涉密应用系统三员分离设计与研发

黄梁标，郭正华
（南昌航空大学航空工程制造学院，南昌 330000）

摘要：为解决涉密应用系统中管理员权限过于集中问题，提高系统自身的安全性。通过以 Oracle 为数据库平台，采用 VC++程序设计语言，设计与实现涉密应用系统三员（即系统管理员、安全保密管理员、安全审计员）分离。其中，为进一步提高系统权限分配的适应性以及用户信息安全性，减少用户授权操作的复杂性，采用了基于角色的访问控制（RBAC）策略和 Rijndael（Research on the Rijndael Algorithm）算法设计。结果表明，系统三员分离可以有效的减少管理员权限过于集中给系统带来的危害。

关键词：涉密应用系统；基于角色的访问控制；Rijndael 算法；三员分离设计
中图分类号：TP311.1 **文献标识码：**A **文章编号：**1007-9599 (2013) 01-0010-02

系统管理员的权限过于集中导致信息安全问题越来越受人们重视，为加强公司涉密应用系统的运行的安全管理与审计管理，依据国家保密标准 BMB20-2007《涉及国家秘密的信息系统分级保护管理规范》的有关规定，涉密信息系统应配备系统管理员、安全保密管理员和安全审计员三类安全保密管理人员，分别负责系统运行、安全保密和安全审计工作。三员应该相互独立、相互制约，且每个角色应配置 A、B 角互为备份，安全保密管理员和安全审计员不得由一人兼任。根据以上规定，我们需要为涉密应用系统进行相应的三员分离设计。

1 系统总体设计

涉密应用系统三员分离设计采用 C/S（Client/Server）两层架构，即客户端服务器端架构。其中在客户端完成绝大多数的业务逻辑和界面展示，如图 1 所示，主要需要完成 3 部分内容设计：①登入模块设计，②系统三员分离设计，③Rijndael 加密算法对用户信息加密设计。而服务器端采用的是 Oracle 数据库服务器端，在服务器端进行数据库设计。客户端使用基于组件的数据库编程接口 ADO（ActiveX Data Objects）来访问 Oracle 数据库的数据，通过与数据库的交互（即 SQL 或存储过程的实现）来达到数据的持久化。

此外，为进一步加强涉密应用系统的安全性，除客户端设计之外，特别添加了两项保密措施：应用程序同 oracle 客户端一起打包，以及 Oracle 数据库触发器代码编写。

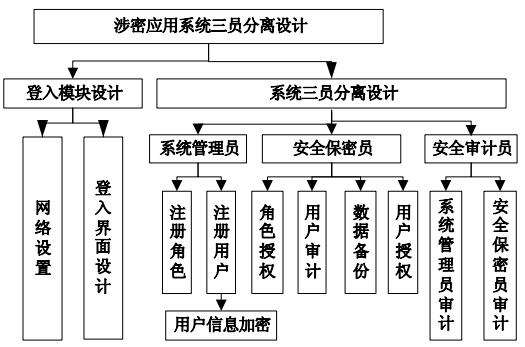

图 1 客户端业务逻辑和界面展示

2 客户端程序设计

客户端程序是系统用户直接使用的程序，其功能是依据系统用户拥有的管理员角色不同，加载不同的功能模块，客户端程序设计就是对每个功能模块的详细设计，其中包括了注册系统用户帐号和角色名称、为用户和角色赋权并使帐号生效、审计用户访问行为与应用系统数据备份与恢复、审计系统管理员和安全保密员的行为等功能的设计，并依照图 2 所示的流程进行系统设计。

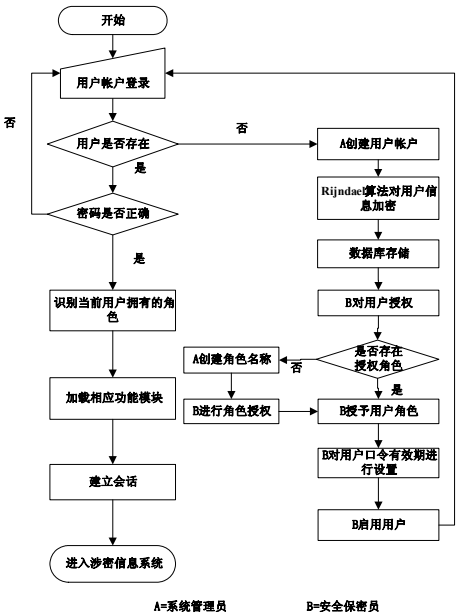

图 2 系统流程设计图

2.1 登入模块设计

登入模块是进入并访问涉密应用系统的第一道防线，是系统统一且惟一的访问入口，在该入口需实现用户认证和身份鉴别。该模块中对未经用户认证及身份鉴别的用户，禁止其访问应用服务，只有获得了涉密应用系统发放的第一道通行证，访问用户才获得继续访问应用服务的权利[2]；登录模块包括两部分设计：登录界面设计和网络设置设计。登录界面设计是用户能否进入数据库的关键，在登录界面中完成用户名及密码的校对以及用户所拥有的角色的识别，依据识别用户的角色加载相应的功能模块。网络设置主要用于识别登录系统的机器类型、局域网内数据库所在服务器的 IP 和所要连接的数据库服务名，是客户端是否能够连上数据库的关键。涉密应用系统通过审核访问用户的

