基于 TPM 的可信文件系统 CIVFS 的研究和实现*)

张伟伟　石文昌

(中国科学院软件研究所　北京 100080)[1]　(中国科学院研究生院　北京 100039)[2]

(中国人民大学数据工程与知识工程教育部重点实验室　北京 100872)[3]

摘　要　本文提出一种利用可信计算技术增强文件系统可信性的方法，以 Linux 为基础，设计实现了一个可信文件系统原型 CIVFS。CIVFS 是一个结合加密和完整性校验两种保护措施的文件系统，它借助堆式文件系统技术，嵌入在 Linux 内核中，添加了文件加密和完整性校验模块，利用 TPM 芯片提供的可信计算和安全存储等功能，增强了对系统安全组件和数据的安全保护。

关键词　可信文件系统，可信计算，堆式文件系统，TPM

Research and Implementation of a Trusted File System CIVFS Based on TPM

ZHANG Wei-Wei　SHI Wen-Chang

(Institute of Software, Chinese Academy of Sciences, Beijing 100080)[1]

(Graduate University, Chinese Academy of Sciences, Beijing 100039)[2]

(Key Lab of Data Engineering and Knowledge Engineering, Renmin University of China, Beijing 100872)[3]

Abstract　This paper proposes a method for utilizing the trusted computing technology to enhance the trust of file system, and designs and implements a prototype system CIVFS based on Linux. CIVFS combines two file protection measures: encryption and integrity verification. With the stackable file system technology, CIVFS is implemented in Linux kernel. CIVFS adds file encryption and integrity verification modules to the file system, and strengthens the security of system components and data with the functions of trusted computing and secure storage supplied by TPM chip.

Keywords　Trusted file system, Trusted computing, Stackable file system, TPM

1　引言

文件加密和文件完整性校验是两种重要的文件保护方式，分别提供了文件的机密性和完整性保护。它们对文件的保护依赖操作系统对加密模块、密钥和校验值的保护。

可信计算是解决计算机所面临的安全威胁和信任危机等问题的一种关键技术。根据可信计算组织(TCG)发布的规范[1,2]，可信计算部件主要由底层的 TPM 芯片和上层的软件栈 TSS 组成。TPM 是一个独立的计算平台，它提供了与软件相比更安全、更可靠的计算环境。本文设计和实现一种将文件加密和完整性校验结合在一起的文件系统(Cryptographic and Integrity Verifying File System，简称 CIVFS)，利用 TPM 芯片提供的可信计算功能来增强文件系统对文件的保护。

2　相关工作

2.1　常见文件保护系统

典型的加密文件系统有 Cryptoloop、CFS、Cryptfs、EncFS 等多种类型。这些系统普遍存在的一个问题是密钥的管理和保护。以用户口令作为加密密钥或者把密钥存放在磁盘中等方法，都可能导致整个系统被攻破。加密功能模块的安全也是一个重要的问题。文件完整性检查系统可以工作在系统层(PFS)和应用层(Tripware)。Tripware 工具尽管有效地检测出攻击者对文件的修改，但是其缺点也很明显：文件校验值存放在应用空间，可能被攻击者篡改；完整性检测时性能开销很大；不能实时检测入侵行为。PFS 等系统同样存在着如何保护文件校验值，防止重放攻击等问题。将两种文件保护措施结合起来已经被研究人员提出[3]。

以上系统的共同点是与安全相关的功能模块都运行在通用操作系统中，而没有可以信赖的可信计算基。

2.2　TPM 芯片在文件保护中的应用

Trusted Linux Client[4]在文件打开和执行之前进行完整性度量，设置基于扩展属性的强制访问控制规则，从而实现了一个支持完整性度量的可信平台。IMA 和 PRIMA[5]是另一种在 Linux 系统中实施的完整性度量结构，它们把度量点设置在可执行文件、可加载模块、脚本文件执行前，保证了可执行代码的完整。类似地，Dartmouth 大学开发了 Enforcer 模块，提供 Linux 系统下运行时系统文件完整性保护。

