文章编号:1001-9081(2010)09-2339-05

异构数据库加解密系统的关键技术研究与实现

郝文宁,赵恩来,刘玉栋,黄 亚,刘军涛
(解放军理工大学 工程兵工程学院,南京 210007)
(lgdxzel@163.com)

摘 要:对数据进行加密是保护信息机密性的一种有效途径,针对一般加解密系统的异构数据库兼容性差以及密文查询方式单一的问题,提出了一种新的数据库加密方式:以领域元数据为支撑,采用对象关系映射模型屏蔽异构数据库,通过构建密文索引来实现灵活多样的密文查询;设计并实现了异构数据库加解密系统。实验结果和理论分析表明:系统能够支持多种类型数据库的加解密,提供多种方式的密文查询,并提高了数据库加密的安全性。

关键词:异构数据库;密文索引;数据库加密;对象关系映射;Lucene索引
中图分类号:TP309.7 **文献标志码**:A

Research and implementation of key techniques of encryption and decryption system on heterogeneous database

HAO Wen-ning, ZHAO En-lai, LIU Yu-dong, HUANG Ya, LIU Jun-tao
(*Engineering Institute of Corps of Engineers, PLA University of Science and Technology, Nanjing Jiangsu* 210007, *China*)

Abstract: Data encryption is an effective way to guarantee the confidentiality of information. Concerning the poor compatibility of heterogeneous database and single type query of encrypted data in the general encryption and decryption systems, the authors provided a new database encrypted mode on the basis of domain metadata: adopted the Object Relation Mapping (ORM) mechanism for heterogeneous databases, improved flexible and various cryptograph query by building ciphertext index, and designed and implemented the encryption and decryption system on the heterogeneous database. The theoretic analysis and the experimental results show that the system can support various types of heterogeneous databases, provides multi-mode cryptograph queries and increases the security of database encryption.

Key words: heterogeneous database; ciphertext index; encryption database; Object Relation Mapping (ORM); Lucene index

0 引言

随着数据库技术的不断发展与应用,数据库安全日益成为人们关注的热点。目前数据库的安全性主要通过访问控制来保障,当访问控制被攻破时,整个数据库的安全体系也就随之瓦解,目前解决该问题的主要方法是采用数据库加密[1]。数据库加密技术在保证数据安全性的同时,也给数据库系统带来一些不利影响。数据库加密的局限性主要有:系统运行效率受到影响;难以实现对数据完整性约束的定义;对数据库的SQL语言及SQL函数操作受到制约;密文数据更易受到攻击等[2]。因此,如何保证数据库中数据的机密性、完整性以及灵活可用性一直是数据工程研究的热点和难点。

文献[3]通过多重桶划分建立复合索引的思想,在增强密文索引的安全性的同时进一步提高了密文查询的命中率。文献[4]在文献[3]的基础上,提出了一种最佳桶划分策略,平衡了密文索引的安全性和查询效率问题;但由于建立多重索引增加了服务器的运行时间,效率不高。文献[5]采用常规加密方法进行加密,并把数据处理后的特征值作为附加字段与加密数据一起存储,但不支持数值型数据的加密。文献[6]在分析数值型数据的保序加密方法的基础上,提出了一种针对字符数据的模糊匹配加密方法,但不支持数值型数据的加密,亦不能防止推断攻击。文献[7]提出一种基于字段加密的数据库加密方法,通过引入随机数转换加密密钥的方式增强密文数据库的安全性;不足之处采用乱序表提高安全性,但是乱序表自身安全性不保。

虽然目前各大数据库厂商开发的数据库管理系统均支持数据加密功能,然而各大厂商实现的数据库加密功能均依托于自己底层的元数据,这些元数据之间存在着较大的差异,且各个数据库厂商所支持的数据加密算法、密钥长度甚至实现机制均各不相同,由于用户仅关心数据的使用和查看,并不关心数据的具体存储、加解密等。故加密系统应对异构数据库中的数据实施统一加解密管理。

