

引用格式:徐平江,袁艳芳,付青琴.基于状态机的芯片访问控制实现[J].微电子学与计算机,2019,36(7):98-102. [XU P J, YUAN Y F, FU Q Q. Implementation of chip access control based on state machine[J]. Microelectronics & Computer, 2019, 36 (7):98-102.]

基于状态机的芯片访问控制实现

徐平江,袁艳芳,付青琴
(北京智芯微电子科技有限公司,北京 100172)

摘 要: 本文提出了一种芯片访问控制方法,解决了多应用芯片中文件的访问权限控制问题.该方法首先设计了一种安全权限状态机,该状态机将访问权限细分为 32 级,在用户指令的驱动下安全状态机发生安全状态的变迁,只有安全状态满足预置的文件访问条件,文件访问才被许可.同时为了隔离不同应用的访问权限,定义了环境权限状态机和应用权限状态机.环境状态机控制芯片整体访问权限,应用状态机控制单个应用的访问权限,通过两种状态机的组合应用,实现芯片文件访问控制的精细化管理.

关键词: 安全芯片;访问控制;安全状态;文件系统

中图分类号: TP311 **文献标识码:** A **文章编号:** 1000—7180(2019)07—0098—05

Implementation of chip access control based on state machine

XU Ping-jiang, YUAN Yang-fang, FU Qing-qin

(Beijing Smartchip Microelectronics Technology Company Ltd, Beijing 100192, China)

Abstract: A chip access control method is proposed in this paper to solve the file access control problem of multi-applications chip. Firstly, a security state machine is designed, which subdivides the access right into 32 levels. Driven by user instructions, the security state machine undergoes security state transition. Only when the security state meets the preset file access conditions, file access is allowed. At the same time, in order to isolate the access right of different applications, the environment state machine and the application state machine are defined. The environment state machine controls the access authority of the chip as a whole, the application state machine controls the access authority of a single application, and through the combination of two state machines, the fine management of chip file access control is realized.

Key words: security chip; access control; state of safety; file system

1 引言

随着微电子技术的迅猛发展,各种各样的电子设备如雨后春笋般的涌现.这些设备极大的帮助人们更好的工作、学习,然而这些设备的使用有时候会需要进行用户的身份识别.在各种终端上建立用户名和密码是解决身份认证问题的一个手段,然而这种认证方法将终端与用户绑定,当用户更换设备后

往往无法直接使用.同时一些第三方的服务提供商为终端设备提供附加服务,这些服务需要载入服务商自己的硬件设备.为了实现更多样的身份认证及附加服务,安全芯片类的产品被广泛应用于当前的电子产品中.人们熟知的如智能手机 SIM 卡、金融领域的 U 盾,或者各类不被人知、在设备中默默工作的 ESAM、PSAM 芯片等.

这些安全芯片一般具有独立的 CPU 和具备国

密资质的算法模块,安装到终端主机中为其提供计算服务、数据保存和安全认证功能. 为了实现不同的业务功能,安全芯片会搭载一个芯片操作系统(COS). COS 通常是一个分层结构分为固件系统、存储管理系统、文件系统、安全体系及应用指令系统,如图 1 所示. 其中应用指令中的通用指令部分、安全体系、文件及存储管理部分是 COS 中相对稳定的部分,被称为操作系统平台. 平台在更换固件层后可以在不同芯片硬件上使用,而不同的行业应用是 COS 中最灵活的部分,通常应用功能都需要通过相关行业的检测认证才能给用户 提供^[1-3].

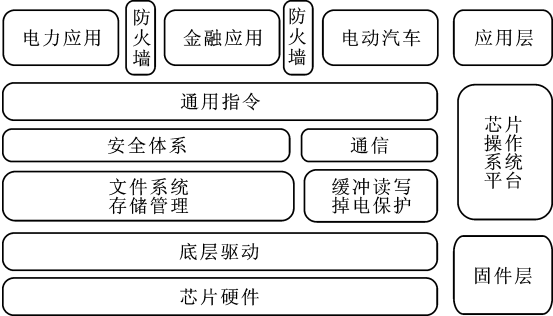


图 1 芯片操作系统架构图

2 安全状态机的定义

芯片中的安全防护主要包括对人员及终端设备身份合法性验证及片内文件访问的权限校验. 通常的方法是芯片执行认证指令成功后获得对应操作的安全权限,然而芯片上需要进行安全控制的操作非常多,为这些操作全部设置相应的权限认证指令过于繁杂、难以实现. 为了实现安全的精细管理,又不增加过多的冗余指令需要设计一个巧妙的安全体系来满足芯片操作权限控制需求.

不同的应用对安全体系的要求不同,本文将实现一个安全功能状态机,通过安全状态机的状态变迁,在不同层次满足应用对安全的要求.