上述研究侧重于系统文件的完整性，实现了一个可信的运行环境。与系统中相对固定的可执行文件、可加载模块相比，用户对文件的操作带有很大随意性。对文件的可信管理应该结合文件系统才能有效实现，开发一种可信文件系统对解决用户文件安全存储问题具有重要意义。

3　CIVFS 系统的提出

3.1　设计目标

文件保护系统在保护文件的同时，其自身的安全也是必不可少的，尤其是核心运算部件及其运行的环境。本文使用可信计算技术来解决文件保护系统存在的密钥和校验值的保护问题。类型不同的文件侧重的安全属性不尽相同，例如个

*)基金项目：国家自然科学基金项目(60373054)资助。张伟伟　硕士研究生，主要研究方向：信息安全与系统软件；石文昌　博士，研究员，教授，CCF 高级会员，IEEE 会员，主要研究方向：信息安全、可信计算、系统软件与虚拟机技术。

人隐私、财务数据库文件以机密性为主，而对于可执行代码则需要保护其完整性。

本文设计的 CIVFS 是一个结合加密和完整性校验两种保护措施的文件系统，它的设计目标是：CIVFS 系统可以根据规则来判断文件侧重的安全属性（机密性或完整性），从而选择合适的保护方式；CIVFS 系统实施的加密过程要确保是安全的，整个安全模块有可以信赖的可信计算基支持；CIVFS 系统能兼容通用文件系统结构，而且系统整体性能也应该得到保证。

3.2 工作原理

CIVFS 系统以 Linux 操作系统作为开发平台。Linux 文件系统包含五个主要的数据结构对象，如 file、dentry、inode 等，它的主要特征是虚拟文件系统（VFS）为物理文件系统（如：Ext2）提供了一个统一的接口，内核其他系统都是通过 VFS 来访问底层文件系统。

堆式文件系统（Stackable File System）[6,7] 是一种以现有文件系统为基础开发新功能的有效技术。它通过在虚拟文件系统层和物理文件系统层之间插入一个或者多个处理层来添加新的功能，而整体系统的性能不会有太大的降低。CIVFS 系统采用堆式文件系统结构，以动态模块的形式加载到内核，然后在文件系统中注册，将 CIVFS 系统与底层文件系统的文件对象串联起来。如图 1 所示，水平箭头代表同一层文件数据结构对象的关系，垂直箭头代表上层 CIVFS 系统的文件对象最终调用底层 Ext2 系统的文件对象。

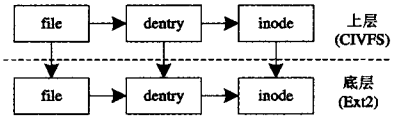

图 1 上下层数据结构对象的关系

3.3 系统框架及功能描述

CIVFS 系统介于 VFS 和 Ext2 系统之间，添加了文件的加密和校验功能。该系统对于上层用户是透明的，而且与底层文件系统兼容。CIVFS 系统自上而下分为三层：用户层、内核层和硬件层。内核层是 CIVFS 系统的主体，包括加密引擎、文件安全属性和密钥管理三个主要部分；用户层为用户提供了访问内核层的接口、系统加载和卸载模块、用户认证和授权模块等；CIVFS 系统通过底层 Ext2 系统访问物理磁盘，而使用 TSS 提供的接口访问 TPM，TSS 包含四层：TPM 驱动程序 TDD、驱动程序库 TDDL、核心服务层 TCS 和服务提供层 TSP。