在对数据库中的数据进行加密之后,如何灵活高效地查询加密数据将成为一个难题。通常有两种方法:一种方法是首先对加密数据进行解密,然后对解密数据进行查询。这种方法需要对整个数据库进行加/解密操作,开销巨大,在数据量比较大的情况下是不可行的。另一种方法首先对查询语句中的条件值进行相同的加密处理,然后与数据库中存储的加密数据进行比较。但是,由于数据加密操作,数据的有序性、可比性等遭到破坏,该方法的适用范围也受到限制[8]。

收稿日期:2010-03-11;**修回日期**:2010-05-08。

作者简介:郝文宁(1971-),男,山西运城人,副教授,博士,主要研究方向:军用数据工程、海量高维数据规约、作战效能评估; 赵恩来(1985-),男,江苏淮安人,硕士研究生,主要研究方向:聚类分析、时间序列数据挖掘; 刘玉栋(1983-),男,河南周口人,硕士研究生,主要研究方向:作战指挥训练模拟、军事建模与仿真; 黄亚(1983-),男,江苏盐城人,硕士研究生,主要研究方向:作战指挥训练模拟; 刘军涛(1983-),男,山东烟台人,硕士研究生,主要研究方向:运动实体数据采集、数据集成。

针对上述两个基本问题(异构数据加密、灵活高效的查询),本文提出了一种新的数据库加密方式:以领域元数据为支撑,综合集成密文索引组件、对象—关系数据库加解密映射组件(Cryption-Object Relation Mapping, Crypt-ORM),实现了异构数据库加解密系统。该系统可以对异构数据库中的数据进行统一加密处理;可以处理所有常见的数据类型,支持字符串精确匹配、字符串模糊匹配、字符串范围查询、数值型数据精确查询、数值型数据范围查询、数值型数据统计查询等灵活多样的密文查询,并且可以防止推断攻击,从而提高了数据库加密的兼容性、灵活性和安全性。

1 系统体系结构

数据库加密系统的加密方式,可以归纳为以下三种:客户端加密、服务器端加密以及客户端与服务器端混合加密[9]。本文实现的异构数据库加解密系统是基于服务器端加解密的,其体系结构如图1所示。

