第 42 卷 第 6A 期
2015 年 6 月

计 算 机 科 学
Computer Science

Vol. 42 No. 6A
June 2015

一种面向特定应用的内核级文件加密技术

许国春 殷红武
(江南计算技术研究所 无锡 214083)

摘 要 内核级文件加密系统如 eCryptfs、dm-crypt 等能够有效防止存储介质丢失导致的数据泄露,但它们都未区分访问文件的进程,对于木马程序无防范能力。提出一种面向特定应用的内核级文件加密技术,内核页缓冲只存放密文,仅对指定应用提供明文,杜绝了木马程序获得加密文件明文的可能性,提高了信息系统的安全性。

关键词 内核,加密文件系统,透明加密,木马,安全
中图法分类号 TP309 文献标识码 A

File Encrypting Method on Kernel Level for Specific Application

XU Guo-chun YIN Hong-wu
(Jiangnan Institute of Computing Technology, Wuxi 214083, China)

Abstract Encryption file system such as eCryptfs and dm-crypt can avoid information leakage by storage lost. But they do not distinguish processes accessing the file, so they can not prevent information leakage by the trojan program. This paper introduced a method which puts the cryptograph in the kernel page cache, and only the specific application can access the plain text. This method eliminates the way by which the trojan program accesses the plain text, improves the security of information system.

Keywords Kernel, Encryption file system, Transparent encrypt, Trojan horse, Security

1 引言

用户数据安全在信息安全中至关重要,文件加密将用户数据加密后存放在存储介质,即使介质丢失,也不会造成信息泄露。Linux 的文件加密通常有用户层加密和内核级加密,其中内核级文件加密直接在内核中加解密文件内容,对用户使用透明、安全性好。但传统的内核级加密出于通用性、性能等多方面考虑,在通常的文件系统之上构建虚拟层,并在内核文件页缓冲中存放明文,使得文件内容可能被木马程序盗取。本文在传统内核级文件加密的基础上,提出了一种面向特定应用的内核级文件加密技术,克服了普通内核级加密只针对用户、不区分应用的缺点,在文件系统和页缓冲中存放的都是密文,只有特定的应用才能够得到明文,从根本上杜绝了木马程序得到明文的可能性,提高了信息系统的安全性。

2 内核级加密文件系统

2.1 页缓存

性能和效率是 Linux 内核开发的重要因素,在文件系统中广泛应用了缓存来提高系统速度。页缓存处理内存页,当用户调用系统调用读写文件时,系统先查询目标页是否在缓存中,若命中,则直接读写相应的页缓存;否则,先和设备交互,将相应内容读入页缓存,再对页缓存读写[1]。因为用户读写文件最终都要与页缓存打交道,设计加密文件系统时必须仔细考虑这个因素。

