1996 年 7 月 重庆大学学报 (自然科学版) Vol. 19, №. 4
第 19 卷第 4 期 Journal of Chongqing University (Natural Science Edition) Jul. 1996

(8 37-43

# 一个关于企业管理信息系统的安全策略和安全技术*

# The Security Strategy and Security Controls and Techniques for the Enterprise Management Information System

曾 一
Zeng Yi
(重庆大学计算机系,重庆,630044;第一作者 34 岁,男,讲师)

F270.7
TP309

A
**摘 要** 介绍一个基于 UNIX/ORACLE 的企业管理信息系统所采用的安全策略、安全控制与技术及其在系统中的实现方法。其安全控制是通过分层分级来实现的,特别是面向最终用户的应用层,应用软件有其自身的安全控制机制。特别强调安全教育与管理是安全系统的一个重要方面。

**关键词** 安全措施;安全技术;安全教育;管理信息系统;数据库管理系统;操作系统
**中国图书资料分类法分类号** TP309

**ABSTRACT** This paper attempts to present the security strategy and the security control and its implementation via the hierarchy of securities used in an enterprise management information system based on unix and oracle. It describes the useful and critical techniques applied to every level of the hierarchy of securities, espcially the security control mechanism built in the applications. It also stresses and explains the importance of the safety education and management.

**KEYWORDS** safety measures; safety technics; safety education; management information systems; database management systems; operating systems

## 0 引 言

计算机科学与技术的不断发展和计算机的广泛应用,促进了社会的进步和繁荣,为人类创造了巨大的财富,但同时也对其自身构成了新的威胁。诸如计算机本身的不可靠性、环境干扰、自然灾害、工作失误、操作不当、未授权窃取、未授权修改、蓄意破坏、以及计算机病毒的侵害等涉及到计算机系统硬件、网络通讯、系统软件、应用软件、数据人员及相应的组织机构等各个方面,潜伏着严重的不安全性、脆弱性和危险性。软件作为计算机信息处理系统的核心,是使用计算机的工具,也是计算机安全控制中关键的技术措施,同时也成为危害计算机安全的环节和手段[1-3]。

* 收文日期 1995-09-28

事实上,企业管理信息系统的实际应用几乎都是在操作系统的支持下,基于数据库管理系统,并通过应用软件来实现的。整个应用系统的安全性,不仅依赖于操作系统、DBMS 等,而且依赖于目前计算机安全中最重要又是最薄弱的应用软件的安全性。因此,将安全管理和防范策略与技术应用于信息管理系统的各开发阶段和过程中,可以加强系统抵御意外的或蓄意的未授权存取的能力,防止数据的未授权修改和传播,从而提高企业信息安全的机密性、完整性和有效性。笔者在本文中将介绍一个实际运行的企业信息管理系统所采用的安全策略、安全控制与技术以及安全管理措施等。

## 1 安全策略与安全层次

### 1.1 安全策略

建立一个安全的计算机系统,需要在一系列需求之间求得平衡。安全策略必须尽可能避免各种因素发生矛盾以防影响系统的其它特性,当安全性与其它特性发生抵触时,应当根据它对系统的重要性作出取舍,否则可能会建立无实际意义的超高水平的保护体系而浪费财力,或可能经常发生错误的报警信号,干扰系统正常工作而实际上却又缺乏应有的安全防护能力,或可能影响效率,甚至拒绝服务。

笔者在建造系统的过程中所遵循的原则是:

1) 把安全性作为一种需求,在系统开发的一开始就加以考虑,使得安全要求从开始就作为系统目标的一部分,使其在系统开发过程中起主导作用。实际经验证明,安全保护的事后追加措施是很难达到安全性能要求的;

2) 在系统的不同层面上,利用不同的安全控制机制,实施不同精度的安全控制。同时尽可能减少各层次之间的安全相关性[1],以便于确定安全控制的可靠性和可行性。特别是应用软件的安全设计有其特殊性,除了一般安全性所包括的保密性、完整性和可用性外,应特别注意直接面向用户的人机交互界面的设计;

3) 系统而合理地使用安全技术和控制安全粒度,使系统在安全性与其它特性之间求得平衡。这些特性包括能力、效率、灵活性、用户友好界面和成本等;

4) 在安全性和其它特性之间不能取得一致,或在某些安全保护不能或很难通过技术手段来实际的情况下,通过制定各级人员安全操作规程和明确安全职责,并以行政管理的手段付诸实施来弥补技术方面的不足;

