第 34 卷　第 22 期　　计　算　机　工　程　　2008 年 11 月
Vol.34　No.22　　Computer Engineering　　November 2008
·安全技术·　　文章编号：1000—3428(2008)22—0162—03　　文献标识码：A　　中图分类号：TP309

大集中环境下的商业银行信息安全系统

崔颖安[1,2]，陈　皓[2]
（1. 西安交通大学电子与信息工程学院，西安 710048；2. 西安理工大学计算机科学与工程学院，西安 710048）

摘　要：提出全国数据大集中环境下银行综合业务系统信息安全的解决方案，介绍信息安全系统的架构、主要功能和关键技术，包括层次密钥管理、数据认证码、报文认证码、柜员及客户身份认证。该系统已在某国有商业银行的全国数据集中工程 N21-Core Bank 系统中实施，运行效果良好。

关键词：信息安全；密钥管理；数据认证码；报文认证码；共享内存

Information Security System of Commercial Bank in Data Concentration Environment

CUI Ying-an[1,2], CHEN Hao[2]
(1. School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710048;
2. School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048)

【Abstract】This paper presents a solution of the integrated operation system security of commercial bank in data concentration environment. It introduces the framework of information security system, the main function and some key technological problems, including the design and implementation of hierarchy key management, Data Authority Code(DAC), Message Authority Code(MAC) and so on. The solution has been successfully implemented in N21-Core Bank countrywide of a large-scale commercial bank and runs well.

【Key words】information security; key management; Data Authority Code(DAC); Message Authority Code(MAC); shared memory

信息安全历来是银行信息化建设中的关键问题。尤其是在全国数据大集中的计算环境下，不仅要满足数据存储安全、传输安全、客户密码校验、柜员身份认证等基本的信息安全服务需求，还需特别关注系统的运行性能与实施成本，以提高可用性和经济性。本文以某大型国有商业银行全国数据集中工程 N21-Core Bank 中的信息安全系统为背景，介绍了系统设计与实现的若干关键技术。

1　信息安全系统架构设计

银行业主流的信息安全系统架构有 2 种：(1)信息安全处理与账务处理分离的模式，如图 1 所示；(2)信息安全处理与账务处理一体的模式，与模式(1)的区别是没有安全机，业务主机既负责账务处理又承担信息安全服务[1]。

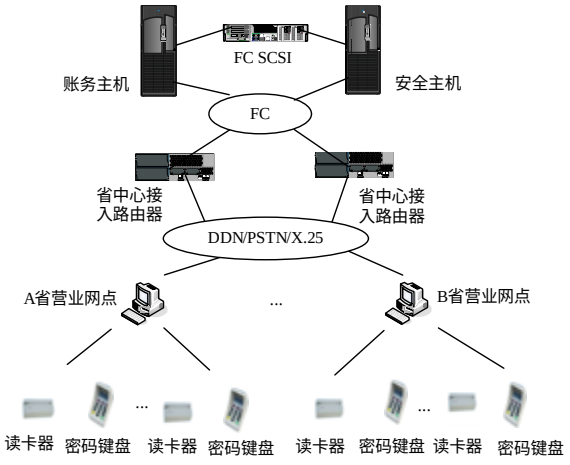

图 1　安全机模式

在模式(1)中，安全机负责银行主密钥的存储、密码校验、MAC 校验等所有的信息安全服务，业务主机仅负责账务处理，账务主机通过 RPC 方式调用安全机上的各种信息安全服务，信息安全处理与账务处理完全分离。安全机和账务主机使用千兆双网卡设置在不同网段，通过子网分离进一步提高系统的安全性。

在全国数据集中环境下，所有业务均提交给数据中心处理，峰值业务约为 2 000 万笔/天，其中约有 3/4 涉及信息安全服务。若采用该方案，必须配备大型机作为安全机。如果再考虑信息安全系统的并行处理、灾难恢复及未来业务处理能力的预留，仅安全机硬件投资就达 6 000 万元左右。由于投资过大，因此模式(1)更适合数据规模相对较小的系统。

在模式(2)中，生产主机既负责账务处理，也承担信息安全服务，不配备专门的安全机。在全国数据集中工程中，生产主机一般按峰值业务的 2 倍~3 倍来配置主机资源，将信息安全服务并入生产主机中并不会过度加重其负担，反而可以更加充分地利用主机资源，这样既满足了信息安全处理的需要又节约了投资。因此，该方案更适合在全国数据大集中的环境下使用。

营业网点有 3 种方案可选：硬件加密卡方式，IC 卡与软件加密相结合的方式，纯软件方式。加密卡中有专用于信息安全处理的 DSP 芯片，虽然它可以较快地完成特定的信息安