2.2 eCryptfs

eCryptfs 是一个堆栈式加密文件系统[2],通过在传统文件系统之上增加一层虚拟文件系统实现,其结构如图 1 所示。

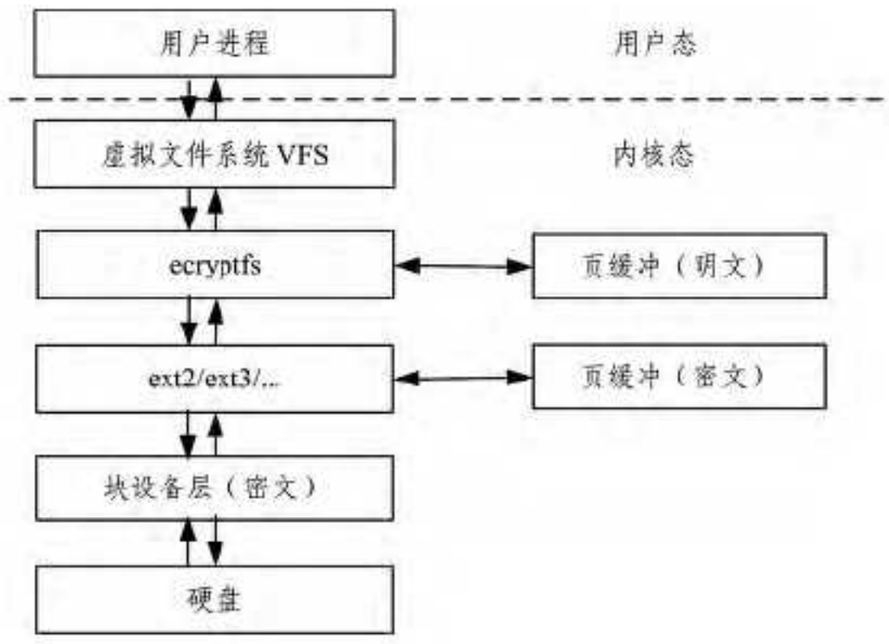

图 1 eCryptfs 结构图

由于是新增加的文件系统,eCryptfs 和底层文件系统(ext2、ext3 等)对应的页缓存是不同的。底层文件系统存储的是密文,其对应的页缓存中也是密文;eCryptfs 中存储的是明文,对应的页缓存也是明文。这种设计使得加密文件系统对用户透明,在性能上也有一定的优势。这是因为只要有一个进程访问过加密文件,该文件解密后的明文就存在 eCryptfs 的页缓存中,此后其他进程访问这个加密文件时,无需再次解密就能从页缓存中获得相应明文。然而,这一设计也带来了安全隐患。eCryptfs 在安全性方面存在两个弱点:

本文受核高基项目(2013ZX01029002-001)资助。
许国春(1979—),男,硕士,工程师,主要研究方向为操作系统安全;殷红武 高级工程师,主要研究方向为操作系统,E-mail:xgcnju@139.com。

(1)如果用户权限设置不当,其他用户可能访问到本用户的加密数据,文献[3]已经针对该弱点提出了改进;(2)如果用户运行了木马程序,由于该用户下所有进程都能访问到该用户的所有文件,可能造成信息泄漏。文献[4]讨论了 window 下文件缓存对加密文件系统的影响,并试图对文件缓存进行清除,但还是存在漏洞。本文针对这个问题给出了更好的解决方案。

2.3 Dm-crypt

Dm-crypt[5]利用 device-mapper 机制实现了基于块的文件加解密。Device mapper 是 Linux 2.6 内核中提供的一种从逻辑设备到物理设备的映射框架机制,为在实际的块设备之上添加虚拟层提供一种通用灵活的方法,在该机制下,用户可以很方便地根据自己的需要制定实现存储资源的管理策略,如条带化、镜像、快照等[6]。

图 2 是 dm-crypt 的加密文件系统结构图。Dm-cyrpt 利用 device-mapper 在普通文件系统和块设备之间增加了一层虚拟的块设备。如图所示,通过虚拟块设备这一层,可以将块设备这一层看到的数据(密文)和上层文件系统页缓冲看到的数据(明文)进行一个变换,从而实现文件的加解密。Dm-crypt 和 eCryptfs 的缺陷一样,dm-crypt 文件系统一旦安装后,文件就以明文方式暴露给所有进程,既可能由于用户权限设置不当造成其他用户可能访问保密的文件,也可能由于用户不小心运行了木马程序造成泄密。

