利用数字水印技术实现数据库的版权保护

牛夏牧,赵 亮,黄文军,张 慧
(哈尔滨工业大学,信息对抗技术研究所、(深圳)研究生院信息安全技术研究中心,黑龙江哈尔滨 150001)

摘 要: 数字水印技术是实现多媒体数据版权保护的一种有效手段. 随着关系型数据库的广泛使用,也随之产生了在关系型数据库中嵌入水印信息的需求. 通过在关系型数据库中所嵌入的版权信息,可以将数据库与其拥有者联系起来,从而实现数据库的版权保护. 本文结合现有的关系型数据库水印算法,提出了一种可以在数据库中嵌入具有实际意义字符串的水印算法,并介绍如何通过调整算法中的参数,实现不同的鲁棒性和不可见性需求. 在 Oracle 数据库中所进行的仿真实验结果表明,该算法可以在实际数据库中得到很好的应用.

关键词: 数字水印; 版权保护; 关系型数据库
中图分类号: TN919 **文献标识码**: A **文章编号**: 0372-2112 (2003) 12A-2050-04

Watermarking Relational Databases for Ownership Protection

NIU Xia-mu, ZHAO Liang, HUANG Wen-jun, ZHANG Hui
(*Dept of Automatic Test and Control, Harbin Institute of Technology, Harbin, Heilongjiang* 150001, *China*)

Abstract: Watermarking technique has been widely applied to multimedia data as an effective method for ownership protection. The increasing use of relational databases is creating a similar need for applying this technique to databases in order to associate ownership of these databases with their owners. This paper presents a robust algorithm for embedding a string of bits into a database relation. by improving the current relational database watermarking algorithms. We also show how to adjust parameters according to different robustness and imperceptibility requirements. Using implementations running on Oracle, we demonstrate that the performance of our algorithm allows for its use in real world applications.

Key words: watermarking; ownership protection; relational databases

1 引言

数字水印技术[1]是网络环境下保护版权的新型技术,可以确立版权所有者,识别购买者或者提供关于数字内容的其他附加信息,并将这些信息以人眼不可见的形式嵌入在数字图像,数字音频和视频序列中,用于确认所有权和跟踪行为. 数字水印作为一种新兴的信息安全技术已经被许多应用领域所采用.

关系型数据库[2]是以二维平面表作为数据模型的数据库系统. 关系数据库技术出现在 20 世纪 70 年代、经过 80 年代的发展到 90 年代已经比较成熟. 目前,无论是 Oracle 公司的 Oracle 9i、IBM 公司的 DB2、还是微软的 SQL Server 等都是关系型数据库. 随着关系型数据库的广泛使用,也随之产生了在关系型数据库中嵌入水印信息的需求. 同多媒体数据一样,数据库也面临着版权保护的问题. 比如说,对于那些提供信息服务(如气象信息、医疗信息、人才市场信息、股票交易信息、电子元器件参数信息等等)的公司,其主要资产便是存储于数据库里的大量数据. Internet 的快速发展将促使这些数据供应商提供远程访问其数据库的服务,用户在支付一定的使用费之后便可以远程登陆数据库,使用里面的数据. 虽然远程登录服务将会为终端用户提供极大的方便,但数据供应商也同时面临着数据被窃取的危险. 如果不法分子将他从数据库里的获取的大量数据转卖给他人,这些信息公司必然会蒙受很大的经济损失. 此外,随着数据库技术的不断发展,数据库中存储的数据量急剧增大,在大量的数据背后隐藏着许多重要的信息,利用数据挖掘技术可以从看似无规律的数据中挖掘出有用的商业信息,最典型的例子就是一家连锁店通过数据挖掘发现了小孩尿布和啤酒之间有着惊人的联系. 因此,也需对这种隐含有重要商业信息的大型数据库进行版权保护,以防止不法分子从中获利. 通过在关系型数据库中嵌入代表所有权的水印信息,可以将数据库与其拥有者联系起来,从而实现数据库的版权保护. 由此可见,利用数字水印技术实现数据库的版权保护,有着很重要的学术意义和应用价值.

