

安全数据库隐蔽通道的标识技术与实例分析

王保华^{1 3}, 马新强¹, 李丹宁², 李 丹², 章 衡¹, 赵振勇¹

(1. 贵州大学 信息工程学院, 贵州 贵阳 550003 ;

2. 贵州科学院, 贵州 贵阳 550001 ; 3. 淮北煤炭师范学院, 安徽 淮北 235000)

摘 要 LogicSQL 数据库是自主研制基于 Linux 的高安全级别安全数据库。重点对安全数据库的隐蔽通道标识技术进行研究, 为寻求更好的隐蔽通道标识方法, 对共享资源矩阵、信息流公式法、无干扰法等标识方法从理论上进行分析, 以 LogicSQL 安全数据库的隐蔽通道进行实例分析。改进的共享资源矩阵是目前相对比较成功的隐蔽通道标识方法, 无干扰法的实际应用可作以后的隐蔽通道标识方法研究重点。

关键词 隐蔽通道分析; LogicSQL; 安全数据库

中图分类号: TP309.2

文献标识码: A

文章编号: 1673-629X(2007)02-0233-03

Logo Technology and Analysis Examples of Security Database Covert Channel

WANG Bao-hua^{1 3}, MA Xin-qiang¹, LI Dan-ning², LI Dan², ZHANG Heng¹, ZHAO Zhen-yong¹

(1. School of Information Project, Guizhou University, Guiyang 550003, China ;

2. Guizhou Academy of Science, Guiyang 550001, China ; 3. Huaibei Coal Normal College, Huaibei 235000, China)

Abstract LogicSQL database was the independent development based on the Linux high security rank security database. Focus on covert channel logo technology research of security database, in the search for better methods of covert channel logo, for sharing resources matrix, information flow formula method, without interference method and logo technology from theoretical analysis, and covert channel examples analysis of LogicSQL database security. Improved sharing of resources matrix is relatively successful covert channel logo methods. And practical application of the method without interference can become future research priorities of covert channel logo methods.

Key words covert channel analysis; LogicSQL; security database

0 引 言

信息安全已经成为当前研究的热点课题, 作为信息系统核心的数据库的安全尤其成为信息安全的重中之重^[1]。国内大部分企事业单位, 包括国家的一些关键部门大多数都使用国外进口的数据库产品, 如 ORACLE, DB2, SYSDBASE 等。但是国外限制了 B1 以上级别的安全数据库对中国的出口, 在这种情况下, 加强国产安全数据库的研制就显得非常重要。国产 LogicSQL^[2]安全数据库的研发就在这个背景下得到各级政府的大力支持, 取得了快速的发展。主要对 B2 级别安全数据库的隐蔽通道标识技术研究, 并在 LogicSQL 安

全数据库中进行隐蔽通道实例分析。

1 隐蔽通道概念

隐蔽通道的概念最初是由 Lampson 于 1973 年提出, 他在论文“关于限制问题的注释”中这样定义隐蔽通道: 如果一个通道不是设计用于传递信息, 则称该通道为隐蔽通道^[3]。这个定义非常模糊, 很难用于实际的分析。但该定义可使人们获得对隐蔽通道的直观的认识。B2 安全数据库要求系统开发者要彻底搜寻隐蔽存储通道^[4]。下面给出 Tsai 在 1990 年提出的隐蔽通道定义, 它是目前被广泛使用的隐蔽通道的定义。定义为: 一个非自主的安全模型 M , 以及在一个系统中该模型的解释 $\llbracket M \rrbracket$, 在任何两个主体 $\llbracket S_i \rrbracket$ 和 $\llbracket S_j \rrbracket$ 之间的信息是隐蔽通道, 当且仅当在模型 M 中, 对应主体 S_i 和 S_j 之间的任何通信都是非法^[5]。该定义将访问控制策略的模型和解释(实现)分开, 说明了隐蔽通道根源于模型和解释之间的一致性, 并且清楚地