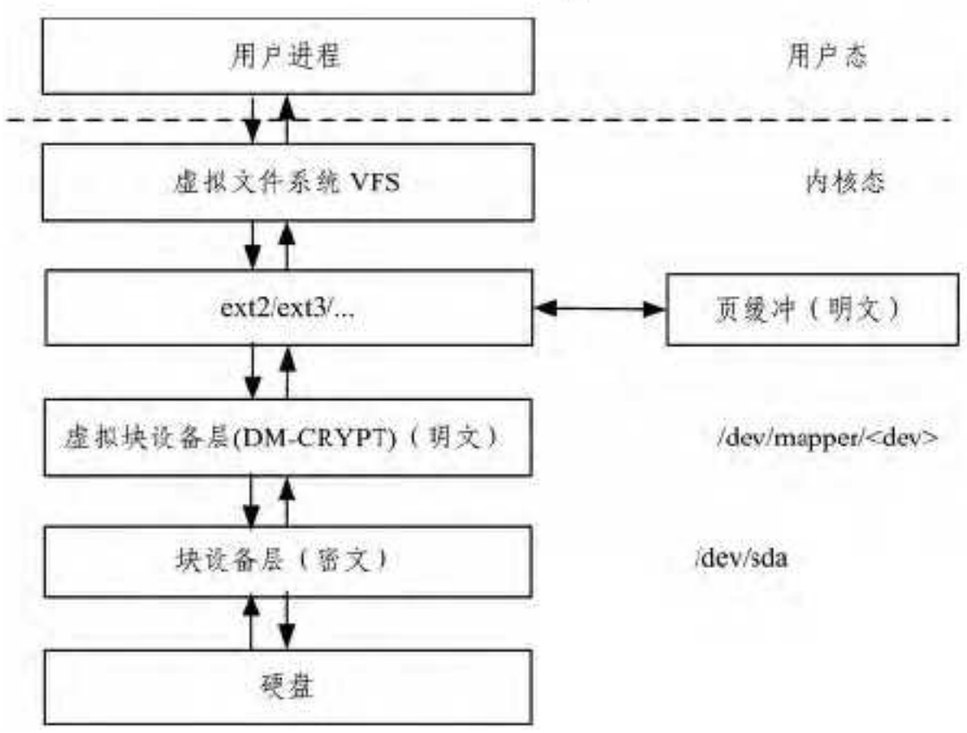

图 2 dm-crypt 结构图

2.4 小结

以上介绍的这些内核级文件加密技术,一般认为对用户层是完全透明的,用户访问文件时不需要知道文件是否被加密,和访问普通文件没有区别。由于页缓存中保存的是明文,使得文件的明文暴露给所有进程,当用户不小心运行了木马程序时,其文件对木马程序完全开放,是这些加密技术中的一个弱点。此外,用户如果要使用这些文件加密技术,用户层必须要运行 mount eCryptfs 命令安装加密文件系统或者 dmsetup 命令进行加密设备的创建,从这个角度看,对用户并不是完全透明的。而且,用户可以人为选择不使用这些技术,就可以绕开这些加密手段,因此这些技术只能做到自主访问控制意义上的安全,没有达到强制访问控制意义上的安全。

3 设计与实现

本文提出的面向特定应用的内核级文件加密技术,只对特定应用暴露明文,系统可以指定信任的应用,例如 office 软件,而未明确指定的其他软件就无法看到该用户文件的明文。而且该技术不依赖特定的文件系统和块设备,强制对需要加密的文件进行加密,普通用户无法旁路该技术,实现了文件的透明加密,无疑提高了信息系统整体的安全性。