2 一位水印信息的嵌入和验证

数字水印技术主要是利用数字产品的数据冗余,将产权等信息作为附加噪声融合在原数字产品当中以嵌入隐藏信息. 数据库关系中元组的数值型属性值同样也存在着冗余[3],

收稿日期:2003-10-08;修回日期:2003-12-26
基金项目:国家自然科学基金(No. 60372052);全国博士学位论文作者专项资金资助项目(No. FANEDD-200238);哈尔滨工业大学跨学科研究基金(No. HIT. MD-2002. 11)

通过在某些数值型属性值中引入少量误差,对其最低有效位(Least Significant Bits)进行位操作,则可实现水印信息的嵌入.同时,将引入的误差控制在某个约束范围以内,并不会影响数据库的使用价值[4].

设待嵌入水印的数据库关系用 $R(P, A_1, \ldots, A_v, \ldots)$ 表示,其中 P 为主码,$A_1, \ldots, A_v$ 为 v 个可嵌入水印的数值型属性列(不包括主码),R 由 n 个元组 $r_1 \ldots r_n$ 组成,每个元组 r 都有 1 个主码 $r.p$ 和 v 个数值型属性值 $r.A_1, \ldots, r.A_i, \ldots, r.A_v$.这样,便可通过更改这 $n \times v$ 个属性值的最低有效位,实现水印的嵌入.

但是,并不是所有这些数值的最低有效位都可以在同样的范围内更改.为了保证数据库关系的使用价值不被破坏,在嵌入水印之前需要引入数值可更改范围的约束限制.仅对满足该约束限制的属性值嵌入水印信息.比如说,如果每个属性值仅能在 0.1%范围内变化而不影响整个数据库关系的使用价值,那么整数 1000 仅有一位最低有效位可以被更改,而小于 1000 的整数则不能嵌入水印信息.设 $b\%$为所有属性值可允许的变化范围,可以计算出每个属性值最低有效位可更改的范围.

在嵌入水印之前,需要首先对属性值进行编号,依据编号的性质嵌入水印.每一个属性值的编号取决于其属性名及所属元组的主码.为了实现算法的鲁棒性,属性值的编号还应该取决于某个密钥值 K,即只有知道 K 才能正确计算编号的大小.利用单向 hash 函数[5]计算出的编号可以满足以上要求.对于属性值 $r.A_i$,属性名为 A_i,所在元组的主码为 $r.P$,其编号为 $index = hash(K, r.P, A_i)$.

选取参数 λ,控制需要嵌入水印信息的属性值在所有可嵌水印的属性值中所占比例,仅对编号能被 λ 整除的属性值嵌入水印.对于数据库关系 R 中的每一个数值型属性值 $r.Ai$,嵌入算法的具体步骤描述如下:

(1)计算 $r.A_i$ 在约束 $b\%$下最低有效位可更改的范围 ε.

(2)若 $\varepsilon \neq 0$,计算 $index = hash(K, r.P, A_i)$.否则返回步骤 1,计算下一个属性值.

(3)若 $index$ 能被 λ 整除,按照步骤 4 在属性值 $r.A_i$ 中嵌入水印信息,否则返回步骤 1,计算下一个属性值.

(4)设 $j = index \bmod \varepsilon + 1$,若 $hash(index, K)$ 为偶数,将 $r.A_i$ 的第 j 位最低有效位置 0,否则,将 $r.A_i$ 的第 j 位最低有效位置 1.

(5)返回步骤 1,计算下一个属性值,直到计算完关系 R 中的所有数值型属性值.

需要说明的是,属性 $r.A_i$ 的第 j 位最低有效位可能在加入水印信息之前就已经是加水印所要设定的值,理论上,在步骤 4 中 $r.A_i$ 的值被改变的概率是 1/2.

嵌入算法的实质是在所选属性值的特定比特位中嵌入一种匹配关系,即使得该属性值编号的 $hash$ 值(奇/偶)与其某一最低有效位的值(1/0)相匹配.验证算法用来验证这种匹配关系是否存在.设 ω 为怀疑存在该匹配关系的属性值的个数,τ 为从这 ω 个属性值中检测出的实际匹配数.验证算法通过 τ 和 ω 之间的关系来判定数据库关系中是否嵌有水印.利用参数 $totalcount$ 和 $matchcount$ 做累加器,计算 ω 和 τ 的值.初始 $totalcount = matchcount = 0$,对于数据库关系 $R(P, A_0, \ldots, A_{v-1}, \ldots)$ 中的每一个数值型属性值 $r.A_i$,进行如下操作:

(1)计算 $r.A_i$ 在约束 $b\%$下可更改的最低有效位范围 ε.

