第 33 卷 第 14 期 计 算 机 工 程 2007 年 7 月
Vol.33 No.14 Computer Engineering July 2007
·软件技术与数据库· 文章编号：1000—3428(2007)14—0071—02 文献标识码：A 中图分类号：TP311

基于视图发布的信息泄漏分析与研究

宋 丽[1,2]，刘国华[2]，佟 冰[2]
（1. 中国银行北京分行，北京 100020；2. 燕山大学计算机科学与工程系，秦皇岛 066004）

摘 要：视图发布给数据交换带来了方便，在视图发布过程中可能造成信息泄漏，“发布视图的安全性”成为数据库安全的一个新课题。该文给出了“查询-视图”安全判定定理以及公开测量算法，分析了基于关键元组的消除信息泄漏算法。实验结果显示，该算法可以消除信息泄漏、保证视图安全。

关键词：数据库安全；视图发布；信息泄漏；关键元组

Analysis and Research of Information Disclosure Based on View Publication

SONG Li[1,2], LIU Guohua[2], TONG Bing[2]
(1. Beijing Branch, Bank of China, Beijing 100020;
2. Department of Computer Science and Engineering, Yanshan University, Qinhuangdao 066004)

【Abstract】Although it is convenient to exchange data by publishing view, some information may be disclosed in the publishing process, Therefore, it is a new issue of database security to ensure the publishing view safety. This paper proposes the theorem of query-view security determining and method of measuring disclosure, and gives algorithm of eliminate the information disclosure. Experimental results show that algorithms eliminate the information disclosure efficiently, and ensure the view security.

【Key words】 database security; view publication; information disclosure; critical tuple

视图发布给数据交换带来了方便，也带来了安全隐患[1,2]。数据交换往往涉及到多个合作伙伴，即使每个合作伙伴所发布的单个视图是安全的，也难以避免信息的泄漏。例如，一个公司如果需要同几个合作伙伴交换数据，就会发布公司生产信息的动态视图。假设视图V_1是面向供应商的，它包含关于某一产品配件的详细信息，视图V_2是面向零售商和顾客的，它包含产品特性、价格等方面的详细信息，视图V_3是面向税务代理公司的，它包含劳动力成本信息。作为商业机密，公司不想让外界知道自己产品的生产成本。如果能从视图V_1和视图V_3得到信息，并把它们组合到一起，那么该公司产品的生产成本就会被计算出来，这样就会泄漏商业机密。“视图发布的安全性”成为数据库安全的一个新课题[3~6]。

在视图发布过程中，防止信息泄漏的方法可分为 2 种[7]：(1)针对视图接收者的方法；(2)针对视图发布者的方法。前者要求视图接收者不把该信息用于其它方面，或有意、无意地泄露出去，加强安全防范措施，防止信息被窃取。后者要求视图发布者对所要发布的视图进行分析、评测，确认没有信息泄漏后，再传送给视图接收者。

在实际应用中，第 1 个方法很难实现，本文重点放在第 2 个方法上，保证在视图接收者接收视图之前，视图存在最小的“信息泄漏”。

1 查询视图安全的判定及测量

1.1 判定定理、信息公开测量中的概念

(1)D：出现在任何关系的任何属性中的所有值。

(2)$Tup(D)$：关系模式中所有关系上的所有元组，其中一个数据库示例 Ii 是 $Tup(D)$的任意子集。

(3)$N_{tup}(D)$：$Tup(D)$中的元组的个数，指关系表中各列的相异属性个数的乘积。

(4)$Q(I)$：就是将查询 Q 应用到数据库示例 I 上的结果。

(5)$N(I)$：表示所有可能的数据库示例集 I 中数据库示例的个数。

(6)$\overline{V}=V_1, V_2,\ldots,V_k$即一组视图，$S$是一个“秘密”查询。

(7)$\overline{V}_{(I)}=\overline{V}$ 指$V_1(I)=v_1 \wedge V_2(I)= v_2 \wedge V_3(I)= v_3 \cdots \wedge V_k(I)= v_k$

(8)I'：数据库示例集。

定义 1 对于所有满足条件的数据库示例集 I' 而言，将查询 S 作用在数据库示例集 I' 上，应答结果是 s 的概率为

$$P[S(I)=s]= Ns(I)/ Nv(I)$$

其中，$Ns(I)$表示 $S(I)=s$ 数据库示例集 I' 中数据库示例的个数；$Nv(I)$表示 $V(I)=v$ 数据库示例集 I' 中数据库示例的个数。