3.1 结构描述

本方法总体结构如图 3 所示。

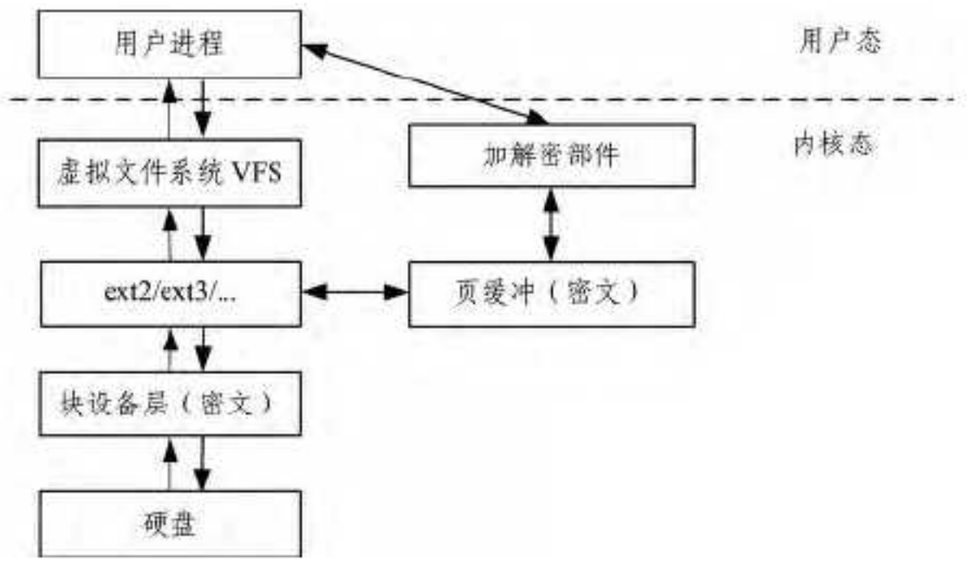

图 3 总体结构图

本文方法并未如传统内核级加密那样在内核文件系统中增加一个层次,而是从数据在核心层和用户层之间流转的角度,进行文件加解密的处理。核心的页缓冲中保存的是文件密文,加解密部件根据进程是否属于指定应用来决定是否对文件对应的页缓冲内容做加解密处理。

3.2 流程描述

3.2.1 系统初始化

在系统初始化时会把指定信任的应用可执行文件全路径名导入内核,系统将这些信息组织为可信任程序列表存放,打开文件时根据该列表判断是否为指定应用。

3.2.2 打开文件

用户打开文件时,系统根据进程对应的可执行程序是否在信任列表中及打开的文件类型决定该文件是否属于需要加密的文件。流程如下:

(1)判断当前进程是否是可信任的指定应用;

(2)根据打开的文件后缀决定是否需要文件加密;

(3)对于需要加密的文件,在打开文件生成的 file 数据结构中设置标记 crypt_file。

3.2.3 读文件

对于读文件,修改系统调用 read 的实现,流程如下:

(1)系统通过 vfs_read 将文件内容读到页缓冲中;

(2)对应的文件是否有 crypt_file 标志,如果没有,将文件内容拷贝到用户层;

(3)如果对应的文件有 crypt_file 标志,读页缓冲内容解密后再拷贝到用户层。

3.2.4 写文件

对于写文件,修改系统调用 write 的实现,流程如下:

(1)系统通过 vfs_write 将用户内容写到页缓冲中,要分两种情况处理;

(2)对应的文件是否有 crypt_file 标志,如果没有,将要写的内容从用户空间拷贝到页缓冲(后续核心会决定最终写入文件的时机);

(下转第 398 页)

参 考 文 献

[1] Eschenauer L, Gligor V D. A Key-management Scheme for Distributed Sensor Networks [C]// Proc. of the 9th ACM Conference on Computer and Communications Security. [S. l.]. ACM Press, 2002: 41-47

[2] Wenliang Du, Jing Deng, Yunghsiang S. Han, Pramod K. Varshney. A pair-wise key pre-distribution scheme for wireless sensor networks [C]// The 10th ACM Conference on Computer and Communication Security(CCS'03). Washington DC, USA, ACM Press, Oct. 2003: 12-21

[3] Liu D, Ning P. Location-based pair-wise key establishments for static sensor networks [C]// Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Security of Ad Hoc and Sensor Networks. Fairfax, Virginia, USA, ACM Press, 2003: 72-82

[4] Eschenauer L, Gligor V D. A key-management scheme for distributed sensor networks [C]// The 9th ACM Conference on Computer and Communication Security(CCS'02). Washington DC, USA, ACM Press, Nov. 2002: 41-47

[5] Blundo C, Santis A D, Herzberg A, et al. Perfectly secure key distribution for dynamic conferences [J]. Information and Computation, 1998, 146(1): 1-23

[6] Wen Mi, Zheng Yan-fei, Ye Wen-jun, et al. A key management protocol with robust continuity for sensor networks [J]. Computer Standards & Interfaces, 2012, 31(4): 642-647

[7] Cheng Y, Agrawal D P. An improved Key Disstribution mechanism for Large-Scale Hierarchical Wire-less Sensor Networks [J]. Ad hoc Networks, 2007: 35-48

[8] Cheng Y, Angrawal D P. Efficient pairwise key Es-tablishmentand manangement in StaticWireless Sensor Network [C]// Proceedings of Moblile Ad-hoc and Sensor Systems Conference. 2005