作者简介：崔颖安(1975－)，男，讲师、博士研究生，主研方向：面向服务的分布式智能计算，复杂网络，软件工程；陈　皓，博士研究生

收稿日期：2007-12-20　**E-mail：**suchdaysuchpeople@126.com

全服务，但若在所有营业网点和自助设备上均安装加密卡，费用较高且实施难度大(POS 机、部分自动提款机、手机微支付系统等都无法安装加密卡)。使用 IC 卡与软件结合的方式比单纯的软件方式多了一道屏障，提高了安全性，且易于实施，成本低廉，综合以上因素，IC 卡与软件加密结合的方式是最恰当的选择。

2 信息安全系统主要功能

2.1 密钥管理

密钥是信息安全系统的核心，本系统共有 4 类密钥：系统主密钥(Mkey)，银行主密钥(Pkey)，传输密钥(Tkey)及各网点的二级密钥。采用的是层次化密钥管理方法[2]。

系统主密钥和银行主密钥是由 4 位银行高级管理者提供的指纹信息组合而成，这意味着密钥提供者自身也不清楚其输入的具体内容，最大限度地保证了密钥采集的安全性[3]。通过对指纹特征值的提取及变换，系统将Mkey转换为密文，用Mkey加密Pkey，由此生成的这 2 类密钥是系统中最重要的密钥，两者均用 3DES算法加密后存储在数据库中。

传输密钥是用于 MAC 服务的密钥。Tkey 在柜员注册时申请，由主机的密钥生成器使用随机数、时间戳、机构代码、柜员号等唯一性因素散列生成，再由各网点二级密钥加密传送到前端，以密文形式存放在网点前端共享内存中。Tkey 在主机中由 Pkey 加密后存放到数据库及共享内存中，正常情况下，系统使用共享内存中的 Tkey 信息，若共享内存信息失效，则读取数据库中的 Tkey，同时写入共享内存供后续处理使用。

各网点二级密钥存放在柜员的员工 IC 卡中，在制卡时完成二级密钥的初始化，每人的密钥都不相同，一人一密。该密钥在使用一段时间或一定次数后，系统自动提示修改，通过 IC 卡(CPU 卡)内的随机数发生器生成一个随机序列，将随机序列结合一些个性化因素再进行一系列的复杂变换得到新的二级密钥。新密钥在原二级密钥的保护下发送到后端服务器。当前端收到密钥同步成功确认信息后，整个系统启用新二级密钥，同步更新共享内存和数据库中的信息。

层次化密钥管理的优点是密钥种类齐全、信息安全度高，既提高了系统的安全性也兼顾了系统的效率，是一种主流的的应用模式。其示意图如图 2 所示。

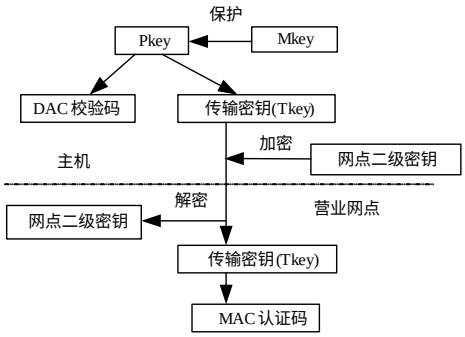

图 2 层次化密钥管理分发示意图

2.2 数据存储安全

在完成数据大集中后，客户的账务信息都集中存放在数据中心的数据库中。从以往的金融犯罪规律来看，银行工作人员的职务犯罪是防范的重点。他们能够利用工作机会访问数据库，对重要数据进行篡改，并能轻而易举地逃避数据库审计，给银行和客户造成巨大损失。针对这种情况，本文采用数据认证码(Data Authority Code, DAC)校验机制确保重要数据在数据库中存储的准确性[4]。该方法的原理是在数据表中增加Dac字段。从表中的关键字段中抽取若干字段组成一个字符串，用银行主密钥加密后再进行一系列复杂变换生成DAC，将结果存放在数据库的Dac字段中。当客户发生金融类交易时，重新计算该记录的DAC，并与表中存放的DAC进行比较，如果不一致，表示数据被非法修改，交易中断，这样就能有效地发现账户信息的异动。以GRZHB(个人账户表)为例说明DAC的生成，其数据结构如下：

字段名	含义	类型	是否参与 DAC 运算
Zh	账号	Char(19)	Y
Pzh	凭证号	Char(19)	Y
Hm	户名	Char(30)	Y
Kl	口令	Char(6)	Y
Csrq	出生日期	Date	N
Wdh	网点号	Char(9)	Y
Yddh	移动电话	Char(11)	N
Khrq	开户日期	Date	N
…	…	…	…
Zjlx	证件类型	Char(1	Y
Zjhm	证件号码	Char(20)	Y
Ye	余额	Decmal(18.2)	Y
Lxjs	利息积数	Decmal(14,)	Y
Yzbm	邮政编码	Char(6)	N
Zhzt	账户状态	Char(1)	Y
Dac	数据验证码	Char(16)	N