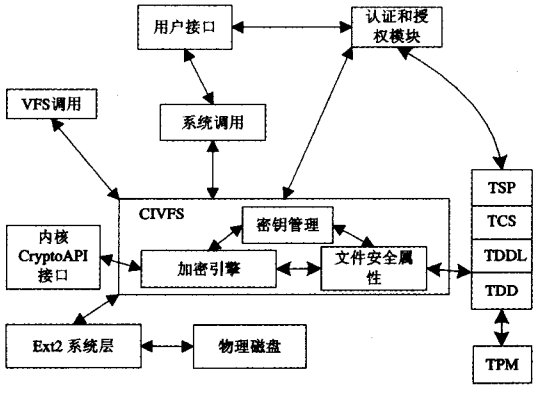

图 2 CIVFS 系统结构

CIVFS 系统的主要功能模块是加密引擎，该模块嵌入在文件系统内核中，完成对文件数据加密或计算校验和等功能；与之相配合的是文件安全属性和密钥管理模块，后两者管理和组织加密引擎中涉及到的用户密钥、文件密钥和校验值等数据；在密钥管理中，CIVFS 系统借助 TPM 的可信计算和安全存储等实现了更安全的保护。此外，认证和授权模块首先验证 CIVFS 系统用户的身份，然后根据用户提供的数据决定是否允许其执行访问 TPM 的操作。CIVFS 系统整体结构如图 2 所示。

（1）加密引擎

CIVFS 系统含有两类加密引擎：一部分文件采用对称加密算法（AES）完成整个文件的加密，另一部分则使用加密散列算法（SHA-1）计算文件的校验值。CIVFS 系统使用 Linux 2.6 版本内核中的 CryptoAPI 接口来实现内核级加密。

（2）文件安全属性

CIVFS 系统扩展文件的安全属性，扩展属性由标志域和数据域两部分组成。标志域长度为一个字节，表示该文件侧重的安全属性（C 代表机密性，I 代表完整性）；数据域包含固定长度的字符串，代表该文件所使用的加密密钥或者该文件的校验和。这些属性保存时需要经过 TPM 芯片加密，防止攻击者对安全属性的窃取和篡改。

（3）密钥管理

密钥管理是加密系统的关键组成部分。CIVFS 系统采用链式加密的方法来保护密钥和文件。其思想是由可信根密钥开始，根密钥加密第一级密钥，第一级密钥作为可信密钥加密下一级密钥，直到加密原始文件数据为止。TPM 提供的密钥生成、加密和安全存储等功能被用于实现该方法。

CIVFS 系统包含三类密钥：可信根密钥 SRK、用户级密钥 URK 和文件级密钥 FK。SRK 是整个平台的可信存储根密钥，它的私钥部分被保存在 TPM 芯片中。每个 CIVFS 用户由 TPM 生成一个密钥对象作为 URK，URK 被 SRK 加密保护。用于授权的共享秘密数据和 URK 对象封装在一起。用户只有在证明他拥有该共享秘密时，TPM 才会授予用户访问 URK 对象的权利。FK 则由内核函数 get_random_bytes() 生成，使用 URK 加密后保存到安全属性中。在使用 TPM 加密密钥时，我们采用了平台环境绑定功能[3]，提高了密钥保护的安全。

（4）认证和授权模块

CIVFS 系统沿用底层操作系统对用户的管理方式，但是添加了用户身份认证和授权模块（下文中简称为 AUTH 模块）。AUTH 模块将用户的认证和 TPM 的认证和授权结合在一起。AUTH 模块首先验证用户的合法身份，然后接收用户设置的共享秘密数据，只有该数据正确时用户才能访问 TPM 并且利用 TPM 解密用户和文件的密钥。AUTH 模块在用户的一次会话期间不需要重新执行，只需在系统加载时或者用户切换时通过系统调用接口进行。与常用的口令认证机制相比，该过程实现了更安全、更可靠的认证。

4 CIVFS 系统核心功能及实现

CIVFS 系统使用堆式文件系统结构，堆叠在通用文件系统（Ext2）之上，增加了安全处理层以实现文件机密性和完整性保护。CIVFS 系统为底层文件系统增强的安全功能主要体现在以下几个方面：更灵活的保护方式、值得信赖的计算基、安全性增强的认证和授权、较高的实时性等。