定义 2 对于数据库示例集 I' 而言，在视图组 V 的查询结果是 v 的前提下，秘密查询 $S(I)=s$ 的概率为

$$P[S(I)=s|V(I)=v]=Ns,v(I)/Nv(I)$$

其中，$Ns,v(I)$表示在数据库示例集 I' 中，S 和 V 共同的数据库示例个数。

定义 3 假定D是一个有限域，Q是一个查询。Qtup(Q)是基于$Q(I)=q$的元组集。在Q查询中的数据库示例集Iq，如果任意一个数据库示例$I \in Iq$，$Q(I-\{t\}) \neq Q(I)$，那么，一个元组$t \in Q$tup(D)对于Q来讲是十分关键的，Q的关键元组集被标记为$crit_D(Q)$。

基金项目：教育部科学技术研究基金资助重点项目(205014)
作者简介：宋 丽(1980－)，女，硕士研究生，主研方向：数据库安全技术，网格技术；刘国华，教授、博士生导师；佟 冰，硕士研究生
收稿日期：2006-09-20 **E-mail：**happyalicia@163.com

1.2 基于关键元组的查询—视图安全判定定理

定理 假设D为一个域，S是一个查询，$\overline{V}$是一个视图组，那么对于每一个概率分布P来讲，当且仅当$\mathrm{crit}^D(S)\cap\mathrm{crit}^D(\overline{V})=\emptyset$时，$S\mid\overline{V}$成立[8]。

1.3 基于关键元组的信息公开测量方法

定义 4 (信息泄漏)假定 S 和$\overline{V}$，其中 $s\subseteq S(I)$，$\overline{V}\subseteq\overline{V}(I)$，信息泄漏的测量公式如下：

$$Leak(S,\overline{v})=\sup_{s,\overline{v}}\frac{P[s\subseteq S(I)\mid\overline{v}\subseteq V]-P[s\subseteq S(I)]}{P[s\subseteq S(I)\mid\overline{v}\subseteq V]}$$

2 基于关键元组的消除信息泄漏算法的相关算法

2.1 信息公开测量算法

2.1.1 算法描述

根据关键元组的判定定理可知，如果$\mathrm{crit}^D(S)\cap\mathrm{crit}^D(\overline{V})\neq\emptyset$，则存在信息泄露，需要对信息泄漏的程度进行测量。分别求出$Ns(I)$、$Nv(I)$、$N(I)$、Ns、$v(I)$。根据$P[S(I)=s]=Ns(I)/Nv(I)$、$P[S(I)=s|V(I)=v]=Ns,v(I)/Nv(I)$进行计算，根据定义得出结果Leak值输出。

2.1.2 信息公开测量算法

输入 秘密查询 $S(I)=s$ 以及视图发布 $V(I)=v$。

输出 leak 值。

MEASURING DISCLOSURE(S,V)

(1)分别计算 $Ns(I)$、$Nv(I)$、$N(I)$、Ns、$v(I)$。

(2)求 $P[S(I)=s]=Ns(I)/Nv(I)$。

(3)$P[S(I)=s|V(I)=v]=Ns,v(I)/Nv(I)$。

(4) $Leak(s,\overline{v})=\sup_{s,\overline{v}}\frac{P[s\subseteq S(I)\mid\overline{v}\subseteq V]-P[s\subseteq S(I)]}{P[s\subseteq S(I)]}$。

(5)输出 leak 值。

2.1.3 算法分析

在计算 $Ns(I),Nv(I),N(I),Ns,v(I)$的基础上，首先求出元组集中的元组，然后求解所有的数据库示例集 I'，对数据库示例集 I'中的 n 个数据库示例进行依次判断，由于每次的查找条件具有动态变化特性，无法先对数据库示例集排序，因此采用一般的顺序查找，从 n 个数据库示例集中找出满足条件的示例集，时间复杂度为 $O(n)$。