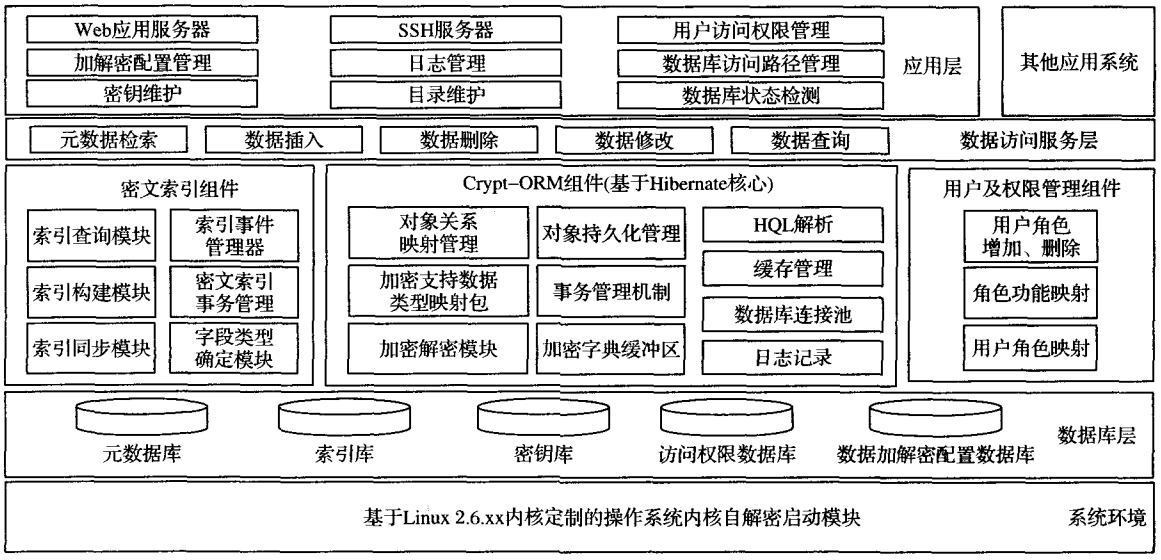

图1 异构数据库加解密系体结构

该系统主要分为五大部分:系统环境、系统支撑数据库、系统组件、数据访问服务层和应用层。

系统环境包括以 Linux 2.6.xx 为内核的嵌入式操作系统和自解密启动模块两大部分,为整个系统提供基础运行环境。整个服务器端软件以一个软件包的方式存储在 CF(Compact Flash)卡中。系统开机时,自解密启动模块主要用于将存储在 CF 卡中的软件包解密,将解密的整个操作系统所有相关的应用程序以内存盘的方式加载到内存中,这样系统正常启动后,则不需要再访问 CF 卡,从而极大地提高了系统的性能和稳定性。

系统支撑数据库主要是指系统运行所依赖的数据库集合,共计6个不同数据库。其中元数据库主要保存所有数据实体的元数据信息,并且以密文形式存储数值型数据的统计特征值,包括最大值、最小值、平均值和方差等;索引库主要用于辅助实现数据快速密文检索功能;密钥库保存系统中所有用于敏感数据加密的可用密钥;访问控制数据库主要用于存储合法用户信息以及用户的权限、密级等信息;数据加解密配置数据库主要存储所有目标数据库敏感数据的加密配置信息,包括指定目标库数据实体属性的安全级别、是否需要加密、加密使用的算法和密钥等信息。

系统组件主要包括三部分,分别是 Crypt-ORM 组件、密文索引组件两大核心模块和用户及权限管理组件,用于实现系统的核心业务功能。其中:Crypt-ORM 组件主要用于进行实体对象与关系表映射以及加解密转换功能;密文索引组件用于对密文数据库的密文索引构建以及快速检索与定位;用户及权限管理组件主要用于用户访问权限的认证以及数据库访问对象的安全等级设定等功能。

数据访问服务以 Web Service 方式提供对密文数据库服务器的访问服务。当一个应用需要访问数据库时,先连接至数据加解密系统,通过身份认证后,调用数据操作接口访问和操作目标数据库,访问结束后断开与系统的连接。

应用层是加解密系统中的独立任务。整个系统主要有9个任务,其中 Web 服务器是系统的主要网络服务器系统,用于实现用户 Web Service 访问接口和其他子系统功能;SSH 服务器主要用于远程管理系统;用户访问权限管理实现用户维护、权限维护以及授权;加解密配置管理用户设置目标数据库中数据的安全需求;日志管理用户记录、查看和维护系统业务日志信息;数据库访问路径管理主要管理目标数据库的访问信息,包括路径、认证等信息;密钥维护用于维护系统的加解密算法和所有预先分配的密钥;目录维护主要获取和查询所有目标数据库的元数据;网络服务和配置管理主要用于启动和停止系统运行的其他应用、配置各应用的基本参数。这些子应用通常通过调用系统组件以实现各自的功能。其他应用系统可以通过数据访问服务,对事先进行授权的目标数据库进行访问。

在上述系统体系结构中,Crypt-ORM 组件屏蔽异构数据库,实现了数据的透明访问;密文索引组件实现了对密文数据库中密文数据的高效灵活多样的密文查询,是系统实现过程中的两大难点,本文主要对 Crypt-ORM 组件和密文索引组件进行了详细的设计与分析。

2 Crypt-ORM 组件

在整个加解密系统的体系结构中,为数据访问层提供服务的 Crypt-ORM 组件是整个加解密系统的核心组件,它屏蔽了异构数据库,实现了数据的透明访问。Crypt-ORM 是基于当前开源社区较流行的 ORM 映射工具 Hibernate[10] 为原型进行改造实现的支持动态加密配置的对象—关系映射工具包。Crypt-ORM 组件工作原理如图2所示。