收稿日期: 2006-04-30

基金项目: 贵州省优秀科技教育人才省长专项资金资助(黔省专合字(2005)88号)

作者简介: 王保华(1976-), 男, 安徽巢湖人, 硕士研究生, 研究方向为信息安全、数据库、人工智能; 李丹宁, 副研究员, 硕士生导师, 研究方向为数字地球与人工智能。

阐明了隐蔽通道与各种访问策略的关系。这个定义表明：

(1) 隐蔽通道与自主安全策略模型无关。

由于自主安全策略模型本身的缺陷导致的问题是,无法在实现中解决的,所以讨论基于自主策略模型的系统的隐蔽通道是没有意义的。

(2) 隐蔽通道依赖于具体的非自主策略模型。

一个信道是不是隐蔽通道,取决于策略模型是否禁止两个主体通过这个信道通信。如果一个模型中禁止这样的通信,它就是隐蔽通道,反之就不是。

(3) 隐蔽通道与保密性也与完整性模型有关。

对实现非自主保密性模型的系统,隐蔽通道是从高安全级别向低安全级别的非法的信息通道。对于实施非自主完整性模型的系统,隐蔽通道是从低完整性级别向高完整性级别的非法信息通道。

(4) 隐蔽通道依赖于 TCB 规约。

系统的 TCB 规约包括主体操作原语、客体操作原语、访问权限和安全级别的操作原语,还包括访问授权、主体客体的创建和删除规则等等。

2 隐蔽通道的识别技术研究

隐蔽通道的识别是隐蔽通道分析中最为困难的一个环节,隐蔽通道的识别方法有以下几种。

2.1 共享资源矩阵法(以下简称 SRM)

基本 SRM 见表 1。

表 1 共享资源矩阵法的基本 SRM 表

资源属性变量	操作(G)
S ₁	R
S ₂	R
T ₁	M
G	R
U	RM

该方法的分析步骤：

(1) 分析所有的 TCB 原语操作,确定通过 TCB 接口用户可见或可修改的共享资源属性。

(2) 构造共享资源矩阵,该矩阵的各行对应于用户可见的 TCB 原语、各列对应于用户可见/可修改的共享资源属性。如果一个原语可以读一个变量,则将该矩阵项 TCB 原语、变量标记为 R,类似地,如果一个原语可以修改一个变量,则将该矩阵项 TCB 原语、变量标记为 M,最后,将既不能读又不能写的变量合并,分析时将它们视为一个变量。

(3) 资源矩阵完成传递闭包操作,具体步骤如下：在矩阵中搜索包含标记 R 的每一项,如果该项所在的

行中出现 M 标记,则检查包含该 M 项的所在列。如果在该列的任意一个行中出现 R 标记,且该行与原始 R 项所在列的对应行中没有 R 标记,则在该矩阵项中增加间接读标记 R,重复以上操作,直到矩阵中无法再增加 R 项时为止。

(4) 分析每个矩阵行,找出同时包含 R 和 M 的行,并删去其他矩阵行,当一个进程可以读一个变量且另一个进程可以写该变量时,如果写进程的安全级支配读进程的安全级,就可能产生潜在的隐蔽通道。

(5) 分析矩阵所有的项,构造潜在隐蔽通道的实际应用场景。可以构造出实际应用场景的潜在隐蔽通道,即为真实隐蔽通道。

此方法的缺点：

- a. 仅适用高度结构化的数据库系统。
- b. 系统存在安全缺陷时,这种方法所标识的大量潜在隐蔽通道将不会是真实隐蔽通道。

改进的共享资源矩阵法首先对共享资源矩阵进行下面的细化(见表 2)：

(1) 分用户与系统之间的信息流和状态属性之间的信息流,将用户的输入与系统的输入分别归并成一行,分别标记为 u-in 和 u-out。

(2) 信息流的流入属性。

(3) 分信息流的产生条件。如表 2 所示：G1 = G(条件成立时调用),G2 = NOT G(条件不成立时调用),G3 = TRUE(无条件时调用)。

通过对改进的矩阵项的分析,可以得到以下 4 种不同类型的通道：

- a. 如果有一个合法的操作与该操作序列的源和结果相同,它是合法通道,不是隐蔽通道。
- b. 该通道无法获得有用的信息,只能传送操作已知的信息,它不是隐蔽通道。
- c. 发送进程与接收进程是同一个进程,它也不是隐蔽通道。
- d. 除了前面三种情况,其他的通道是隐蔽通道。