5) 在应用层除用户口令字之外,不采用任何加密方法[4~5]对数据加密,以提高系统运行效率。

### 1.2 安全层次

这里所介绍的系统实际上是一个 Client/Server 体系结构下的 MRP Ⅱ 系统,从应用角度看提供了覆盖企业人、财、物、产、供、销各个生产活动的信息管理功能。

该系统的服务器环境由 SUNSPARC Server 1000、SUNSPRARC Clasic 等工作点、操作系统 Solaris 2.2、数据库管理系统为 ORACLE7 所组成,客户端工作站环境则由 PC486(386)微机,操作系统为 MSDOS6.0/UCDOS3.2 及 Oracletools 和 FTP2.0 所集成,服务器与客户工作站之间以粗缆(IEEE802.3 10 BASE 5)为主干网,通过收发器、HUB 集线器和双绞线(IEEE-802.3 10 BASE T)将其连接起来,成为一个基于 TCP/IP 网络协议,能实现分布处理、资源共

享、数据共享、支持多用户、多进程并发操作和访问等功能的分布式系统网络环境。不难看出，整个系统大致分为如图所示 4 个层次：

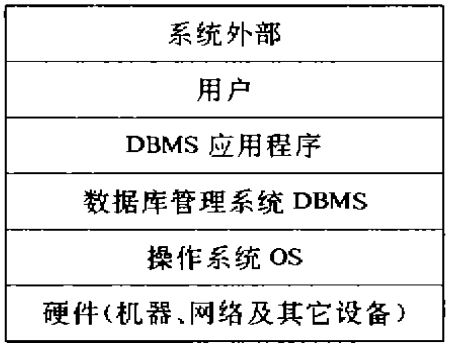


图 1　系统安全层次

关于硬件的安全在诸多著作中均有介绍[1~3]，这里不再论及，但我们假设硬件是安全的，通常，操作系统被认为是安全的，因此，硬件与操作系统是处于安全周界内的[1]，这样，整个系统是安全层面可以分为如下 4 个层面：

第一层，Oracle RDBMS→OS，安全性由操作系统控制，详见 2.1；

第二层，RDBMS 应用程序→RDBMS，安全性由 RDBMS 控制，详见 2.2；

第三层，最终用户→RDBMS 应用程序，安全性由应用程序控制，详见 2.3；

第四层，系统外部→应用系统的运行维护(最终用户)，安全性由安全教育与管理、结合行政手段来保证。它包括应用系统自身不能解决的或很难解决的保护系统安全的所有活动。例如，对 Client 端 PC 工作站的安全管理。

## 2　安全技术及其实现

安全策略最终都要转化为对数据的存取控制。这些存取控制分为两个方面：一方面是安全技术，另一方面是安全管理。

### 2.1　操作系统的用户控制和文件保护

Solaris2.2 实际上是由基于 Unix SVR4 的 SUNOS5.2、NFS 和 Openwindows V3.2 等组成。ORACLE7(RDBMS)作为操作系统的一个用户，其安全性完全受控于操作系统。一方面，ORACLE7 以一个用户的身份可改变自己的口令和自己拥有的一切文件的安全属性，达到对自己的保护目的。另一方面，操作系统的超级用户具有至高无上的权限，因此任何用户及其拥有的文件均可通过超级用户改变用户口令、改变文件属性来达到控制用户和保护文件的目的。实现这些安全保护，可通过下面的命令和方法：

- 定期改变用户 ORACLE7 的口令，可使用 Passwd 命令或修改/etc/passwd 文件；
- 使用 Chown，chgrp 和 chmod 等命令来改变文件的属主、用户组和存取方式；
- 将后备磁盘作为一个文件系统用以存放用 ORACLE 工具 EXPORT 卸出的应用系统的基础数据，备份之后通过命令 umount 实现对文件系统的保护；
- 通过修改 rhosts 文件，来限制服务器之间的远程拷贝(rcp)、

### 2.2　DBMS 的用户控制和数据保护

由于应用软件是建立在 ORACLE7 基础之上，因此，作为应用软件的用户必然是 ORACLE7 的用户，为了保证 ORACLE7 的用户与应用软件的用户一致性，我们作了如下约定：