（1）根据文件的不同安全需求，提供不同的安全保护措施。

CIVFS 系统在继承原有文件属性的同时，还添加了新的

安全属性。该安全属性以 Extended Attributes[8] 方法定义，其内容是该文件的密钥或文件的校验和。CIVFS 系统根据文件的基本属性来选择实施的安全保护策略(机密性或完整性)，一种判断依据可以描述为：如果文件是可执行文件(函数共享库、可加载模块等)，则执行完整性校验；其他属性的文件都进行加密保护。除了该默认规则，CIVFS 系统允许用户动态选择保护策略。

CIVFS 系统采用 Linux 内核提供的 AES 算法，密钥长度为 192 位，CFB 加密模式，以页为单位对文件进行加密/解密操作。或者 CIVFS 系统使用 SHA-1 散列算法，以整个文件为对象校验文件的完整性。当用户打开文件时，计算其校验值与属性中旧的值作比较，判断文件的完整性，出错时选择应对措施：放弃该操作或者提示警告信息等；在释放文件的 inode 节点时，重新计算文件的校验和，更新文件的安全属性。

(2)关键安全组件以可信计算技术为基础，把 TPM 作为可信计算基。

TPM 芯片和 TSS 软件栈是可信计算平台的重要组成部分。TPM 包括随机数生成、密钥生成、加密/解密、签名等功能单元，以及易挥发性和不易挥发性内存等存储单元。

TPM 是一个可以信赖的计算平台，它为 CIVFS 系统提供了多种安全功能，比如可信引导、加密、密钥的安全存储等。可信引导保证了操作系统启动后所处的状态是一个可信的环境，系统所运行的软件不曾被攻击者修改过。TPM 加密与操作系统是隔离的，因而与软件实现的加密函数相比具有更高的安全性。密钥、校验值等重要数据则存储在 TPM 的永久和临时存储单元中。

在设计和实现 CIVFS 系统的过程中，本文采用 IBM 公司发布的一组开发工具，包括 TPM 驱动程序、函数库 Libtcpa 及上层软件栈 TrouSerS 等，搭建可信软件平台，而对于底层 TPM 芯片的功能，则由与上述软件包兼容的 TPM Emulator[9] 来模拟实现(CIVFS 系统经过简单修改后同样可以使用设备厂商生产的 TPM 芯片)。

(3)结合 TPM 芯片实现了更安全的用户认证和授权过程。

TCG 规范规定在使用 TPM 之前，必须首先取得 TPM 的拥有权，这个阶段包括加载共享秘密数据到 TPM 芯片中，生成存储根密钥等操作[1]。共享秘密数据是用户和 TPM 之间认证和授权的凭据。规范中将该数据称为 AuthData，兼有认证和授权的含义。

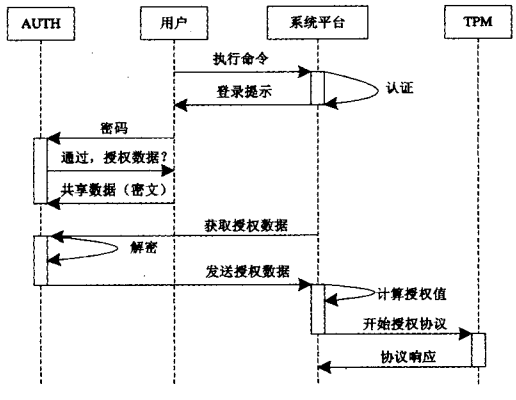

图 3　认证和授权过程

AUTH 模块工作在用户空间，它通过 TSS 访问 TPM。TSS 为每一个被 TPM 保护的密钥对象，定义了相对应的策略对象。这些策略对象支持回调机制，允许实现特定的认证、鉴别方法。我们以 PAM 机制为基础开发了 AUTH 模块，AUTH 模块首先通过用户名和口令的方式验证用户的身份，然后接收用户提供的 secret data，其中，secret data 是共享秘密数据的密文形式。如果用户提供的 secret data 不正确，由 AUTH 模块解密后得到的数据无法通过 TPM 的认证，所以用户最终得不到 URK 和 FK 密钥，进而无法完成文件的加密和校验操作。整个过程如图 3 所示。