抽取 Zh,Pzh,Kl,Hm,Khrq,Zjlx,Zjhm,Ye,Lxjs,Zhzt 等字段构成字符串，使用 3DES 算法对其加密，再进行一系列变换生成 DAC，存放到 GRZHB 的 dac 字段中用于检查校验。关键代码如下：

```
/***  DAC 生成  ***/
Strcpy(Mkey, (char * (readshmmk()); //从共享内存中获得 Mkey
Strcpy(Pkey, (char*(readshmpk(Mkey)); //从共享内存中获得 //Pkey
Rkey=_Des_3( Pkey ); //通过解密，获得 Pkey 真实值——Rkey
Pkey_bcd=change_bcd(Rkey); //Rkey 做 BCD 码变换，方便 //3DES 运算
Strcpy( dac, DES_3( str,Rkey_bcd ); //生成 DAC
_bcd( dac,s_dac ); //对 DAC 反 BCD 运算，混乱 DAC 信息
```

其中，str 为 DAC 字符串。

2.3 报文传输安全

针对数据在网络传输过程中可能受到的攻击，采用报文验证码(Message Authority Code, MAC)确保数据传输的可靠性[4]。该方法的原理是在交易类报文中增加Mac字段(查询类报文不加MAC)，从报文的关键数据中抽取若干位，用传输密钥对其加密再进行一系列复杂变换生成Mac字段，当后端主机收到报文后，采用相同的算法重新计算Mac，并与报文中的Mac进行比较。如果相同，证明该报文合法，交易可继续进行；否则该报文非法，此笔交易作废，这样就确保了关键数据在传输过程中的可靠性。

以对私个人存款交易为例，从 DSLSZ(对私流水表)中抽取 Zh, Jysj, Fse, Wdlsh, Zgh 等字段构成字符串，用 DES 算法对其加密并进行一系列复杂变换生成 Mac，后端收到报文后，使用相同算法重新计算 Mac 并与报文中的 Mac 进行比较，如果数据一致，则进行后续服务；反之则停止交易。使用 MAC 可以确保交易数据在传输过程中的可靠性和准确性。MAC 不负责报文信息的加密，报文的加、解密处理通过 VPN 完成，这样可以充分利用硬件设备的特性，降低软件复杂度，提高

系统的整体性能。关键代码如下：

```
/***     MAC 生成     ***/
Strcpy(Mkey, (char* (readshmmk()));   //从共享内存中获得 Mkey
Strcpy(Pkey,  (char*(readshmpk(  Mkey));     //从共享内存中获得
//Pkey
Strcpy(Tkey, (char * (readshmtk( jgbm ));     //从共享内存中获得
//该机构对应的 Tkey
Rkey=_Des_3( Pkey );   //通过解密，获得 Pkey 真实值——Rkey
Ckey=_Des_3( Tkey );   //通过解密，获得 Pkey 真实值——Ckey
Ckey_bcd=change_bcd( Ckey );     //Rkey 做 BCD 码变换，方便
//3DES 运算
Strcpy( mac, DES_3(str, Ckey_bcd );   //生成 DAC
_bcd(dac, s_mac);         //对 DAC 反 BCD 运算，混乱 MAC 信息
```

其中，str 为 MAC 字符串。

2.4 客户密码个性化处理

客户密码是客户保障自身利益的重要手段，开户时客户留下自己的密码，在发生各类交易时，需要对客户密码进行验证，以确保客户身份的真实性。为了防止银行内部工作人员窃取密码，客户密码在数据库中都以密文的方式存放，并且一人一密，密文绝不重复(即使明文相同)。其原理是截取客户个人账号与凭证号的若干位，与客户密码明文组合成新的密码信息，再以 Pkey 为密钥，采用 3DES 算法对其加密并进行一系列复杂变换得到密文，确保用户密码密文的唯一性。

2.5 柜员认证

为了防止内部操作人员使用他人身份作案，需要对各网点柜员身份的合法性、真实性进行验证。对柜员身份的验证采用柜员密码与员工 IC 卡相结合的双因素检测机制。该方法的原理是为每一位柜员发一张 IC 卡，卡内记录了经过复杂变换的柜员工号和其他柜员的个人基本信息。柜员使用综合业务系统时必须刷卡，同时输入自己的口令，前端软件与 IC 卡相互校验，只有完成 IC 卡和前端软件的真实性校验、口令准确性校验以后，才能进入交易系统正常工作。通过 IC 卡和柜员口令结合认证的方法，可以确保柜员身份的真实、可靠，有效抑制银行内部顶班作案、私设终端等行为的发生。