① 当建立应用软件的用户时，先将其作为 ORACLE7 的用户建立；

② 当删除 ORACLE7 的一个用户时，先将其从应用软件的用户集合中消除。

这里，ORACLE7 的用户是借助于 ORACLE7(RDBMS)基于角色的安全特性而建立的。

建立一个用户的过程是：

1) 首先是 system 用户创建专门用于管理应用系统数据库的 DBA，口令同时生成(以后可随时变换)，此步仅在系统初创时执行一次；

2) 由 DBA 创建组成应用数据库的各表及视图(Table 和 View)；

3) 由 DBA 创建 ORACLE7 的用户(应用数据库的用户)，并给其分配口令(以后可随时更换)；

4) 根据用户的工作职责和工作范围，授予其对表或字段的存取权限(查询、插入、修改、删除等)。

这样，不是由应用数据库 DBA 创建的用户无权访问应用数据库，而由此 DBA 创建的用户其存取权限受到限制。

除此之外，我们还可使用下面的安全技术来加强对数据的安全保护。

1) 数据分布、用户分布和处理分布

利用 ORALE7 支付数据分布的特点，可按应用软件的功能、数据耦合程度、用户访问数据的频度及服务器的自治能力，将数据和用户分布于 SUN 服务器之间。实际上，用户的分布也导致了服务器对客户的服务(处理)的分布)。尽管通过建立数据库链路(DATABASE LINK)和同义词(SYNONYM)来支持用户的透明存取，数据还是在相当程度上被隔离开来[1][3]。

2) 使用审计功能 AUDIT

ORACLE 的安全审计功能为我们提供了一种监控数据存取的途径，但对于大型应用系统来说，审计功能的启动将影响系统的效率；

3) 设置 DATABASE TRIGGER

我们可以将应用程序中较为复杂的数据合法性和有效性检验改为用数据库触发器(DATABASE TRIGGER)来实现。

### 2.3 应用软件的用户控制和信息保护

上述 2.2 对数据进行保护其安全粒度可控制在记录级或字段级，但这样的控制至少有两点值得考虑：

① 由于控制粒度过细将会给应用系统带来效率上的损失和程序灵活性方面的损失；

② 对于较大型的应用系统，负责应用数据库的 DBA 的工作将变得相当繁杂。

因此，结合企业信息管理的特点，除了某些重要的数据，一般不将安全粒度控制太细，以求得应用程序在更为宏观方面的数据安全控制。这里我们采用存取监督器的概念[1]，通过控制用户对程序的访问实现数据在信息级上的保护。如图 2 是一个存取监督器模型。

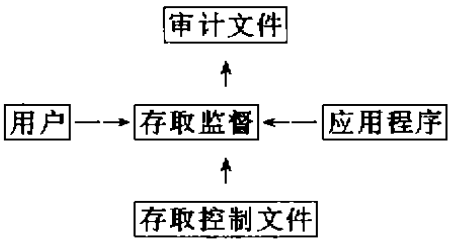

图 2 存取监督器

将这样的存取监督器嵌入每一个应用程序，就可达到信息安全保护的目的。嵌入存取监督器的程序结构变为如图 3 所示的形式。

程序中存取监督器的实现是基于存取控制文件(MUSERS 和 MPQOGS)和审计文件(MUSERSAUDIT 和 MPROGSAUDIT)以及相应的程序控制流程三方面的设计：

· 存取控制文件

MUSERS 用于存放应用软件的合法用户及有关控制信息，包括用户标识(key)、中文标识(中文名字)、保密字、访问级别、功能标识、用户有效到期日等；

MPROGS 用于存放应用程序及有关控制信息，包括程序标识(key)、程序说明、访问级别、程序可用标志、程序有效到期日等。

· 审计文件

MUSERSAUDIT 用于存放对应用软件用户安全审计的信息，包括用户标识(key)、累计使用系统时间、累计登录使用次数、上次使用系统的日期，最后一次修改日期，最后一次使用的程序标识等；

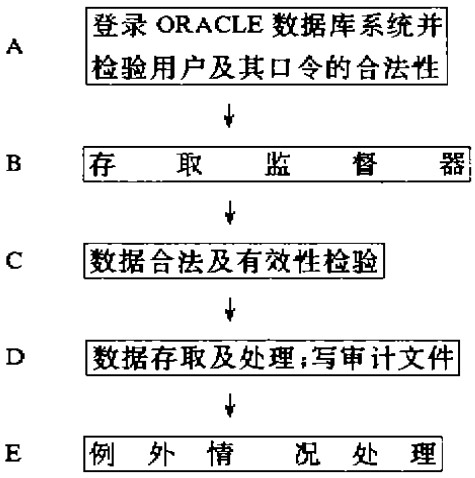


图 3　嵌入存取监督器的应用程序框架

MPROGSAUDIT 用于存放对程序安全审计的信息，包括程序标识(key)，累计使用时间、累计执行次数、上一次执行日期、最后一次使用的用户标识、最后一次修改日期等。

