安全中间件之公共安全操作平台*)

向生建 刘 勇 佘 堃 熊光泽
(电子科技大学计算机科学与工程学院 成都610054)

摘 要 安全中间件是为解决安全的复杂性而引入中间件思想、技术提出的,它与中间件一样,采用了分层的体系架构,从不同的多个层面屏蔽现实生活中所遇到的复杂安全问题,既是平台又是接口,既是框架又是模式解决方案。本文讨论了安全中间件的核心层—公共安全操作平台,以公钥密码加密标准PKCS#11展开设计,强调协作、多平台、多任务的支持。

关键词 安全中间件,公共安全操作平台,公钥密码加密标准(PKCS#11)

Common Security Operating Platform of Security Middleware

XIANG Sheng-Jian LIU Yong SHE Kun XIONG Guang-Ze
(College of Computer Science and Engineering, UEST of China, Chengdu 610054)

Abstract Security middleware is proposed to introduce middleware idea, techonology into security for solving secure complexity. It uses the same hierarchical architecture asmiddleware and hopes to screen the complex secure problems in daily life. It is a platform, and also a interface, at the same time, it is a framework, also a schema solution. In this paper, the core layer—common security operating platform—is discussed, and designed by employing PKCS#11 to emphasize cooperation, multiplatform and multitasking.

Keywords Security middleware, Common security operating platform, Public key cryptographic standard(PKCS#11)

1 引言

据CCID报道,国内安全市场2003年达到了20亿人民币,这包括了防火墙、加密软硬件模块、病毒软件、审计软件、系统恢复软件等。显然,加密软硬件仅仅是安全系统中很小的一部分。其中的深刻原因,就在于信息系统整体化水平不高,需求不高,使得信息系统的增值效应不明显,低水平建设多。当今国内的信息安全公司不下千家,但它们大部分都重复着同样的事情:生产加密设备,包括电话线上、DDN上、帧中继、X.25、IP等链路/路由级加密机,造成国家整体人力、物力资源的大量浪费。因此,公共、通用、可复用、易扩展的安全中间件技术正受到国家相关部门的高度重视。国家创新基金、国家863先后下达了安全中间件方面的课题[1~3]。作为电子政务、电子商务必须解决的问题之一,安全产品通过安全中间件进行互操作、整合和可重用的技术,将在未来1~2年内普及起来。

国外对于安全产品的研究和开发已有较长的历史,企业/个人的安全意识相对较强,安全产品的应用较为广泛,除了传统的加密厂家还在从事这方面的工作外,象IBM、Microsoft这样的业界巨头已经在操作系统、中间件、电子政务/商务工具中引入安全增值服务,在更广的层面上普及安全事业。为此,国外许多厂家都提出了加密设备的互操作标准,如RSA公司的PKCS#11[4]、Microsoft公司的相应竞争产品CryptAPI[5]和Intel公司的CDSA[6]等,这些标准大小雷同,本文以国外常见的PKCS#11作为安全中间件之公共安全操作平台突破口。

公钥密码标准PKCS(Public Key Cryptography Standards)是美国RSA数据安全公司为公钥密码学提供的一个工业标准接口。PKCS包含一系列标准,其中的PKCS#11是加密设备接口标准,该标准详细规定了一个称为Cryptoki(cryptographic token interface的缩写)的编程接口,它可以用于各种可移植的安全设备。Cryptoki给出了一个通用的逻辑模型,用户不需要知道详细的技术细节就可以在可移植的设备上完成加密操作。

PKCS#11规范的提出主要有两个目的:一个目的是屏蔽各种安全实现之间的差异,向用户提供统一的接口,使安全服务的具体实现方法对用户透明化;另一个目的是使用户可以在多个安全服务具体实现之间共享资源,使一个设备可以被多个应用程序访问,一个应用程序也可以访问多个设备。PKCS#11协议中的结构描述正是以这两点为目标展开的,因此PKCS#11不仅提供了安全设备无关的接口,还是安全设备之间互操作、共同协作完成安全任务的平台。

2 PKCS#11设计模型

Cryptoki的通用模型可以用图1来阐述。模型的最上层为一些需要执行密码操作的应用,最下层为执行具体运算操作的密码设备。

Cryptoki通过一系列"slot"提供了对许多密码设备访问的接口,这些设备在slot中处于激活状态。每一个slot对应一个物理读卡器或其他设备接口,它们包括一个或多于一个的"token",Cryptoki仅仅提供了对slot和token的一种逻辑视图,它们也可以代表其他相类似的物理设备,因此多个slot共

*)本文得到国家创新基金、国家火炬计划、国家重点新产品计划和国家军事预研项目支持。向生建、刘 勇、佘 堃 博士研究生,主要研究方向为信息安全。熊光泽 教授,博导,研究方向为实时系统,信息安全。