在安全芯片中用户数据通常以文件形式保存,而功能相关联的基本文件(EF)经常保存在一起形成一个应用目录(ADF). 功能相近的应用目录共同形成环境目录(DF),最终应用目录和环境目录共同归属于系统根目录(MF).

安全芯片的基本文件(EF)按文件数据的保存格式可以将基本文件分为两类:

(1)透明文件:文件数据是通过连续空间中的字节地址进行存取.

(2)记录文件:数据以固定格式记录在文件中,文件内最多可以容纳 255 条记录.

图 2 显示了安全芯片内部文件系统的架构,可以看到不同的应用对应不同的文件目录. 对应不同的目录可以为其定义应用的安全状态机 S_{local} ,而对于整个芯片而言定义一个控制环境的安全状态机 S_{global} . 安全状态机定义为一个 8 bit 的全局变量,定义如表 1 所示.

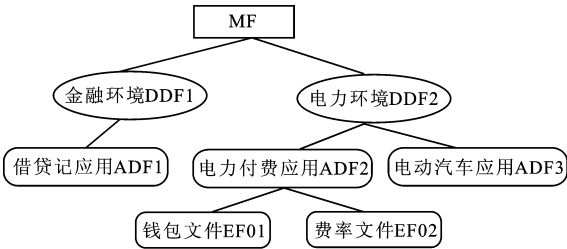


图 2 安全芯片文件系统架构图

表 1 安全状态机数据格式

MSB	Bit7-Bit5	Bit4-Bit0
含义	文件访问模式 M	访问权限 R

为了便于描述,将安全状态机 S 定义如下:
 $S = M \cup R$

安全状态机由两个子状态机组成,其中 M 是文件访问模式状态机, R 是文件访问权限状态机. 下面以图示的方法说明两种状态机的定义.

为了便于描述,引入状态变迁图来说明以上两种状态机的工作过程. 状态变迁图 $L = (R/M, T, F)$,其中 R/M 表示权限状态或者文件访问状态, T 是引发状态改变的指令集合, F 是状态转变弧集合.

图 3 中 $0 \leq R_i < 32$,由于安全芯片中的文件数量较少,因此将其划分为 32 级访问权限.

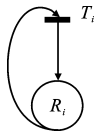


图 3 安全状态机变迁

而 T_i 表示安全芯片可执行的权限命令及其返回结果的集合,即 $T_i = \{APDU \cup SW\}$. 其中 APDU 是可以触发安全状态变化的芯片指令集合,包括校验 PIN 命令、外部认证命令、应用选择指令、芯片初始化指令等. SW 表示指令执行结果的状态

码集合,分为成功状态码和各种不同原因的失败状态码,指令执行结果直接影响芯片状态机当前状态.

图 4 为文件访问模式状态机,由文件访问指令 T_i 触发的文件访问模式一共 8 种,由三种文件基本操作明文读、密文读及 MAC 校验组合而成. 将文件访问状态定义为小于 8 的正整数,则有 $0 \leq M_i < 8$.

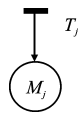


图 4 文件访问模式状态机

可以看到图 3 的安全状态机是封闭路径,而图 4 的安全状态机为开放路径,造成这种区别的原因是状态机的驱动方式不同. 访问权限状态机状态转换和芯片指令的执行结果紧密相关,只有指令执行成功状态机才会改变并影响下一条指令的执行状态. 而访问模式状态机的状态转换只和指令本身有关,指令是否运行成功不会影响状态机,不会对下一条指令有任何影响.

最终安全状态机 S 是文件访问状态 M 和访问权限 R 的集合. 校验状态机 S 时需要分别鉴别这两个分量是否都满足预置安全条件条件.

假设安全芯片的文件访问权限设置为访问上限 $R_{\min}$ 和访问下限 $R_{\max}$,文件访问模式为 M_{spec} ,只有状态机分量满足以下条件才能允许文件访问操作:

$$R_{\min} \leq R < R_{\max} (0 \leq R_{\min} < 32, 0 \leq R_{\max} < 32)$$
$$M = M_{\text{spec}} (0 \leq M_{\text{spec}} < 8)$$

文件访问权限属于文件固有信息,需要保存在文件中. 在安全芯片文件系统设计中,文件通常采用文件头和文件体分离的设计^[4]方法实现. 文件头中保存文件概要信息,便于文件的选择、更新等操作,而文件体保存用户数据. 为了保存安全芯片发行时的为每一个文件确定的访问权限,在文件头中开辟 2 个字节保存其权限,如表 2 所示.