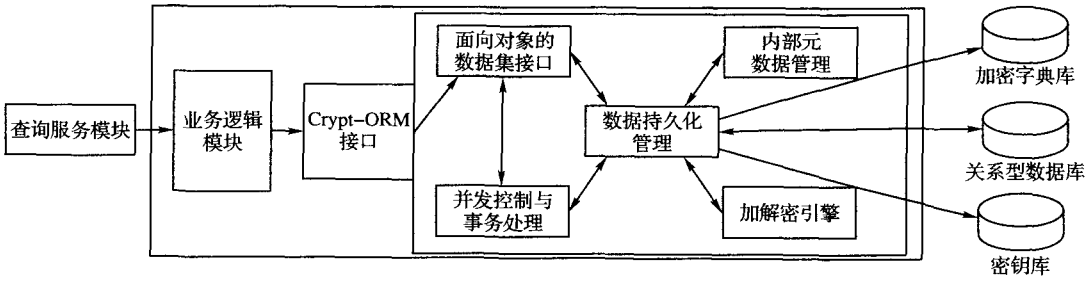

图 2　Crypt-ORM 组件原理

整个系统数据访问流程如图 2 所示,基于本系统开发的应用程序通过加解密系统提供的 Web Service 二次开发接口对数据库中的数据实体进行访问,请求经过转换由加解密映射转换中间件进行处理,中间件负责处理对象关系转换、对象封装,同时根据加密配置信息协同加密密钥库、用户权限库以及密文索引等信息,以调用方透明的方式对数据实体进行加解密。Crypt-ORM 组件屏蔽异构数据库流程如图 3 所示。

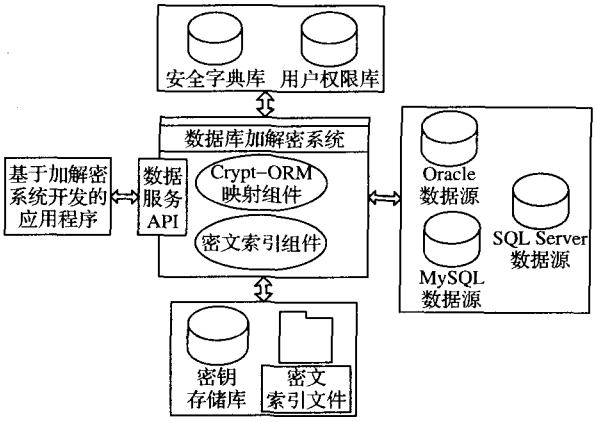

图 3　Crypt-ORM 工作原理

从图 3 中可以看出,客户端直接通过加解密系统暴露的接口存取所需数据,无需关心数据库是在本地还是远程计算机上,无需关心数据库系统运行的操作系统和所处的网络结构,无需关心数据库如何存储,真正实现了数据的透明访问。

3　密文索引组件

3.1　查询理论分析

对密文数据进行检索是加解密系统必须满足的功能,在数据被加密之后,原有明文序列的特性将会被打破,对于加密数据的查询变得不易。本文提出了一种新的设计思路来实现数据实体属性值的密文索引构建与查询,系统利用 Lucene 索引机制[11]在数据实体对象插入数据库的同时同步建立索引,建立密文索引过程与对象插入数据库在同一事务中完成,对数据实体进行查询时再根据密文索引进行记录的快速定位。该策略无需对数据库定义进行较大变化与调整,解决了数据经加密后不支持模糊匹配查询、范围查询以及失去排序能力的问题。

属性类型大致可以分为两大类:字符型和数值型。其中日期型和时间型通过转换成字符型进行处理。对于字符型数据的加密及快速查询采用崔宾阁等人提出的两阶段密文快速查询方法[8];对于数值型数据的加密及快速查询采用D. H. Lee等人提出的桶号、余数和密钥组合加密的方法[12]。可具体参考文献[8,12],获取详细的加解密和快速查询策略。

密文索引组件基于元数据库,并采用修改之后的上述两种加密理论,在加解密系统中进行具体实现。它将字符串数据的划分值和特征值,或数值型数据的桶号和余数,合并于同一个 Lucene 索引字段中。

在对密文数据进行查询时,Crypt-ORM 组件首先利用查询条件值的密文索引值在密文索引文件内进行一次粗糙的查询,返回初步满足查询条件的记录集合,然而在此记录集合上再执行精确的查询,从而获取查询的最终目标。此方式可以避免对全库数据进行解密查询,从而提高了密文数据查询效率,具体的查询原理如图 4 所示。