身份信息、拥有的角色（如系统管理员、安全保密管理员和安全审计员等）、安全等级等加载进入到相应操作界面。访问用户只能对同安全级别或低安全级别的应用服务进行访问，且所有违规的访问行为都将被限制和审计。

2.2 系统功能三员分离设计

系统功能三员分离设计主要是对管理员的权限进行控制，根据涉密信息系统分级保护测评标准，采用最小授权原则对系统三员进行系统权限的赋予，使三者相互间形成制约关系，表 1 指出了角色与系统权限的对应关系。

表 1 角色与系统权限对应表

	系统管理员	安全保密员	安全审计员	其他角色
ALTER ANY ROLE	√	×	×	×
ALTER USER	×	√	×	×
AUDIT SYSTEM	×	×	√	×
BACKUP ANY TABLE	×	√	×	×
BECOME USER	×	√	×	×
CREATE ROLE	√	×	×	×
CREATE SESSION	√	√	√	√
CREATE USER	√	×	×	×
DROP ANY ROLE	√	×	×	×
DROP USER	√	×	×	×
EXPORT FULL DATABASE	×	√	×	×
GRANT ANY OBJECT PRIVILEGE	×	√	×	×
GRANT ANY ROLE	×	√	×	×
IMPORT FULL DATABASE	×	√	×	×
ON COMMIT REFRESH	√	√	√	√

2.2.1 系统管理员模块设计

系统管理员模块是用户拥有系统管理员角色才能进入的功能模块。在该模块中需实现两个模块，即角色注册和用户注册。系统管理员可以根据用户书面申请以及保密工作机构的审核结果，在该模块中完成系统角色的创建、修改与删除，用户身份标识符的生成和删除、初始口令设置以及用户信息（用户姓名、性别、联系方式等）的填写。为确保身份标识符在系统生命周期中的唯一性，需对每一个新注册的身份标识符进行唯一性检查，例如系统管理员创建了一个标识符为 User 的用户，通过查询 Oracle 数据库中所有的用户名，判断数据库用户中是否存在标识符为 User 的用户，如果存在该用户，则用户创建不成功，该判断可以用于解决数据库中存在的多重帐号/口令的问题。依据安全保密要求，对用户口令设置采用强口令限制设计，统一约束口令长度、口令复杂度等，最后完善用户信息。同样在角色注册过程中也需对角色名称进行唯一性检查。

2.2.2 安全保密员模块设计

安全保密员模块是用户拥有安全保密员角色才能进入的功能模块，在该模块中完成用户与角色授权、用户审计和应用系统数据备份 3 部分。为了保护系统内涉密信息和重要信息的安全，需要对系统进行严格的访问控制策略。通过对比多种访问控制策略，基于角色的访问控制策略符合 BMB17-2006 信息安全保密要求，其思想主要是在用户 U（文中的用户指的是访问系统的人或计算机）和权限 P（权限可以被理解为对特定的数据对象进行操作的权利如录入、删除、查询、修改等。）之间添加角色 R（角色在逻辑上更接近用户，表示用户在单位的组织结构中的一种身份，岗位职责或组织团体，含有较多的人性化因素。一个角色可以代表一定的职责，即可以被赋予一定的权限。），通过给角色 R 分派若干权限 P（图 4 所示，使角色具有对客体进行相应操作的权利），用户 U 分派若干角色 R（图 3），使用户 U 最终拥有若干角色 R 中的若干权限，对相应的客体进行访问。其中，角色 R 和用户 U，角色 R 和权限 P 之间的分派关系均是多对多的关系。

图 3 所示，在授予用户角色的同时，还需对用户进行口令有效期设计，要求用户定期更换口令，保证应用系统的安全。可以根据实际要求启用或不启用用户，只有启用的用户才有连接数据库和操作数据表的权限。

图 3 用户授权界面

每个启用的用户进入系统获得角色集合的时候，就会建立一个会话。每个会话都是由用户发起的，因此不是静态产生的，而是动态产生的，而且从属于发起其的用户。只要对这些角色与该用户的关系进行过静态定义，那么会话就会根据用户的职责将它所代表的用户映射到多个角色。

条件约束是通过对角色互斥关系与角色基数进行约束，例如安全保密员与安全审计员不能同时赋给一个用户。

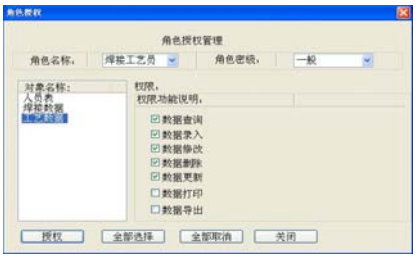

图 4 角色授权界面

2.2.3 安全审计模块设计

同样，安全审计模块只有拥有安全审计员角色的用户才能进入该模块。安全审计主要完成对与安全有关的活动的相关信息的识别、记录、存储和分析。审计记录应包含用户登录信息和数据访问信息，用户登录信息应包括登录时间、用户名、源地点，数据访问信息包括用户名、访问对象、对象密级、时间、操作行为、操作结果等，以确定发生的事件及其来源和结果；且审计事件与唯一的用户标识符关联。审计模块整体框架是把来自各个（审计）数据源，通过审计的数据采集模块，采集审计数据后，通过格式化处理模块把审计信息转化为统一格式，并存储在服务器上，存储审计信息的同时根据审计策略库分析此审计信息是否有异常，如果有异常，存储审计异常信息到异常库方便审计信息审计和提高审计效率[7]。