表 2 文件头中保存文件访问权限的字段

Bit15-Bit13	Bit12-Bit10	Bit9-Bit5	Bit4-Bit0
保留	文件访问模式	访问权限上限	访问权限下限

安全芯片执行身份认证相关操作时,通常需要使用密钥进行计算或验证. 认证通过,则用户将获得密钥中规定的安全权限. 同样用户可以为每一条密钥定制安全权限 $K (0 \leq K < 32)$,其值保存在每一条密钥的密钥头信息中^[5-6].

3 安全状态机的典型应用

3.1 芯片初始化

安全芯片在使用之前需要完成两步初始化操作:预个人化和个人化. 通常由芯片厂家完成操作系统(COS)下载和文件系统建立工作,此过程称为预个人化. 用户完成芯片的个性化数据写入,包括密钥数据、身份信息和缺省设置等,此过程称为个人化. 芯片初始化过程如图 5 所示.

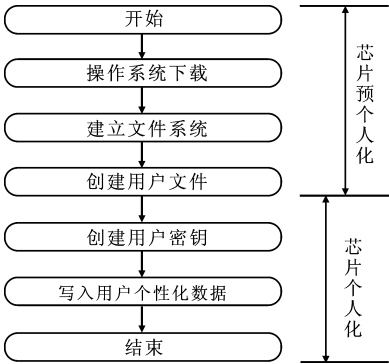


图 5 安全芯片初始化过程

芯片中的文件通常由文件头和文件体组成,结构如表 3.

表 3 文件头结构

File ID	File type	length	File-body address	File-access permission	Key index
2 bytes	1 bytes	2 bytes	4 bytes	2bytes	2 bytes

表 3 中的 file access permission 属性表示该文件的安全属性,该属性在表 2 定义. 当文件被访问时,需要判断安全状态机的当前状态是否满足文件安全属性规定的安全等级.

文件的访问模式指访问文件时,对数据组织形式的要求. 输入或输出数据的组织形式包括明文数据、密文数据、明文数据加校验码和密文加校验码等(在某些应用场合还存在明密文混合打包的情况). 如果用户发给芯片的文件访问指令中隐含的访问模式和文件发行时写入文件头的访问模式不符,此次访问仍然会被芯片拒绝.

文件头中的 Key index 字段表示密钥索引号,通过该密钥索引找到对应密钥进行对文件数据的加密或计算校验码操作^[7-8].

3.2 芯片安全状态机的运行示例

现在通过执行典型的芯片文件访问操作来验证安全状态机是否能正常工作. 假定芯片已经完成发行,文件系统的根目录是 MF,电力应用目录为

ADF01, 数据文件名称为 EF01, EF01 的访问权限为 $R \in (2, 8)$, 文件访问模式明文读, 其属性值为 0 即 $M=0$, 芯片外部认证密钥的权限为 6.

如图 6 所示, 当芯片上电后执行初始化命令 T_1 , 安全状态机为初始状态 $R_1 = 0$. 用户执行选择电力应用命令 T_2 , 安全状态机为应用初始状态 $R_2 = 0$. 此时用户进行身份认证, 发送外部认证命令 T_3 , 芯片收到命令后, 读取外部认证密钥对认证数据进行加密计算并校验. 如果身份认证命令执行成功, 则读取外部认证密钥对应权限 6 并将安全状态机设置为认证通过状态 $R_3 = 6$.

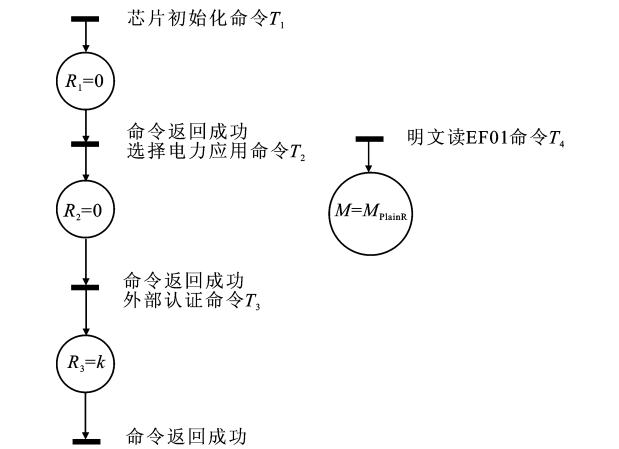


图 6 实际应用中的安全状态机变化

芯片完成上述操作后进行文件访问, 发送明文读 EF01 指令 T_4 , 对文件进行明文读操作, 芯片解析指令后给文件访问状态赋值为明文读, 即 $M = 0$. T_4 指令在读取文件数据前, 将校验已经获得的安全状态和文件发行时配置的条件是否匹配, 即需要满足以下条件:

$$2 \leq R_3 \leq 8 \quad M = 0 \tag{1}$$

只有当安全状态机 S 的文件访问权限 R_3 和文件访问状态 M 都满足式 1 的要求时, 文件访问才被许可.