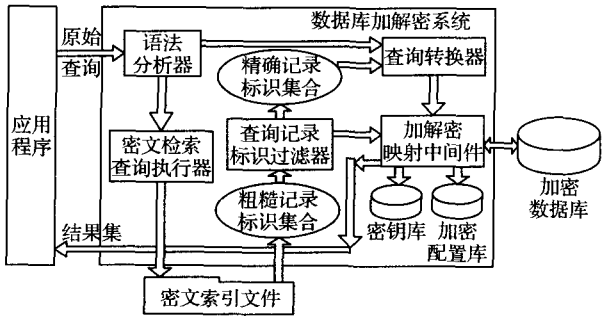

图 4　密文查询原理

3.2　密文检索的构建与查询

利用密文索引构建器对持久化对象进行密文索引构建之前,必须在加解密转换中间件的 XML 配置文件内进行相应的配置。密文索引构建器可以在对象持久化事务过程中被执行,支持对象插入以及对象更新时索引结构的同步,其缺点是在事务执行期间需要进行较多的磁盘 I/O 操作。在支持同步索引构建的同时,系统还支持索引的异步构建,可以在数据库中的数据被更新之后的某个时间对选定实体手工执行密文索引构建。图 5 描述了数据实体对象在插入过程中的同步密文、索引构建过程,数据实体对象在存储过程中需要利用配置的工作密钥对相应的属性值进行加密,而工作密钥在密钥库中是以密文的形式存放的,加解密映射中间件首先必须利用主密钥对工作密钥进行解密,然后利用明文工作密钥对数据实体对象待加密的属性值进行加密并保存,返回对应于数据库记录的 id。其次,密文索引构建器根据要加密字段类型选择划分值/特征值计算函数或桶号/余数计算函数;对具有查询需求的属性进行密文索引值的构建,并结合生成的数据库记录的 id 生成 Lucene 索引;接着密文索引构建器将索引存放于密文索引文件中;同时以密文形式在元数据库中存储该属性的统计信息。至此,整个数据实体存储过程完成,包括实体存储以及密文索引的存储。其过程如图 5 所示。

图 6 描述了利用密文索引对数据实体进行查询的过程。加密系统首先根据密文查询的方式选择调用模块:数值型密文统计查询直接查询事先以密文形式存储在元数据库中的统计数据;其他方式的密文查询根据加密属性的类型调用不同的计算函数,将查询条件语句进行分解,并计算查询条件值的密文索引值,利用 Lucene 索引机制对索引进行检索,返回经

过滤后符合查询条件的精确 id 集合，密文检索器将原始查询条件语句转化为对 id 集合的 in 查询，最后加密系统利用加解密映射转换中间件执行修改过的查询语句并对查询结果进行解密并对象化，最后以对象集合的形式返回查询结果集。