享一个物理设备也是可行的。问题的关键是一个系统有许多slot和通过插在slot中的token进行连接的上层应用。

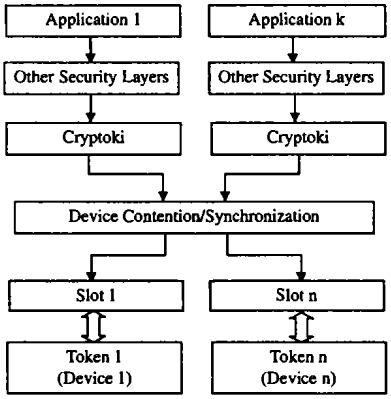

图1

通常密码设备通过特定命令集执行一系列密码操作，典型是通过诸如PCMCIA卡服务驱动或socket服务。Cryptoki使每个密码设备逻辑上看起与其他设备无异。因此应用不需要直接对设备驱动的接口，Cryptoki把这些细节隐藏了。实际上下面说的"设备"可以是纯软件的实现，没有任何硬的东西。

"Device Contention/Synchronization"协调上层应用对多个安全设备的均衡使用，使PKCS#11在简单的安全透明接口基础上，提供了安全互操作的平台。

2.1 用户类型

最新版的Cryptoki提供了两种类型的用户。一种是安全官员(SO)，另一种类型是普通用户。只有普通用户在被授权认证后才可以访问token中的对象。SO的角色是：初始化一个token、设置修改普通用户的PIN；在某些特定情况下SO和普通用户通常为一个人。

2.2 应用和Cryptoki的使用

对Cryptoki，一个应用程序包括一个单独的地址空间，本应用控制的线程都在其间执行。调用C_Initialize一个应用程序开始了一个"Cryptoki application"，其后本应用程序可以调用其他的Cryptoki函数。调用结束后，应用程序通过Cryptoki函数C_Finalize来中止Cryptoki应用。

2.3 应用(Application)和进程(Process)

一般来说一个应用程序包括一个单独的进程。

调用C_Initialize一个UNIX进程P成为一个Cryptoki应用，然后fork()一个子进程C。因为P和C有单独的地址空间，所以如果C需要使用Cryptoki，它需要再次调用C_Initialize。而且如果C需要经由Cryptoki登录Tokens，即使这个时候P已经登录，C也必须做一次登录操作。

2.4 应用(Application)和线程(Thread)

一些应用程序会以多线程的方式访问Cryptoki函数库。Cryptoki要求应用程序提供库需要的一些信息以保证对多线程的支持。特别是，当一个应用程序通过调用C_Initialize初始化Cryptoki库时，它可以指定下面四种多线程模式：

①应用程序不会在多线程方式中使用PKCS#11，这样库可以不使用同步机制。

②应用程序要在多线程方式中使用PKCS#11，库必须使用本地操作系统的同步机制来保证线程安全性。

③应用程序要在多线程方式中使用PKCS#11，库使用应用程序提供的方法来保证线程安全性。

④应用程序要在多线程方式中使用PKCS#11，要使用本地操作系统的同步机制或应用程序提供的方法来保证线程安全性。

在后两种情况中，应用程序要自己提供方法来实现同步机制。它需要提供四个方法来操作互斥对象，包括CreateMutex、DestroyMutex、LockMutex、UnlockMutex。

PKCS#11定义了四个函数指针来指向这四种方法，然后用这四种指针和一个flags来组成一个称为CK_C_INITIALIZE_ARGS的结构。调用PKCS#11的C_Initialize方法时，就是用一个指向这样的结构的指针作为参数。关于这几个函数和结构的具体情况参看PKCS#11文档。根据flags中的CKF_OS_LOCKING_OK位的值和四个函数指针的值可以指定应用程序的四种线程方式。

小结 本文讨论了安全中间件的基础层公共安全操作平台，并根据PKCS#11规范进行了设计和开发。该产品已在信息安全系列产品中获得应用，将诸多信息安全设备——PCI加密卡、密码服务器、USBKey和智能卡——统一在一个可互操作的平台上面。整个产品的代码量达到了30000行C代码。

参考文献

1 国家创新基金项目. 新方向安全中间件. 2001

2 余堃. 国家863安全中间件项目863-301-7-9验收自评估报告. 2002.1

3 余堃，周明天. 安全中间件核心——公共安全服务. 小型与微型计算机，2003，24(7)

4 PKCS #11 v2.11 Final Draft: Cryptographic Token Interface Standard. RSA Laboratories, June 2001

5 Entrust, Implementing Client-side PKCS #11 v2 Libraries for Entrust. http://www.entrust.com/resources/pdf/PKCS #11v2.pdf

6 Microsoft. The Microsoft Internet Security Framework: Technology for Secure Communication, Access Control, and Commerce. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnsecure/html/msdn_misf.asp