3 关键技术

3.1 共享内存的使用

本系统采用私钥加密算法，为了提高系统性能，可将使用频率高的密钥存放在共享内存中，便于多个进程间共享密钥，减少不同进程读取相同密钥的 I/O 操作。经测试，共享内存方式比硬盘读取方式的密钥存取效率高出近 100 倍。关键代码如下：

```
/***     IPC——共享内存的创建、使用     ***/
# define KEY_FILE  "/etc/KEY_CTL"   //共享内存 KEY 文本
# define KEY_SIZE  15000         //最大分支机构数
struct    tr_key       //共享内存装载数据结构
{char addr[10];         //机构编码长度 9 位+1 位尾标
char      key[17]; };       //密钥 16 位+1 位尾标
# define          TR_SIZE              sizeof ( struct tr_key )
# define HEAD_SIZE sizeof(int) + 2*sizeof(char)*17
char      *workdir, *getenv();
char      profile[30];
int               shmid,shm_k, *key_num，jgnum;
char      profile[30];
memset ( profile, 0, 30) ;
if ( ( workdir = getenv("HOME") ) == NULL )
return -1;
sprintf ( profile, "%s%s", workdir, KEY_FILE );
shm_k=( ftok ( profile, 256);     //使用 ftok 创建 IPC 唯一关键字
shmid  =  shmget  (  shm_k,  KEY_SIZE*TR_SIZE+HEAD_SIZE,
opperm );   //创建共享内存区
if ( shmid == -1 )       return -1;
shmaddr = ( char * ) shmat ( shmid, ( char * ) 0, 0 );     //将共享内
//存区映射到进程空间
if ( shmaddr == ( char * )-1 )      return -1;
jgnum=get_fzjg( )     //取分支机构数量
for ( i=0; i<jgnum; i++ ){
$ update puttrankey     //数据库中同步最新的传输密钥
set trankey = $q_key
where wdh = $q_address;
if ( sqlca.sqlcode != 0 ){
shmdt ( shmaddr );
return -2;      }
offset = shmaddr + HEAD_SIZE + i*TR_SIZE;       //传输密钥分
//别装入共享内存
strcpy ( tkey.addr, address );
strcpy ( tkey.key, key );
memcpy ( offset, &tkey, TR_SIZE );
jgnum ++;    }
```

3.2 动态链接库

信息安全子系统开发完成后，以库的形式提供给各应用程序调用。库是一种文件，它包含函数或其他可以在应用程序中使用的资源。在Unix中库有 2 种形式：(1)静态库，以.a为文件后缀，它在编译和链接时加载在文件中，是应用程序可执行文件的一部分，不能在多个进程间复用；(2)动态库，以.so为文件后缀，在程序执行时动态加载，一旦载入，其代码可在多个进程间共享[5]。使用动态库可以有效地节省内存，提高程序执行效率。测试证明，动态库的效率比静态库高 20 倍左右，因此，本系统使用动态库。

4 结束语

信息安全系统的开发是一项系统工程，它需要从软件需求、硬件处理能力、算法复杂度、投资成本、未来业务发展的适应性等多方面因素综合考虑，并非加密强度越高、投资越大就越好，只有全面考虑，才能开发出具有较好可用性、可管理性、经济性的信息安全系统。

本系统在 2006 年正式投入使用，自实施以来运行良好，有效地提高了综合业务系统的安全性。通过引入共享内存、动态链接库等技术，明显改善了系统性能。测试表明，该系统能满足 4 000 万笔/天信息安全处理的要求，可以较好地适应当前和未来业务发展的需要，系统所采用的技术方案对同类系统的开发具有参考价值。

参考文献

[1] 林　松, 戴宗坤. 国际实时汇兑系统安全解决方案[J]. 计算机工程, 2005, 31(20): 226-228.

[2] 卫剑钒, 刘　欣, 段云所. 信息系统分布式安全检测模型及其设计[J]. 计算机工程, 2005, 31(20): 141-143

[3] 刘建勋, 杨　鑫, 李恒华. 指纹识别在商业银行系统中的应用[J]. 计算机工程与应用, 2003, 38(3): 201-204.

[4] 范　雯. 信息安全风险模型[J]. 武汉大学学报: 理学版, 2005, 51(S2): 195-198.

[5] 曾　乐, 谢新桂. 一种应用安全中间件设计实例[J]. 计算机工程, 2005, 31(1): 218-221.