of outsourced database using signature aggregation and chaining. In: Lee ML, Tan KL, eds. Proc. of the 11th Int'l Conf. on Database Systems for Advanced Application. LNCS 3882, Heidelberg, Berlin: Springer-Verlag, 2006, 420−436.

[23] Sion R. Secure data outsourcing. In: Koch C, Gehrke J, Garofalakis MN, Srivastava D, Aberer K, Deshpande A, Florescu D, Chan CY, Ganti V, Kanne CC, Klas W, Neuhold EJ, eds. Proc. of the 33rd Int'l Conf. on Very Large Data Bases. New York: ACM Press, 2007. 1431−1432.

[24] Xie M, Wang HS, Yin J, Meng XF. Integrity audit of outsourced data. In: Koch C, Gehrke J, Garofalakis MN, Srivastava D, Aberer K, Deshpande A, Florescu D, Chan CY, Ganti V, Kanne CC, Klas W, Neuhold EJ, eds. Proc. of the 33rd Int'l Conf. on Very Large Data Bases. New York: ACM Press, 2007. 782−793.

[25] Xie M, Wang HX, Yin J, Meng XF. Providing freshness guarantees for outsourced databases. In: Kemper A, Valduriez P, Mouaddib N, Teubner J, Bouzeghoub M, Markl V, Amsaleg L, Manolescu I, eds. Proc. of the 11th Int'l Conf. on Extending Database Technology: Advances in Database Technology, Vol.261. New York: ACM Press, 2008. 323−332.

[26] Sion R. Query execution assurance for outsourced database. In: Bohm K, Jensen CS, eds. Proc. of the 31st Int'l Conf. on Very Large Data Bases. New York: ACM Press, 2005. 601−612.

[27] Wang HX, Yin J, Perng CS, Yu PS. Dual encryption for query integrity assurance. In: Proc. of the 17th ACM Conf. on Information and Knowledge Management. New York: ACM Press, 2008. 863−872.

[28] Mouratidis K, Sacharidis D. Pang H. Partially materialized digest scheme: An efficient verification method for outsourced databases. The VLDB Journal, 2009,18(1):363−381. [doi: 10.1007/s00778-008-0108-z]

[29] Pang H, Tan KL. Verifying completeness of relational query answers from online servers. ACM Trans. on Information and System Security, 2008,11(2):1−50. [doi: 10.1145/1330332.1330337]

[30] Li FF, Hadjieleftheriou M, Kollios G, Reyzin L. Dynamic authenticated index structures for outsourced databases. In: Proc. of the ACM SIGMOD Conf. New York: ACM Press, 2006. 121−132.

[31] Chor B, Goldreich O, Kushilevitz E, Sudan M. Private information retrieval. Journal of the ACM, 1998,45(6):965−982. [doi: 10.1145/293347.293350]

[32] Razborov AA, Yekhanin S. An $\Omega(n^{1/3})$ lower bound for bilinear group-based private information retrieval. Theory of Computing, 2007,3(1):221−238. [doi: 10.4086/toc.2007.v003a012]

[33] Beime A, Stahl Y. Robust information-theoretic private information retrieval. Journal of Cryptology, 2002,20(3):295−321.

[34] Saint-Jean F. Java implementation of a single-database computationally symmetric private information retrieval (cSPIR) protocol. Technical Report, YALEU/DCS/TR-1333, Department of Computer Science, Yale University, 2005.

[35] Aggarwal G, Bawa M, Ganesan P, Garcia-Molina H, Kenthapadi K, Motwani R, Srivastava U, Thomas D, Xu Y. Two can keep a secret: A distributed architecture for secure database services. In: Proc. of the 2nd Biennial Conf. on Innovative Data Systems Research. 2005. 186−199.

[36] Gasarch W. A survey on private information retrieval. Bulletin of the EATCS, 2004,82:72−107.

[37] Yekhanin S. Towards 3-query locally decodable codes of subexponential length. Journal of the ACM, 2008,55(1):1−16.

[38] Kedlaya KS, Yekhanin S. Locally decodable codes from nice subsets of finite fields and prime factors of Mersenne numbers. In: Proc. of the 2008 IEEE 23rd Annual Conf. on Computational Complexity. Washington: IEEE Computer Society Press, 2007. 175−186.

[39] Sion R, Carbunar B. On the computational practicality of private information retrieval. 2007. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.70.793

[40] Goldberg I. Improving the robustness of private information retrieval. In: Proc. of the 2007 IEEE Symp. on Security and Privacy. Washington: IEEE Computer Society Press, 2007. 131−148.