2.3 Rijndael 加密法对用户信息进行加密

Rijndael 加密法具有 128 比特的分组长度，三种可选密钥长度：128 比特、192 比特和 256 比特。AES 是一个迭代型密码，轮数 Nr 依赖于密钥长度，当密钥为 128 比特时，Nr=10；当密钥为 192 比特时，Nr=12；当密钥为 256 比特时，Nr=14[9]。本文采用密钥长度为 128，比特轮数 Nr=10，采用 AES 算法将定义一个明文 x，将用户信息（State）初始化为 x，并进行 AddRoundKey 操作，将 Roundkey 与 State 异或；依次进行中的 SubBytes（即对 Nr-1 轮中的每一轮，用 S 盒对进行一次代换操作）、对 State 进行一次置换 ShiftRows 和 MixColumns 操作；再进行一次 AddRoundKey 操作，并将 State 定义为密文 y。Rijndael 加密法的成功运用实现了用户信息的加密存储，进一步保护了用户信息安全。

2.4 oracle 客户端打包

为了保证客户端应用程序软件是连接进入数据库的唯一入口，特将应用程序和 oracle 客户端一起打包，禁止用户通过 sql*plus 这个与 oracle 进行交互的客户端工具或 CMD 命令，对数据库内容进行创建、修改、删除等操作，进一步保证数据库内的涉密信息的安全。

3 数据库设计

通过调查和分析系统中数据的使用情况，依据所用数据的种类、范围、数量以及它们在业务活动中交流的情况，确定用户对数据库系统的使用要求和各种约束条件等。在系统三员分离设计中需要设计一张用户信息表，用于保存用户基本信息，包括用户名、性别、联系方式等等，它是访问控制的载体。为保证用户在数据库中的唯一性，防止管理员越过涉密应用系统，而直接使用数据库或 sql*plus 工具创建用户，使用户未能保存在用户信息表中，还需要编写触发器代码，解决该问题。以下为创建用户时执行的触发器代码。

```
create or replace TRIGGER "TR_AC_USER"
AFTER CREATE ON DATABASE
DECLARE
v_user VARCHAR2（50）;
n_exist NUMBER;
BEGIN
--判断若是创建用户，执行此触发器代码
IF Dictionary_obj_type = 'USER' THEN
v_user ：= ''||Dictionary_obj_name||'';
SELECT COUNT（工号） INTO n_exist FROM NH 用户信息 WHERE 工号 = v_user;
IF n_exist = 0 THEN
INSERT INTO NH 用户信息（工号） VALUES（v_user）;
END IF;
END IF;
END;
```

4 结论

通过对涉密应用系统管理员权限过于集中的问题进行分析，系统详细的设计三员职责，采用最小授权原则对系统三员进行授权，编程实现的涉密应用系统三员分离权限管理设计，可以有效的防止由于管理员的权限过于集中所带来的安全隐患，真正做到事前复核、事后审计，降低了出错概率，避免由单个管理员负责权限分配所带来的系统安全风险。

参考文献：

[1]耿伟,吴肖炎.涉密信息系统安全保密管理人员的职责要求与权限划分[J].信息安全与通信保密.2009,29(7):114-115.

[2]施正哗.sso 单点登录模型的优化研究[J].计算机光盘软件与应用.2012,7:190-191.

[3]覃章荣,王强,欧镇进,张超英.基于角色的权限管理方法的改进与应用[J].计算机工程与设计.2007,28(6): 1282-1284.

[4]赵静,杨蕊,姜滦生.基于数据对象的 RBAC 权限访问控制模型[J].计算机工程与设计.2010,31(15):3353-3355.

[5]倪东英,张晓丽.基于 RBAC 的用户权限管理的设计与实现[J].济南大学学报:自然科学版.2010,24(2):167-171.

[6]范收平,高艳.基于三员分离及数据限定的 RBAC 权限管理模型[J].计算机应用.2011,31(2),112-114.

[7]杨光宏,朱行林,黄聪敏.涉密应用系统安全审计解决方案[J].计算机技术与发展.2011,3(21),179-181.

[8]卢正鼎,廖振松.Rijndael 算法的研究[J].计算机工程与科学.2005,27(6),72-74.

[9]邱方亮.航空企业档案资料数据库设计研究及其实现[J].计算机工程与设计.2010,31(9),1954-1957.

[作者简介]黄梁标（1987-），男，浙江省东阳市人，硕士生，研究方向：计算机辅助工艺设计；郭正华（1972-），男，江西省南城市，博士，硕士生导师，研究方向：材料加工数字化技术。