(2)若 $\varepsilon \neq 0$,计算 $index = hash(K, r.P, A_i)$.否则返回步骤 1,计算下一个属性值.

(3)若 $index$ 能被 λ 整除,$totalcount$ 的值增 1,按照步骤 4 在元组 r 中检测匹配信息.若 $index$ 不能被 λ 整除,则 $totalcount$ 的值不变,返回步骤 1,计算下一个属性值.

(4)设 $j = index \bmod \varepsilon + 1$,$t = r.A_i$.若 $hash(index, K)$ 为偶数,将 t 的第 j 位最低有效位置 0,否则,将 t 的第 j 位最低有效位置 1.若 t 的值仍等于 $r.A_i$,$matchcount$ 增 1,否则,$matchcount$ 的值不变.

(5)返回步骤 1,检测下一属性值,直到计算完关系 R 中的所有数值型属性值.

(6)依次检测完所有属性值后的 $totalcount$ 值即为怀疑存在匹配关系的属性值的个数 ω,$matchcount$ 的值等于检测出的实际匹配数 τ.

步骤 4 实际上是检测 $r.A_i$ 是否满足预设的匹配关系.这里需要注意的是,即使没有嵌入水印,$r.A_i$ 第 j 位最低有效位也可能与 $hash(index, K)$ 值相匹配,其概率为 1/2.也就是说,利用此验证算法对于没有嵌入水印信息的属性值进行验证,步骤 4 中 $matchcount$ 的值也可能增 1,其概率为 1/2.

对嵌入了水印的数据库关系进行检测,理论上 $\omega = \tau$.考虑到数据库里的数据可能在嵌入水印后受到攻击,导致一部分水印信息(匹配关系)丢失,所以 $\tau \leq \omega$.因此在提取算法中引入阈值参数 $\tau_{\min}$,只要 $\tau \geq \tau_{\min}$,即可判定数据关系中嵌入了水印信息.$\tau_{\min}$的大小取决于 ω 和一个显著度参数 α.下面对参数 $\tau_{\min}$的选取进行分析.

3　算法分析

在上一节里曾经提到,即使数据库关系中没有嵌入水印,利用水印验证算法验证数据库关系中的元组,也可能从某一个待检定的元组中检测出匹配信息来,概率为 1/2.从概率统计的角度分析,其分布规律为 0-1 分布(也称贝努里分布或两点分布).若从 ω 个元组中检测出了 τ 个元组含有匹配信息,实际上可以看作是一个成功 τ 次,失败 $\omega - \tau$ 次的 ω 重贝努里实验,其概率为:

$$b(\tau, \omega, \frac{1}{2}) = \binom{\omega}{\tau}\left(\frac{1}{2}\right)^{\tau}\left(\frac{1}{2}\right)^{\omega-\tau} = \frac{\omega!}{\tau!\,(\omega-\tau)!}\left(\frac{1}{2}\right)^{\omega} \tag{1}$$

设 $\theta = \tau/\omega$,θ 的概率分布满足二项分布,分布曲线如图 1 示.

由图 1 可知,如果数据关系中没有嵌入水印,$\theta = 0.5$(即 $\tau = 0.5\omega$)的概率最大,并且 θ 值集中分布于 0.5 附近,θ 值接近 1 的概率很小(近似于 0).θ 分布于区间 $[\tau_0/\omega, 1]$(即 τ 分布于区间 $[\tau_0, \omega]$)的概率为:

$$B(\tau_0, \omega, 1/2) = \sum_{\tau=\tau_0}^{\omega} b(\tau, \omega, 1/2)$$

$$= (1/2)^{\omega} \sum_{\tau=\tau_0}^{\omega} \frac{\omega!}{\tau!\ (\omega - \tau!)} \qquad (2)$$