[41] Damiani E, De Capitani di Vimercati S, Foresti S, Jajodia S, Paraboschi S, Samarati P. Key management for multi-user encrypted databases. In: Proc. of the ACM Workshop on Storage Security and Survivability. New York: ACM Press, 2005. 74−83.

[42] Zych A, Petkovi M. Key management method for cryptographically enforced access control. 2006. https://www.cosic.esat.kuleuven.be/wissec2006/papers/5.pdf

[43] Petković M, Conrado C, Hammoutne M. Cryptographically enforced personalized role-based access control. In: Proc. of the 21st IFIP Int'l Information Security Conf. Boston: Springer-Verlag, 2006. 364−376.

[44] De Capitani di Vimercati S, Foresti S, Jajodia S, Paraboschi S, Samarati P. A data outsourcing architecture combining cryptography and access control. In: Proc. of the ACM Workshop on Computer Security Architecture. New York: ACM Press, 2007. 63−69.

[45] De Capitani di Vimercati S, Foresti S, Jajodia S, Paraboschi S, Samarati P. Over-Encryption: Management of access control evolution on outsourced data. In: Koch C, Gehrke J, Garofalakis MN, Srivastava D, Aberer K, Deshpande A, Florescu D, Chan CY, Ganti V, Kanne CC, Klas W, Neuhold EJ, eds. Proc. of the 33rd Int'l Conf. on Very Large Data Bases. New York: ACM Press, 2007. 123−134.

[46] Damiani E, De Capitani di Vimercati S, Foresti S, Jajodia S, Paraboschi S, Samarati P. Selective data encryption in outsourced dynamic environments. Electronic Notes in Theoretical Computer Science, 2007,168:127−14. [doi: 10.1016/j.entcs.2006.11.003]

[47] Damiani E, De Capitani di Vimercati S, Foresti S, Jajodia S, Paraboschi S, Samarati P. Metadata management in outsourced encrypted databases. In: Jonker W, Petkovic M, eds. Proc. of the 4th VLDB Workshop on Secure Data Management (SDM 2007). LNCS 3674, Heidelberg: Springer-Verlag, 2007. 16−32.

[48] De Capitani di Vimercati S, Foresti S, Jajodia S, Paraboschi S, Pelosi G, Samarati P. Preserving confidentiality of security policies in data outsourcing. In: Proc. of the 7th ACM Workshop on Privacy in the Electronic Society. New York: ACM Press, 2008. 75−84.

[49] Miklau G, Suciu D. Controlling access to published data using cryptography. In: Freytag JC, Lockemann PC, eds. Proc. of the 29th VLDB. Berlin: VLDB Endowment, 2003. 898−909.

[50] Bertino E, Ferrale E. Secure and selective dissemination of XML documents. ACM Trans. on Information and System Security, 2002,5(3):290−331.

[51] Akl SG, Taylor PD. Cryptographic solution to a problem of access control in a hierarchy. ACM Trans. on Computer System, 1983, 1(3):239−248. [doi: 10.1145/357369.357372]

[52] Anciaux N, Benzine M, Bouganim L, Pucheral P, Shasha D. Ghostdb: Querying visible and hidden data without leaks. In: Chan CY, Ooi BC, Zhou AY, eds. Proc. of the ACM SIGMOD Conf. New York: ACM Press, 2007. 677−688.

[53] Challenger D, Yoder K, Catherman R, Stafford D, Van Doorn L, Wrote; Zhao B, Yan F, Yu FJ, Trans. A Practical Guide to Trust Computing. Beijing: China Machine Press, 2009. 9−30 (in Chinese).

[54] Sion R. Trusted hardware: Can it be trustworthy? In: Proc. of the 44th Annual Design Automation Conf. New York: ACM Press, 2007. 1−4.

**附中文参考文献**:

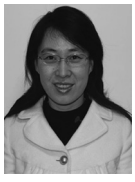
[3] Schneier B,著;吴世忠,祝世雄,张文政,译.应用密码学协议、算法与 C 源程序.北京:机械工业出版社,2006.

[18] 曹珍富.公钥密码学.哈尔滨:黑龙江教育出版社,1993.158−195.

[53] Challenger D, Yoder K, Catherman R, Stafford D, Van Doorn L,著;赵波,严飞,余发江,译.可信计算.北京:机械工业出版社,2009. 9−30.

**田秀霞**(1976－),女,博士生,副教授,主要研究领域为密码学,电子商务安全,数据库安全与隐私保护.

**高明**(1980－),男,博士生,主要研究领域为不确定数据管理,数据流管理技术.



**王晓玲**(1975－),女,博士,副教授,主要研究领域为 XML 数据管理.

**周傲英**(1965－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为数据挖掘,XML 数据管理,P2P 对等计算,海量计算.