由以上可以看到, 当用户为每一个文件预置访问权限区间及访问模式后, 文件访问变得可控, 文件能否访问的关键就是用户设置的密钥权限和用户命令隐含的访问模式. 用户可以通过验证 PIN 码、身份认证等方式获得访问权限, 通过命令本身可以解析出访问模式. 同时当应用目录切换或芯片初始化后其权限清除.

安全芯片一个应用环境下可能会承载多个应用, 例如电力环境下会派生电动汽车应用和电表缴费应用. 不同的电力应用会有共同的用户数据文件,

比如用户名信息、密钥信息等. 这些文件作为共有信息保存在应用环境下. 不同应用独有的费率信息、运行参数及钱包文件等必须放在对应的应用目录下. 应用环境由多个子应用组成, 此时用一个安全状态机对不同的子应用进行访问控制显然不能满足安全访问要求.

为了满足不同应用的安全需求, 在芯片中定义了环境安全状态机和应用安全状态机, 如图 7 所示.

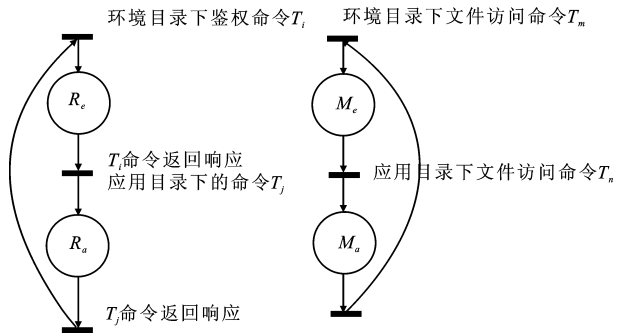


图 7 多应用环境下的安全状态机

在环境目录下, 对文件的访问需要检查环境权限状态机, 当用户选择具体应用以后, 环境状态机变为低优先级, 权限校验优先使用应用权限状态机, 只有该应用没有定义自己的安全状态时才会使用环境状态机作为缺省的安全保护措施.

环境权限状态机为 $S_e = M_e \cup R_e$, 应用权限状态机为 $S_a = M_a \cup R_a$.

4 结束语

本文提出了一种芯片安全防护方法, 根据芯片的应用特点设计了一种安全权限状态机. 该状态机在芯片指令的驱动下实现芯片安全状态的变迁, 个性化的为文件定制访问权限区间及访问状态. 只有安全状态机的权限等级 S 包含在文件访问权限区间且安全状态机的文件访问状态等于预置的文件访问状态时, 文件访问才被许可.

为了细化安全管理, 定义了管辖权限不同的安全状态机. 满足用户在不同应用切换时不同的安全需求.

依据本方法设计的安全芯片 COS 系统, 由于采用模块化分层设计, 具有非常好的可移植性和可扩展性, 而且在功能测试中验证了其良好的安全性. 由于安全保护的实施会带来芯片执行效率的下降, 因此在满足安全需求的条件下尽量简化安全设计是本职工作的目标^[9].

参考文献：

[1] 刘丁丽，张大方，宁佐廷，等. 基于 SM1 算法的文件安全机制设计与实现[J]. 计算机应用与软件，2015，32 (12) :316-320.

[2] FU Q Q. An improved smart card authentication method[J]. Computer Engineering and Science. 2014，36(1)，94-98.

[3] CHEN H J. An implement of the digital certificate on electrical IC card [J]. CRC Press/Balkema，2015，725-727.

[4] 徐平江. 基于链表方式的智能卡文件系统设计[J]. 微计算机信息，2011，27(11):49-50.

[5] 赵东艳，王于波. 一种智能卡写保护机制的实现[J]. 电子技术应用，2014(12):32-34.

[6] 王爱英. 智能卡技术 [M]. 北京：清华大学出版社，2009.

[7] 明华科技. SmartCos-PK-manual Version 1. 0[M]. 深圳：深圳市明华澳汉科技股份有限公司，2000.

[8] 李涛，曾英. 一种新的基于动态口令的远程双向认证[J]. 微计算机信息，2007，11(3):38-40.

[9] 韦小刚，黄益彬，杨维永. 一种基于 ISO/IEC 7816 的 COS 的设计与实现[J]. 计算机技术与发展，2012，22 (3):230-232.

作者简介：

徐平江 男，(1977-), 硕士, 工程师. 研究方向为安全芯片操作系统. E-mail: xupingjiang@126. com.

袁艳芳 女，(1980-), 硕士. 研究方向为智能电网.

付青琴 女，(1982-), 硕士，副高. 研究方向为电力芯片测试.