选取显著度参数 $\alpha(0 < \alpha < 1)$,满足 $B(\tau_0, \omega, 1/2) \leq \alpha$ 的 τ_0 的最小值即为 $\tau_{\min}$. 设 $\theta_{\min} = \tau_{\min}/\omega$,依据概率统计中的"小概率原理",当 α 值很小时,θ 值分布于区间 $[\theta_{\min}, 1]$ 这一事件为一小概率事件(或实际不可能事件). 也就是说,如果数据库关系中没有嵌入水印,θ 的值几乎不可能落到区间 $[\theta_{\min}, 1]$ 内. 嵌入水印的实质即为改变 θ 值的概率分布,使其分布于二项分布的小概率区间. 因此,我们可以把 θ 值是否位于小概率区间 $[\theta_{\min}, 1]$ 内作为判定数据库关系中是否嵌入水印的标准,即使数据库关系在嵌入水印后受到攻击,部分水印信息丢失,使得 $\theta < 1$,但只要 $\theta \geq \theta_{\min}$,就可判定数据库关系中嵌入了水印.

4 鲁棒性分析及参数选择

本文所介绍的数据库数字水印技术的目的是为了实现数据库的版权保护,即通过数据库里的水印信息证明数据库的所有权,从而在与非法盗用者的法律纠纷中获得胜诉. 非法盗用者盗用数据库的目的是为了利用数据库里的数据获利,因而不会对数据库进行较大的改动,否则数据库对他也就失去了应有的价值. 因此本文主要考虑以下攻击[4]:

(1) 子集选取:盗用者只盗用数据库的一部分数据.

(2) 子集增加:盗用者盗用了整个数据库或数据库的一部分,同时又在数据库中增加了一部分数据.

(3) 更改子集:盗用者盗用了整个数据库或数据库的一部分,并更改了其中的一部分数据. 所有这些攻击的实质都是导致部分属性值的匹配信息丢失,使得 τ 值变小. 但只有当 τ 值小于 $\tau_{\min}$ 时,攻击才算是成功的. 设鲁棒系数 $Rb = (\omega - \tau_{\min})/\omega = 1 - \theta_{\min}$ 表示数据库关系中所嵌入的匹配信息可容许的最大改动比例,显然 $\theta_{\min}$ 越小,Rb 值越大,水印的鲁棒性就越好. 图 2 和图 3 分别给出了改变 ω 和 α 值对于系数 Rb 的影响.

由图 2 可以看出,Rb 值随验证的显著度 α 值增加而增加,因此增大 α 值也会增强水印的鲁棒性. 由图 3 可知,嵌入匹配信息数 ω 越多,则 Rb 值越大,水印的鲁棒性就越好。

增大 α 值会增强水印的鲁棒性,但是 α 值并不是越大越好. 从没有嵌入水印的数据库关系中验证出了水印,称为误报. 显然对于未嵌入水印的数据库关系,误报的概率即为 τ 值落在区间 $[\tau_{min}, \omega]$ 的概率 α. 因此,α 值越大,误报的可能性也就越大,相反,取较小的 α 值,则会减小水印系统的误报律.

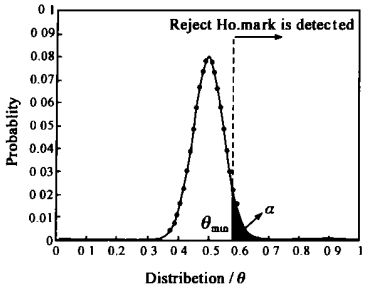

图 1 未嵌入水印的数据库关系中 θ 值的概率分布曲线

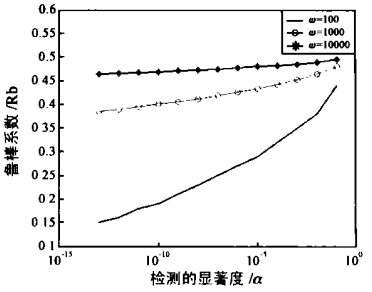

图 2 改变 α 值对于鲁棒性的影响

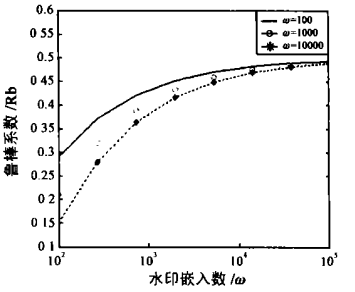

图 3 改变 ω 值对于鲁棒性的影响

同样,增大 ω 可以增强水印的鲁棒性. 但是水印的嵌入是通过在元组的属性值里引入误差实现的. ω 值的增加将给数据库关系带来更大的误差,ω 值过大会对数据库的使用造成影响,这样便破坏了数据库水印的"不可见性".