此外，CIVFS 系统将安全功能模块嵌入到系统内核中，在一定程度上增强了系统的安全性，同时提高了安全操作的实时性。

CIVFS 系统兼容 Linux 文件系统的语义，提供的文件安全保护操作对上层应用程序是透明的。用户在使用 CIVFS 系统时，需要将一个现有文件系统的目录 A 以 CIVFS 格式加载到目录 B 上，从而分为两层，上层目录 A 是 CIVFS 类型的文件系统而下层目录 B 仍然保留着原有文件系统的语义。用户在访问 A 目录下任意文件之前，需要经过 CIVFS 的认证和授权。用户所访问的文件都由 CIVFS 生成一个扩展安全属性，用户可以采用默认值或者手动设置该属性标志域，CIVFS 系统根据属性标志域决定对文件采取的保护手段。整个安全处理过程对用户是透明的，当出现错误时系统会提示用户，由用户决定采取的补救措施。

结束语　本文设计和实现了一个兼顾文件机密性和完整性的可信文件系统原型 CIVFS。CIVFS 系统针对不同文件类型对安全属性的不同需求，根据规则选择机密性保护或者完整性保护；CIVFS 系统以 TPM 为基础向上扩展安全保护，使得文件的加密或校验过程有了可以信赖的计算基。与已有的类似系统相比，CIVFS 系统实现了更灵活的保护方式，使用可信计算技术增强加密部件和数据、认证和授权的安全性，并且提高了文件安全操作的性能和实时性。

参 考 文 献

1　Trusted Computing Group. TPM Main-Part1 Design Principles - Specification Version 1.2[DB/OL]. [2006-03-29] https://www.trustedcomputinggroup.org/specs/TPM/Main_Part1_Rev94.zip

2　Trusted Computing Group. TCG Software Stack (TSS) Specification - Version 1.2 - Level 1 - Part1: Commands and Structures [DB/OL]. [2006-01-06] https://www.trustedcomputinggroup.org/specs/TSS/TSS_Version_1.2_Level_1_FINAL.pdf

3　Sivathanu G, Wright C P, Zadok E. Enhancing File System Integrity Through Checksums. Technical Report FSL-04-04, Stony Brook University, May 2004. www.fsl.cs.sunysb.edu/docs/nc-checksum-tr/nc-checksum.pdf

4　Safford D, Zohar M. A Trusted Linux Client (TLC). http://www.research.ibm.com/gsal/tcpa/tlc.pdf

5　Jaeger T, Sailer R, Shankar U. PRIMA: Policy Reduced Integrity Measurement Architecture. In: Proceedings of the Eleventh ACM Symposium on Access Control Models and Technologies. Lake Tahoe: 2006. 19～28

6　Zadok E, Iyer R, Joukov N, et al. On Incremental File System Development. ACM Transactions on Storage (TOS), 2006, 2(2): 161～196

7　Zadok E, Nieh J. FiST: a language for stackable file systems. In: Proceedings of the 2000 USENIX Annual Technical Conference. San Diego, 2000. 55～70

8　Ts'o T Y, Tweedie S. Planned Extensions to the Linux EXT2/EXT3 Filesystem. In: Proceedings of the Freenix Track: 2002 USENIX Annual Technical Conference. Monterey, 2002. 235～243

9　Strasser M. A Software-based TPM Emulator for Linux. Department of Computer Science Swiss Federal Institute of Technology Zurich. Summer Semester, 2004