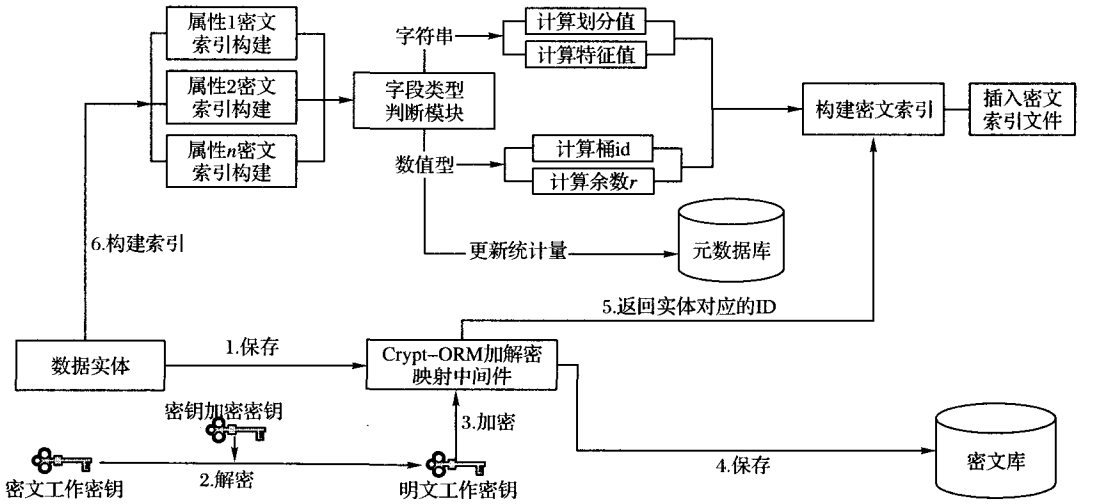

图5 数据实体索引构建

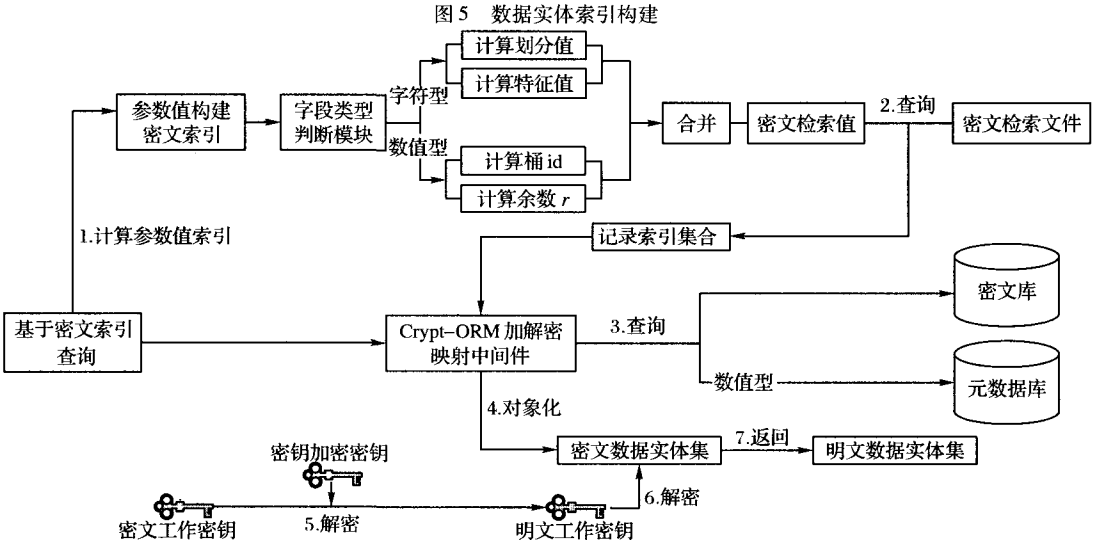

图6 数据实体密文索引查询

4 实验及系统安全性分析

4.1 实验

本文采用 Load Runner 工具对加解密系统的数据访问服务进行测试，主要对影响加解密系统性能的因素利用实验手段进行了测试与分析。

1）对于特定的数据集，不同类型的数据库产品在实施数据实体加密存储时，其性能表现如图7所示。

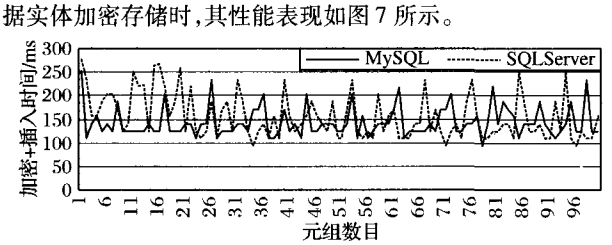

图7 不同类型数据库的性能表现差异

从图7可以看出，当后台数据源分别是 MySQL 和 SQLServer 时，加解密系统的性能表现优良；服务器运行比较稳定；系统可以对不同类型的数据库进行加解密；大部分时间消耗在数据的加密之上。