需要说明的是,对于固定的数据库关系,可嵌入水印的属性值数目 n 一定,ω 值的大小取决于参数 λ,由于 hash 函数的结果均匀分布,因此 $\omega \approx n/\lambda$. 这样,可以通过调整参数 λ 的大小得到合适的 ω 值.

综上所述,对于每个特定的数据库关系,水印系统应综合鲁棒性、误报律、系统误差等多方面因素选取合适的参数值.

5 嵌入和提取多位水印信息

前文所述的水印算法是一种验证算法,验证的结果只能是"有"或"无"的信息,而不是有一定具体意义的水印信息. 显然,利用这种信息证明数据库的版权所有,并不是十分令人信服的. 如果能在数据库中嵌入一些有实际意义的比特值(比如代表公司名称的字符串),水印的使用价值会更高.

实际上,可以把验证算法的结果看成 1 位比特信息,如果有水印,结果为"1",没有水印,结果则为"0",也就是说,水印的嵌入算法相当于在数据库关系中嵌入了一个"1". 考虑到原算法只利用了数据库关系中的一个很小的子集(编号可以被 λ 整除的元组的集合),如果能够利用数据库关系的 m 个子集($m < \lambda$),每个子集中嵌入 1 位比特信息,便可嵌入一个长度为 m 的比特串.

多位水印信息的嵌入算法如图 4,首先对各个可嵌水印的属性值进行编号,然后利用编号模除参数 λ 的结果对各个属性值进行分组. 对于属性值 a_i,编号为 $index(a_i)$,设 $j = index(a_i) \mod \lambda + 1$,分组的结果使得 a_i 位于子集 s_j 中. 设水印为 W,长度为 $m(m < \lambda)$,水印的每一位为 $w_k (1 \leq k \leq m)$,依

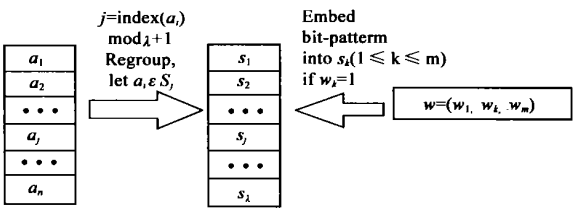

图 4 嵌入多位水印信息

据 w_k 是否为"1"决定是否在数据库关系的第 k 个子集中嵌入水印.也就是说,如果 $w_k = 1$,则按照第 2 节所描述的水印嵌入算法在子集 s_k 中嵌入水印(即在 s_k 的各个属性值中嵌入预定的匹配关系).

水印的检测算法如图 5 所示.同样,先将各个可嵌水印的属性值进行编号,再分成 λ 个子集 $s_1, s_2 \ldots s_\lambda$,最后按照第 2 节所描述的的水印验证算法验证各个子集中是否有水印信息.若验证 sk 的结果为"有"(即 s_k 中满足 $\tau \geq \tau_{\min}$),则水印的第 k 位 w_k 为 1,否则 w_k 为 0.

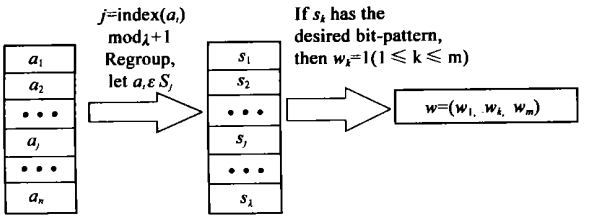

图 5　提取多位水印信息

6　仿真实验

仿真实验所用到的数据库关系由某地森林的植被信息所组成,数据库中包含了此森林的海拔、平面形状、坡度,距离水源及公路距离,土壤类型等参数的测量信息.数据库关系(表)中共有 100000 个元组,每个元组有 61 个属性,选取其中的 10 个数值型属性嵌入水印.仿真试验时,后台数据库系统使用 Oracle9i,前端编程环境使用 JBuilder8.0,通过 JDBC 连接到 Oracle 数据库.仿真试验时选取参数 $b\% = 0.1\%$, $\lambda = 1000$, $\alpha = 0.0001$,测得 172736 个属性值可嵌入水印.嵌入字符串"ownership protection"(每 8 比特表示一个字符,共 160 位).嵌入水印对数据库数据各属性列的整体影响见表 1,可以看出由嵌水印所引入的误差非常微小.