· 程序中存取监督器控制流程

这里，我们只给出图 3 中第 2 框存取监督器的控制流程，如下图 4 所示。

对于 MUSERS、MPROGS、MUSERSAUDIT 和 MPROGSAUDIT 表均由前述的应用数据库管理员 DBA 建立，并保证其安全性。对于如何设置和维护好 MUSERS 及 MPROGS 中的数据是运用存取监督器的关键，下面对此进行必要的说明：

1) 维护 MUSERS 和 MPROGS 的程序应该只能被少数几个具有应用系统最高权限的应用系统管理员执行；

2) 对 MUSERS 中的保密字进行加密后存入。密码选用 Caesar 密码[1,5]，加解密算法也因此较为简单，不会影响程序执行速度；

3) 用户的访问级别主要是根据用户对访问数据所负有的责任大小而定，同时考虑用户所属部门和工作范围；如

生产部计划员＞采购部计划员；

应用系统管理员＞部门数据维护人员＞部门数据查询人员，等。

4) 程序的访问级别是根据程序所完成的功能、存取数据的多寡和种类来确定的；如

库存结帐程序＞库存入/出库程序＞库存查询程序；

5) 用户的功能标识与程序的功能标识有相同的位数，位数的多少是依据系统所划分的子系统的个数来确定的，每一位代表一个子系统。若某用户有权使用某子系统，则 MUSERS 中的功能标识字段中相应位设置为 1，否则为 0；若某程序被组合到某子系统中，则 MPROGS 中的功能标识字段相应位设置为 1，否则为 0.