2）图8演示了数值型密文数据的范围查询，查询 Book 实体类的 price 属性在 300～400 的查询，结果如图8所示。

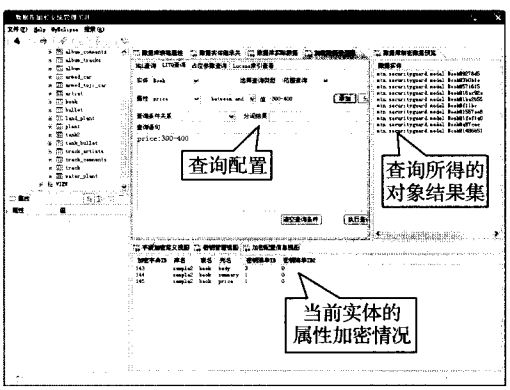

图8 数值型密文数据范围查询视图

4.2 系统安全性分析

本文主要从两个方面对加解密系统的安全性进行分析，即加密算法的安全性和密文索引组件的安全性。

4.2.1 加密算法的安全性

系统采用的加密算法在国际上被广泛认可，保证了加密的有效性，同时工作密钥与被加密数据分布在不同的两台服务器中，且密钥服务器中的工作密钥是经主密钥加密之后被存入数据库中的，主密钥亦通过密钥分离方式进行保密存放。即使攻击者通过某些途径获取了密文数据库中的部分数据，但由于无法获知明文状态的工作密钥，无法对数据实施解密。

除此之外,系统在保证密钥物理存储安全性的同时,对同一记录的不同字段采用了不同的加密方式进行保护,增加了攻击者破译密文的难度。

4.2.2　密文索引组件的安全性

加解密系统在数据实体进行透明加密存储时,对实体属性的明文值利用分桶划分函数以及特征值提取函数计算得出与之对应的密文索引值,密文索引的构建过程并不依赖字段加密所采用的加密算法,所以实体数据的密文索引构建过程是独立的。如果实体属性值发生变化,必须同步更新 Lucene 构建的密文索引结构,否则可能导致密文索引值与对象属性值不一致的现象。数据库加解密系统的密文索引构建器保证了索引构建事务过程与 Crypt-ORM 的事务保持一致,在数据保存、修改以及删除等服务接口实现过程中,始终保持事务的一致性,克服了密文索引与明文属性值的不一致现象。系统在构建密文索引的过程中,分桶划分函数所依托的桶划分元数据信息是严格保密的,同时其特征值计算函数是单向的,其计算过程是非可逆的,攻击者即使获取了密文索引文件,也无法根据密文索引值反向生成明文值。通过以上的分析,可知加解密系统对于数据采用的基于 Lucene 密文索引构建查询方式是能够满足网关系统的安全性需求的。

5　结语

本文详细阐述了异构数据库加解密系统的设计与核心组件的实现,解决了异构数据库加解密、字符串数据和数值型数据灵活多样的查询,最后对其安全性进行了分析。当然,保护一个数据库系统的安全,仅仅依靠数据库加密技术是远远不够的,严格用户身份认证、数据安全的存取控制模型、细粒度的访问控制权限设置以及防火墙的有效配置都是数据库安全不可缺少的有效措施。数据库加密虽然在一定程度上影响了数据库操作的性能以及可用性,但对于业务数据的安全来说,是非常必要的。在下一步的工作将继续完善密文检索策略、优化查询转换的实现方法。

参考文献:

[1] 赵晓峰,叶震. 几种数据库加密方法的研究与比较[J]. 计算机技术与发展,2007,17(2):219 - 223.

[2] 朱勤,骆轶姝,乐嘉锦. 数据库加密与密文数据查询技术综述[J]. 东北大学学报:自然科学版,2007,33(4):543 - 549.

[3] 王迪,刘国华,于醒兵. 基于多重桶划分的密文索引技术[J]. 电子技术应用,2007(3):141 - 144.

[4] 王迪,刘国华,于醒兵. 基于最佳桶划分策略的密文索引技术[J]. 小型微型计算机系统,2008,29(4):649 - 653.

[5] 王正飞,汪卫,施伯乐. 加密数据的一种高效查询方法[J]. 计算机工程与应用,2008,44(12):30 - 33.

[6] 李亚秀,刘国华. 数据库中字符数据的加密方法[J]. 计算机工程,2007,33(6):120 - 123.