表 1　水印在各属性列所引入的总体误差

属性名	均值	方差	均值改变比例	方差改变比例
A1	2862.0	231.4	3.6 E-7	9.2 E-7
A2	138.1	103.8	0.0	0.0
A3	11.8	6.5	0.0	0.0
A4	260.6	202.8	0.0	0.0
A5	35.2	42.6	0.0	0.0
A6	3344.2	1776.8	2.5 E-6	4.3 E-6
A7	218.2	20.9	0.0	0.0
A8	225.5	16.7	0.0	0.0
A9	139.3	31.2	0.0	0.0
A10	3589.6	1781.4	0.0	0.0

对水印进行鲁棒性测试,模拟子集选取攻击,子集增加攻击和子集更改攻击,仿真结果分别见表 2,表 3,表 4.可以看出,本文提出的水印算法对于以上攻击都有较好的鲁棒性.

表 2　子集选取攻击所产生的误码率

选取比例	10%	12%	14%	16%	18%	≥20%
误码率	30.8%	17.0%	10.1%	2.5%	0.6%	0

表 3　子集增加攻击所产生的误码率

增加比例	≤75%	100%	125%	150%	175%	200%
误码率	0	0.6%	2.5%	4.4%	7.5%	10.7%

表 4　子集更改攻击所产生的误码率

更改比例	≤20%	25%	30%	35%	40%	45%
误码率	0	0.6%	0.6%	3.1%	5.0%	13.8%

7　结论

数字水印处理技术是解决数字产品知识产权问题的最有效和最有潜力的技术.本文所提出的水印算法可以在数据库中嵌入多位信息,并且通过实验证明算法具有较好的鲁棒性和不可见性,能够很好的解决数据库的版权保护问题.研究如何利用数字水印技术实现数据库的安全保护,以及在数据库关系的非数值型属性值中嵌入水印是本论文今后要继续开展的研究工作.

参考文献:

[1] Stefan Katzenbeisse, Fabien A P Petitcolas. Information Hiding Techniques for Steganography and Digital Watermarking [M]. Boston, London: Artech House, 1999.

[2] 萨师煊,王珊. 数据库系统概论 [M]. 北京:高等教育出版社, 1990.

[3] Rakesh Agrawal, Jerry Kiernan. Watermarking Relational Databases [A]. Proceeding of the 28th VLDB Conference [C]. Hongkong, China: 28 VLDB, 2002.

[4] Radu Sion, Mikhail Atallah, Sunil Prabhakar. Rights protection for relational data [A]. Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data San Diego [C]. California: ACM SIGMOD, 2003. 98 - 109.

[5] M Atallah, S Wagstaff. Watermarking with quadratic residues [A]. Proc. of IS&T/ SPIE Conference on Security and Watermarking of Multimedia Contents [C]. USA: SPIE, January 1999.

[6] B Schneier. Applied Cryptography [M]. USA: John Wiley & Sons, Inc. 1996.

[7] Radu Sion, Mikhail Atallah, Sunil Prabhakar. On watermarking numeric sets [A]. Proceedings of the Workshop on Digital Watermarking [C]. USA: IWDW. 2002.

作者简介:

牛夏牧　男,1961 年 5 月出生于辽宁省锦州市,教授,博士生导师,哈尔滨工业大学,信息对抗技术研究所所长、(深圳)研究生院信息安全技术研究中心主任,分别于 1982 年、1989 年、2000 年获哈尔滨工业大学学士、硕士、博士学位;2000 年至 2002 年在德国 Fraunhofer-IGD, Dept. of Security Technology in Graphics and Communication Systems 任研究员;2003 年作为特殊引进人才回国,主要从事研究的方向:(1) 信息处理及安全技术;(2) 信息隐藏与保密通信技术;(3) 信息对抗技术,共发表学术论文 70 余篇,其中 20 余篇被 SCI/ EI 收录,主编与参编著作 3 部,2002 年荣获全国百名优秀博士论文奖;2003 年荣获国防科工委百名优秀博士论文奖.