由此可见，存取监督器的安全控制是极其灵活的，可根据实际应用要求，通过调整访问级别和功能标识来达到新的安全目标。

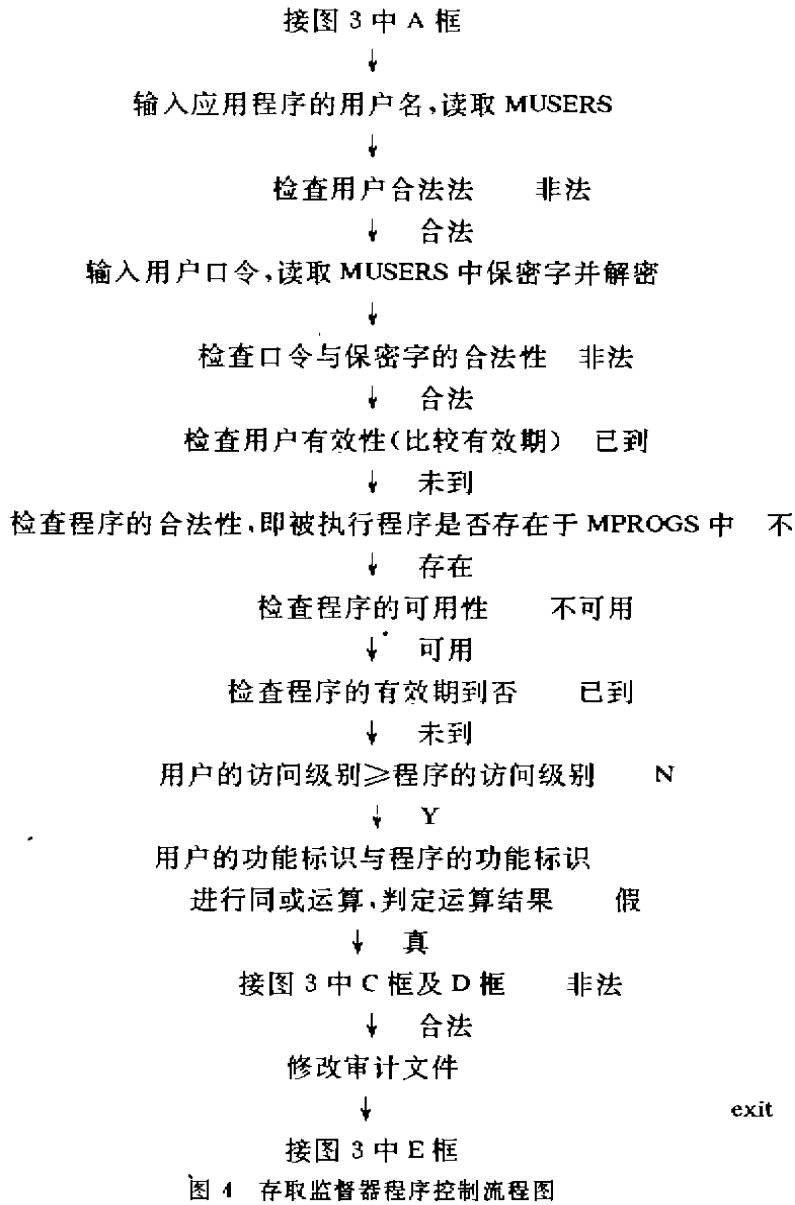


图 4 存取监督器程序控制流程图

下面举一个例子说明它的用法。假设 MUSERS 和 MPROGS 分别有两条记录,部分内容分别为:

MUSERS:

| 用户 | 访问级别 | 功能标识 | 注释 |
|---|---|---|---|
| U1 | 3 | 11111 | 计划员 |
| U2 | 5 | 00010 | 库管员 |

MPROGS:

| 用户 | 访问级别 | 功能标识 | 注释 |
|---|---|---|---|
| PA | 1 | 11111 | 库存查询 |
| PB | 4 | 00010 | 入库处理 |

从上面的设置可知,用户 U1 可执行程序 PA 而不能执行 PB,尽管他拥有使用 5 个子系

统的权限；U2 即可执行程序 PA 又可执行 PB，因为 PA、PB 都属库存管理子系统，但他无权使用其它任何一个子系统。

### 2.4 存取控制矩阵

| 仓管员标识 | 可操作的仓库员标识 |
|---|---|
| ⋮ | ⋮ |

对于某些数据的存取控制，前述技术是无法实现的。如外协仓库、外购仓库和自制仓库中可能都存放曲轴这种零件，而入出库历史记录文件（表）又是三个仓库共享，那么如何控制某仓库的入出库记录不会记录到其它仓库的帐上呢？我们的办法是使用存取控制矩阵。在 ORACLE 上的实现则是建立一个二维表：

当仓管员利用入/出库程序对仓库进行入出库处理时，应先查表检查仓管员对仓库（号）的操作的合法性，当合法时，才能进行入出库处理。这样的合法性检查应加入相关程序的如图 3 所示 B 框与 C 框之间。

## 3 安全管理

如何防止计算机病毒的感染和传播是安全管理的一个很好的例子。实践表明，开发一个安全的企业信息管理系统，安全教育与管理是非常重要的一个方面。从机房安全、设备安全、网络安全等实体安全到开发系统、维护运行系统等过程安全，都需要对各层人员进行安全教育，以提高安全管理水平。从系统开发伊始，就应分析安全需求，建立符合企业实际管理的安全目标，针对不同工作环境、不同工作人员，制定安全职责和安全操作规程。实际上，安全管理可以解决应用软件无法解决的某些问题。

没有 100%的安全系统，任何时候，安全性是相对而言的。无论怎样，制定安全目标和安全策略对于建立一个安全的计算机系统是举足轻重的。可选择不同的安全粒度，如记录级、文件级、信息级等，在系统的各个层次展开安全控制是非常有利的。在应用软件层面上设置安全控制是加强整个应用系统安全性的重要步骤。安全教育与管理不容忽视，应该把它作为项目管理的一部分。

虽然不能严格地用安全评价标准如“桔皮书”，对本系统进行安全认证，但由于本系统是建立在 UNIX（$C_2$ 安全级）和 ORACLE（$C_2$ 安全级）之上，同时，从一年来的系统运行情况来看，我们所采用的安全策略、安全控制与技术和管理方法是可行的，且是有效的，运行速度和安全效果令人满意。

## 参 考 文 献


1 莫瑞，加瑟著，吴亚菲译．计算机安全的技术和方法．北京：电子工业出版社，1992

2 苏诚信等著，计算机安全指南．北京：清华大学出版社，1993

3 V. P. Lane. Security of computer Based on Information System. MACMILAN, 1985

4 游之墨等著．计算机安全保密入门．北京：人民邮电出版社，1988

5 [日]一松信主编．数据保护和加密研究．北京：科学出版社，1991