2.2 基于关键元组的消除信息泄漏算法

2.2.1 算法描述

该算法的核心思想就是选择性地去除输入集中的元组，且最少量地去除，每一次需要判断 leak 值，当 leak 值满足要求，则结束。

2.2.2 基于关键元组的消除信息泄漏算法

输入 视图 V 的关键元组以及查询 S 的关键元组的交集U(元组个数设为 n)。

输出 去除相应元组的视图元组集以及与之相对应的、满足要求的 Leak($S,\overline{V}$)值和θ值。

ELIMINATE INFORMATION DISCLOSURE(U)

初始化：f=0/*设置标记*/；k=2 /*每次去除关键元组的个数*/

For(i=0,i<=n,i++)

{将第 i 个元组依次去除后计算 leak 值，即为 leak[i]，分别对应于 a[i]；}

采用快速排序方法对 n 个 leak 值进行从小到大的排序；

while (k<=n-1) and(f=0)

{i=1;

While (i<=n-k+1) and(f=0)

{依次选择 a[i]，a[i+1]，a[i+k-1]进行去除，计算 leak(S, V)值；

if (leak(S, $\overline{V}$)>=θ) then{i=i+1}

else{f=1}}

if{f=0} then{k=k+1}}

if {f=1} then { return(V 的关键元组集，以及相应的 leak 值);}

else {输出：该视图存在的安全隐患极大，不能发布}

2.3 算法分析

本算法的基本思想是：首先根据每个关键元组对于信息泄漏的影响程度，进行排序，然后选择性地将其去除。由于快速排序的最坏时间复杂度为$O(n^2)$、平均时间复杂度为$O(n\log n)$，而对leak值的平均时间复杂度、平均时间复杂度均为$O(n^2)$，因此“基于关键元组的消除信息泄漏算法”的最坏和平均时间复杂度均为$O(n^2)$，与任意选择关键元组算法的时间复杂度$O(2^n)$相比有了很大程度的降低。

3 实验

本文在两个数据集上测试了该算法，结果发现两种数据集的结果是相同的，时间复杂度也大大降低了。

3.1 实验设置

(1)数据集：实验使用两个数据集：

1)视图发布和秘密查询的结果都是按照常量形式给出；

2)试图发布和秘密查询的结果是按照变量形式给出。

数据集 2)需要考虑的问题比较复杂，处理的数据也较多。关键元组个数与 leak 值的数据变化见表 1。公共元组集中元组个数随 leak 值的变化见图 1。

(2)机器配置：CPU AMD1800+；内存 256MB SDRAM；操作系统 Windows2000；C 语言编程环境。

表 1 关键元组个数与 leak 值的数据变化

N_{tup}	$N(S)$	$N(V)$	$N_{s,v}$	$P(S)$	$P(S\mid V)$	$Leak$
2	1	3	1	0.25	0.33	0.25
4	3	3	1	0.187 5	0.33	0.312 5
6	3	7	1	0.046 9	0.142 9	0.611 8
12	15	63	3	0.003 7	0.047 6	0.922 3
18	127	127	3	$0.038\ 2*10^{-4}$	0.011 7	0.939
27	255	255	3	$0.037\ 8*10^{-4}$	0.007 8	0.999 5

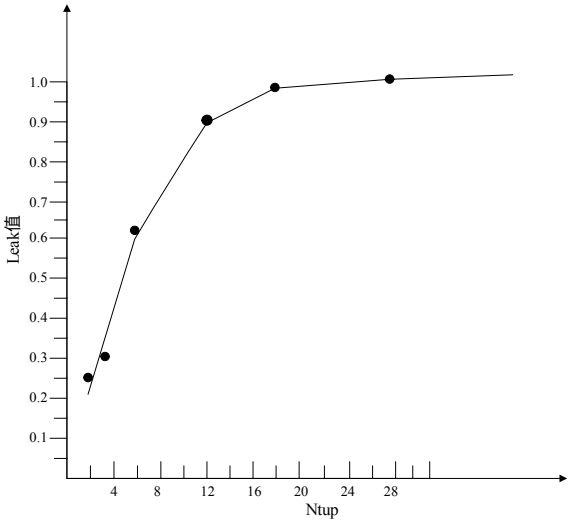

图 1 关键元组个数随 leak 值变化曲线

3.2 结论

随着关键元组数目的不断增加，leak 值逐渐变大，去除相应的关键元组必定会降低 leak 值，根据给定的值去除关键元组个数及其满足条件的结果。实验证明，在不影响视图发布的情况下本方法可以有效地降低信息泄漏程度。

(下转第 91 页)