[7] 于涵,赵亮,徐伟军,等. 一种新的数据库加密及密文索引方法研究[J]. 电子学报,2005,33(12):2539 - 2543.

[8] 崔宾阁,刘大昕,王桐. 支持快速查询的数据库加密方法研究[J]. 计算机科学,2006,33(6):115 - 118.

[9] 黄小栗. 基于 B/S 结构的数据库加密研究[D]. 郑州:郑州大学,2005.

[10] BAUER C, KING G. Hibernate 实战[M]. 杨春花,彭永康,俞黎敏,译. 北京:人民邮电出版社,2008.

[11] 邱哲,符滔滔. 开发自己的搜索引擎:Lucene 2.0 + Heritrix[M]. 北京:人民邮电出版社,2007.

[12] LEE D H, LEE S M, NAM T Y, *et al*. A bucket ID transformation scheme for efficient database encryption [C]// ICOIN2008: International Conference on Information Networking. Busan: IEEE Computer Society, 2008: 1 - 5.

(上接第 2338 页)

后才能返回 Top-k 个结果,而基于 BDBF 只需要查询一部分,并且随着 k 值的增大而两种所需时间趋于接近。

经过以上分析可知,采用 BDBF 后的多关键字查询对查询的准确率影响不大,却较大地减少了网络流量的产生和查询所需的时间,特别是 k 值较小时这种效果更为明显。

5　结语

针对 P2P 多关键字查询问题,本文提出了一种 BDBF 的 Bloom Filter 以减少网络流量。该方法既能较好适应数据量递增的 P2P 应用环境,又能高效地在 P2P 环境中进行多关键字的 Top-k 查询。

参考文献:

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

[2] CHEN H, JIN H, WANG J, *et al*. Efficient multi-keyword search over P2P Web [C]// Proceedings of the 17th International Conference on World Wide Web. New York: ACM Press, 2008: 989 - 998.

[3] GUO D, WU J, CHEN H, *et al*. Theory and network applications of dynamic Bloom filters [C]// 25th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies. Barcelona: IEEE Press, 2006: 1653 - 1664.

[4] FAN L, CAO P, ALMEIDA J, *et al*. Summary cache: A scalable wide-area Web cache sharing protocol [J]. ACM Transactions on Networking, 2000, 8(3): 281 - 293.

[5] BONOMI F, MITZENMACHER M, PANIGRAHY R, *et al*. An improved construction for counting Bloom filters [C]// Proceedings of the 14th Conference on Annual European Symposium, LNCS 4168. Berlin: Springer, 2006: 684 - 695.

[6] FICARA D, GIORDANO S, PROCISSI G. MultiLayer compressed counting Bloom filters [C]// 27th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies. Phoenix: IEEE Press, 2008: 311 - 315.

[7] SAAR C, YOSSI M. Spectral Bloom filters [C]// Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data. San Diego: ACM Press, 2003: 241 - 252.

[8] AGUILAR-SABORIT J, TRANCOSO P, MUNTES-MULERO V. Dynamic count filters [J]. SIGMOD Record, 2006, 35(1): 26 - 32.

[9] 肖明忠,代亚非,李晓明. 拆分型 Bloom Filter[J]. 电子学报,2004,32(2):241 - 245.

[10] MITZENMACHER M. Compressed Bloom filters [J]. IEEE/ACM Transactions on Networking, 2002, 10(5): 604 - 612.

[11] 严华云,关佶红. Bloom filter 研究进展[J]. 电信科学,2010,26(2):31 - 36.

[12] Wikipedia. Median [EB/OL]. [2009 - 12 - 01]. http://en.wikipedia.org/wiki/Median/.

[13] SINGHAL A. Modern information retrieval: A brief overview [J]. IEEE Data Engineering Bulletin, Special Issue on Text and Databases, 2001, 24 (4): 35 - 43.

[14] LIU HONGBIN. Bloom filter [EB/OL]. [2009 - 08 - 23]. http://wwwse.inf.tu-dresden.de/xsiena/bloom_filter.

[15] 邱哲,符滔滔. 开发自己的搜索引擎[M]. 北京:人民邮电出版社,2008.