表 2 改进的共享资源矩阵的细化 SRM 表

资源属性变量	操作(G1)	操作(G2)	操作(G3)
S ₁	R		
S ₂		R	
T ₁	M	M	R
G	R	R	
u-in(输入)	R	R	R
u-out(输出)			M

2.2 信息流公式

信息流公式也称安全验证条件(SVC)。该方法一

般用于对形式化的系统描述进行自动化的信息流分析。从下面的例子来了解如何通过信息流公式来进行隐蔽通道分析。

```
Var T1 S2 G
Procedure OR( Var UR )
Begin
  If G then T1 := S1 ;
  else T1 := S2 ;
  UR = T1 + UR ;
End
```

这里先把这段程序按照依赖关系三元组

```
{T1 ,{S1 ,G} ,G}
{T1 ,{S2 ,G} ,not G}
{UR ,{T1 ,UR} ,true}
```

假设存在一个级别映射函数 $Leve()$,它给出每一资源属性的安全级别。生成的信息流合法性公式如下：

```
G→Leve( T1 )≥Leve( S1 )
¬G→Leve( T1 )≥Leve( S2 )
Leve( T1 )≥Leve( G )
Leve( UR )≥Leve( T1 )
```

用一阶谓公式来验证系统的安全性。如果这些公式都成立 ,则说明没有隐蔽通道的的存在 ;否则 ,还要进一步人工分析 ,看是否确实存在隐蔽通道。

这种方法的缺点：

- (1)可能有大量的公式无法证明或证明为假 ,但是其中往往只有一小部分是隐蔽通道。
- (2)增加语义分析时 ,不同的语言需要开发不同的编译器和自动化信息流分析工具。

2.3 无干扰方法

这是一种受到重视的方法。它的思想是如果一个用户看到的系统运行情况与另一用户的操作行为无关 ,那么在他们之间就不会有隐蔽通道。

下面是无干扰的形式化定义 :给定一个状态机 TCB ,令 X 和 Y 为两个用户进程 , i 为一个输入序列 ,它的结尾是 Y 的输入。令 i/X 表示从 i 中删除所有 X 的输入后剩下的子序列。假设在初始状态输入 i 后 , Y 得到的输出为 $Y(i)$,称进程 X 与进程 Y 无干扰 ,如果对于所有可能的以 Y 的输入为结尾的输入序列 i ,都有 $Y(i) = Y(i/X)$ 。

根据这个定义来证明没有隐蔽通道 ,需要对所有的可能的输入序进行证明 ,这样做的工作量很大。在应用无干扰方法时 ,为了避免分析无穷多个输入序列 ,应当将 TCB 的状态分成不同的等价类(等价定理)。称两个状态是 Y - 等价的 ,如果 :a. 对同一个 Y 输入具有相同的 Y 输出 ;b. 对于任何输入 ,相应的下一个状态也是 Y - 等价的。

进程 X 与进程 Y 无干扰 ,当且仅当对于 X 的任何输入 ,都使当前状态迁移到一个 Y - 等价状态(展开定理)。因此 ,展开定理使人们能够分析单独的 TCB 函数和原语。只要给出说明 TCB 状态和状态迁移的形式化规范 ,就可以应用展开定理进行无干扰分析。

该方法的缺点是：

- (1)没有支持的自动工具 ,单纯依靠手工分析不仅工作量大 ,而且增加人为因素 ,容易出错；
- (2)该方法是一种“ 乐观 ”方法 ,只能用于形式化的系统 ,在实际中没有成功的例子。