[9] 成奋华. 传感器网络中基于信誉模型的对偶密钥建立算法 [J]. 计算机应用, 2011, 7(31): 1876-1879

[10] 程伟, 程良伦. 基于信任的无线传感器网络动态密钥管理方案 [J]. 计算机测量与控制, 2011, 19(9): 2315-2318

[11] 程芳权, 彭智勇. 可信云存储环境下支持访问控制的密钥管理 [J]. 计算机研究与发展, 2013, 50(8): 1612-1627

(上接第 394 页)

(3)如果对应的文件有 crypt—file 标志,将用户要写的内容加密后再写入页缓冲。

3.3 文件的密钥存储保护

与 eCryptfs 类似,本方法生成的加密文件在文件中有一页头来描述这个被加密的文件,主要包括这一类型的加密文件标志、被加密的文件加解密密钥,以及使用的加解密算法等。由于多了一页的文件偏移,在 open 文件和 lseek 系统调用中,对文件的偏移需要自动修正为真正的文件偏移,确保对用户使用透明。

文件加解密密钥的保护密钥可以设计为每个用户拥有自己独立的密钥或者整个系统使用一个保护密钥。

4 应用

采用这种面向特定应用的内核级加密文件技术对 office 文件进行加密,配置只有 office 程序能够访问 office 文件的明文,就能确保其他程序无法正常访问 office 文件。office 文件无论在本地还是网络的流转都只能以密文方式进行,只有本地的 office 进程能够得到明文。即使在一个被植入木马程序的系统上,office 文件还是受到很好的保护。

大多数特洛伊木马在被入侵的目标主机上运行服务器端,开启端口等待连接。黑客通过客户端控制木马程序执行各种操作,一般木马在进驻目标机器后通过网络与外界通信,发回所搜集到的各种敏感信息,并接受黑客的指令完成其他各种操作。比如 Linux. Backdoor. Kaiten[7] 和 Linux. Backdoor. Rexob[8] 就是这种类型的木马。但是这些木马只能访问 office 文件的密文,无法获取 office 文件的明文,office 文件就不会真正被泄露。

结束语 本文提出了一种面向特定应用的内核级文件加密技术,改进了 eCryptfs、dm-crypt 等内核级加密文件系统可能由于木马程序造成信息泄密的问题。下一步将对该技术进一步扩展,将可信技术、利用硬件可信根的度量及硬件的加密能力用于用户信任的可执行文件度量和文件的加密,进一步提高信息系统的安全性。

参 考 文 献

[1] Wolfgang Mauerer. 深入 Linux 内核架构[M]. 北京: 人民邮电出版社, 2010

[2] Halcrow M A. ecryptfs: An enterprise-class encrypted filesystem for linux[C]// In Proceedings of the Linux Symposium. Ottawa, Canada, July 2005: 201-218

[3] 唐晓东, 付松龄, 何连跃. 基于 eCryptfs 的多用户加密文件系统设计和实现[J]. 计算机应用, 2010, 30(5): 1236-1238

[4] 陈忠贵, 舒远仲, 吴文俊. 加密文件系统中缓冲技术的研究[J]. 南昌航空大学学报, 2010, 24(2): 67-71

[5] Peters M. Encrypting partitions using dm-crypt and the 2.6 series kernel[OL][2004-6-6] http://archive09.linux.com/feature/36596

[6] Red hat. Logical Volume Manager Administration. Appendix A. The Device Mapper[OL]. [2013-09-29]https://access.redhat.com/documenation/en-US/Red—Hat—Enterprise—Linux/6/html/Logical—Volume—Manager—Administration/device—mapper.html/

[7] Symantec. Linux. Backdoor. Kaiten[OL]. http://symantec.com/security—response/writeup.jsp?docid=2006-021417-0144-99&tabid=2

[8] Symantec. Linux. Backdoor. Rexob. [OL]. http://symantec.com/security—response/writeup.jsp?docid=2007-072612-1704-99&tabid=2