3 LogicSQL 隐蔽通道实例分析

假设有用户 A 和用户 B ,其中用户 A 是机密级用户(S 表示机密级) ,用户 B 是非机密级用户(U 表示非机密级)。用户 A 和用户 B 都能对表 3 具有修改表的模式的权限。

表 3 多级关系 U

职工姓名	职务	工资
张三(U)	科员(U)	1200.00(U)

现在假如用户 B 对上面的表插入机密级数据后 ,用户 B 访问看到的是整个多级关系表 ,如表 4 所示。但用户 A 能访问看到的表仍是表 3。

表 4 多级关系表 S

职工姓名	职务	工资
张三(U)	科员(U)	1200.00(U)
李四(S)	处长(S)	1800.00(S)

当用户 A 尝试使用命令来修改表的模式 ,这条命令就不能执行。这样就产生一隐蔽通道 ,机密级用户插入数据后 ,非机密级用户就可以通过尝试能否修改表的模式来判断机密级用户是否插入数据。处理这个隐蔽通道的措施是 :改变 TCB ,规定这个非机密级用户任何时候都可以修改表的模式 ,这样就可以避免这个隐蔽通道。

4 结 论

隐蔽通道分析是一个具有挑战性的课题 ,在隐蔽通道分析中 ,隐蔽通道标识是这个课题中最为困难的问题 ,也是一个不断发展中的课题。困难的程度 ,依赖于具体的系统和所采用的分析方法。一般地说 ,系统规模越大 ,系统越复杂 ,分析的难度就越高。改进的共享资源矩阵法是相对比较成功的一种隐蔽通道标识方法^[6]。无干扰方法 ,是一种理论上严谨的形式化方法 ,但离实际应用还是有一段距离 ,是以后的研究重点。

if v is odd then $\{v \leftarrow v - 2 ; R \leftarrow R - 2Q ;\}$

else $\{v \leftarrow v - 1 ; R \leftarrow R - Q ;\}$

return $R ;$

其中 $u[i-1 \rightarrow i-w+1] || 1$ 表示取 u 的第 $i-1$ 至 $i-w+1$ 位组成位串与 1 组成新的位串 $[2^i]R$ 表示对 R 进行 i 次倍点运算。算法 2 首先置标量 u, v 为奇数, 而对奇数标量 u, v 按照类似 NAF 算法转换成 $0 \dots 0x \dots 0 \dots 0x$ 的形式, 其中 x 为奇数, 且 $x \in \{\pm 1, \pm 3 \dots, \pm 2^{w-1} - 1\}$, 具体方法: 如果 $u[i] = 0$ 则 $t_1 \leftarrow t_1 - 2^w$, 如果 $u[i] = 1$ 则 $t = u[i-1 \rightarrow i-w+1] || 1$, 其中 $a || b$ 表示将 a 和 b 连接, 这样保证转换后的结果成为 $0 \dots 0x \dots 0 \dots 0x$ 的形式^[5]。

2.2 性能分析

(1) 内存空间需求。

由于 x 为奇数, 且 $x \in \{\pm 1, \pm 3 \dots, \pm 2^{w-1} - 1\}$, 则需要预计算 $3P, 5P, \dots, (2^{w-1} - 1)P, 3Q, 5Q, \dots, (2^{w-1} - 1)Q$ 。因此需要 2 次倍点运算和 $2^{w-1} - 2$ 次点加运算, 需要预存储 2^{w-1} 个点。

(2) 非零密度。

标量在进行点乘运算时转换成了 $0 \dots 0x \dots, 0 \dots 0x$ 形式, 其中 $0 \dots 0x$ 为连续 $w-1$ 个 0 和 1 个奇数 x , 故非零密度为 $1/w$, 因此新的算法主计算阶段需要进行 $n+1$ 次倍点运算和 $2n/w$ 次点加运算。

算法 1 由于在主计算之前必须分别将标量 u, v 转换成其相应的 w NAF 表示形式, 而 w NAF 表示形式必须从右到左进行计算(即从标量的最底位向最高位进行), 故需要首先计算并存储标量的 NAF 表示形式。而由于新的算法对标量编码是从最左到右进行, 因此新算法的编码阶段和主计算阶段合并在一起, 不需要存储标量 u 和 v 的新的编码, 这样可以节省存储标量 u, v 的 NAF 表示形式的编码, 故可以节省内存空间, 这对于内存空间受限的设备来说尤其有益。

(3) 安全性分析。

攻击者虽然可以通过测试电量的消耗来区分点加和倍点运算, 但由于新提出的算法通过采用固定模式 $0 \dots 0x \dots 0 \dots 0x$ 对标量进行处理, 始终得到相同的序

列 $D \dots DA \mid D \dots DA \mid \dots \mid D \dots DA$, 其中 D 表示倍点运算, A 表示点加运算, 因此通过 SPA 攻击得不到秘密 u 和 v 。故新的算法是抗 SPA 攻击的。

3 结束语

文中, 在内存空间和计算时间负担增加不多的情况下, 基于 interleaving 多点乘算法, 提出了一个新的抗 SPA 的多点乘算法。虽然本算法是抗 SPA 的多点乘算法, 是针对多点乘运算的, 但通过同构等方法, 本算法也同样适用于安全的点乘运算。

参考文献:

- [1] Kocher P, Jaffe J, Jun B. Introduction to Differential Power Analysis and Related Attacks[EB/OL]. 1998. URL: <http://www.cryptography.com/dpa/technical/index.html>.
- [2] Kocher P, Jaffe J, Jun B. Differential Power Analysis[C]//In Proceedings of CRYPTO '99, LNCS vol 1666. [s. l.]: Springer-Verlag, 1999: 388-397.
- [3] Coron J S. Resistance Against Differential Power Analysis for Elliptic Curve Cryptosystems[C]//In Proceedings of CHES '99, LNCS vol 1717. [s. l.]: Springer-Verlag, 1999: 292-302.
- [4] Montgomery P L. Speeding the Pollard and Elliptic Curve Methods for Factorization[J]. Mathematics of Computation, 1987, 48: 243-264.
- [5] Okeya K, Takagi T, Vuillaume C. On the Exact Flexibility of the Flexible Countermeasure against Side Channel Attacks[C]//In The 9th australasian conference on information security and privacy, ACISP 2004, LNCS vol 3108. [s. l.]: Springer-Verlag, 2004: 466-477.
- [6] Okeya K, Takagi T. The Width- w NAF Method Provides Small Memory and Fast Elliptic Scalar Multiplications Secure against Side Channel Attacks[J]. IEICE Transactions, 2004, E87-A: 75-84.
- [7] Lee Mun-Kyu. SPA-Resistant Simultaneous Scalar Multiplication[C]//In Approaches or Methods of Security Engineering Workshop, LNCS vol 3481. [s. l.]: Springer-Verlag, 2005: 314-321.

(上接第 235 页)

参考文献:

- [1] 张剡, 夏辉, 柏文阳. 数据库安全模型的研究[J]. 计算机科学, 2004, 31(10): 101-103.
- [2] Li-Yan Yuan. The Documentation of LogicSQL[M]. Canada: Alberta University, 2005.
- [3] National Computer Security Center. A guide to understanding covert channel analysis of trusted systems[R]. NCSC-TG-030. [s. l.]: [s. n.], 1993.
- [4] 张敏, 徐震, 冯登国. 数据库安全[M]. 北京: 科学出版社, 2005.
- [5] 卿斯汉, 刘文清, 温红子. 操作系统安全[M]. 北京: 清华大学出版社, 2004.
- [6] 卿斯汉. 高安全等级安全操作系统的隐蔽通道分析[J]. 软件学报, 2004, 15(12): 1837-1849.