基于密码的访问控制研究 *

马 康, 陈松政
(国防科学技术大学 计算机学院, 长沙 410073)

摘 要: 针对现有访问控制策略和机制复杂的特点,结合标签机制和多级安全的策略,以强制访问控制为基础,提出了一种新的访问控制机制。该机制的思想是根据客体的访问密钥来最终决定主体对客体有何种访问权限。基于这种思想将访问控制策略和机制进行了设计,给出了一种在 LSM(Linux 安全模块)框架下基于密钥对文件进行访问的策略实现方法。该访问控制方法通过设置密钥给了用户一定的自主权,限制了高级用户的部分权限,实现了文件共享。

关键词: 机密性; 访问控制; 共享密钥; Linux 安全模块; 多级安全
中图分类号: TP309.2 **文献标志码**: A **文章编号**: 1001-3695(2012)01-0305-03
doi:10.3969/j.issn.1001-3695.2012.01.084

Research on cryptography based access control

MA Kang, CHEN Song-zheng
(*College of Computer*, *National University of Defense Technology*, *Changsha* 410073, *China*)

Abstract: For solving the complexity and difficulty problems in configuration in access control mechanism, this paper introduced a MAC(mandatory access control) mechanism which combined label mechanism and multilevel security. CBAC described how a subject might accessed an object by the specific key. Based on this method, it gave a design for the new access control policy and mechanism, also the policy and realization for CBAC on LSM frame work. As a result, by setting keys, it limited the super-user's ability, put appropriate rights to the normal users and achieved file sharing.

Key words: confidentiality; access control; shared key; Linux security model; multilvel security

安全操作系统[1]是在操作系统上增强了安全机制的系统,它用来保护计算机系统上资源的保密性、完整性和可用性。基于 Linux 的开源操作系统是主流的操作系统之一,得益于全球开发者的参与,其安全性和稳定性得到越来越多用户的认可。Linux 的主要安全技术包括 Linux 能力机制、PAM 机制[2]、入侵检测、加密文件系统、安全审计、基于 ACL 的自主访问控制、强制访问控制[2]等。

从世界上第一个安全操作系统 Adept-50[3]到最近几年研究比较多的 SELinux[4],Linux 系统安全技术取得了很大的进展。但是,Linux 安全还有很多问题有待解决,如超级用户权限过大、安全策略和模型比较复杂、难以配置和应用、GPL 规则认识不统一等。现在,开发人员可以在 LSM[5]框架下灵活地根据自己的安全策略实现相应的可加载内核安全模块,用户也可以根据其需求选择适合的安全模块加载到 Linux 内核中,简化了模块的开发过程。本文正是基于 LSM 安全框架,提出了一种轻量级的密码访问控制机制并给出实现过程。

1 机关研究

1.1 密码访问控制的一些相关研究工作

2003 年,丹麦科技大学的研究组提出了一个针对分布式文件系统密码学安全访问控制算法[6]。该算法将加密算法和访问控制算法相结合来提高传统访问控制算法的机密性,同时提供完整性保护。算法是基于开放的网络体系,利用加密密钥、对称密钥、解密密钥来控制读和写两种访问。该算法计算复杂度比较高,而且系统存储负荷也较大。此后的一些基于密码的访问控制机制也都是基于密钥和混合加密算法。

2010 年,美国肯塔基州立大学的 Al-Hamdani[7]提出了一种基于 RBAC 的 PKI 访问控制模型。该模型是面向医疗网络系统这一具体应用,其主要特点是可以支持群组和交叉认证。

以上都是基于网络和 CS 模型具体应用的密码访问控制。在操作系统内核体系结构上,众多开发者也做了关于密钥方面的大量工作并取得了一些成果,如典型的内核密钥管理系统调用 Keyctl。Keyctl 提供了丰富的密钥管理函数,如连接一个密钥到密钥环:keyctl link〈key〉〈keyring〉。内核密钥保留服务[8]是一个在 Linux 2.6 内核中引入的比较新的方法,用于在 Linux 平台上处理身份验证、密码学等安全问题。

1.2 访问控制策略和模型讨论

传统的访问控制主要有 DAC 自主访问控制、MAC 强制访问控制以及最近几年研究比较多的 RBAC[9]基于角色的访问控制。自主访问控制根据访问者的身份和授权来决定访问方式,访问主体对访问控制有决定权利。这种权利容易使得访问权限关系在信息移动过程中发生改变,很容易产生安全漏洞,所以安全级别较低。强制访问控制是系统强制主体服从的控制策略,它将主体和客体分级,然后根据主体和客体的级别标记来决定访问模式。它的缺点是过于偏重机密性,实现工作量

收稿日期: 2011-06-07; **修回日期**: 2011-07-28 **基金项目**: "核高基"重大专项基金资助项目(2010ZX01036-001-001)

作者简介:马康(1986-),男,河北石家庄人,硕士研究生,主要研究方向为操作系统(makang0506@126.com); 陈松政(1971-),男,安徽祁门人,副研究员,硕士,主要研究方向为系统软件和计算机安全.

大,管理不方便。基于角色的访问控制是对前二者的改进,基于用户在系统中的作用规定其访问权限。

传统的访问控制一直以来是主要的安全防范机制,但已不能满足日益复杂的安全威胁和日益丰富的私有保护需要,因此便产生了加强的安全策略—SELinux。SELinux 定义了一种混合的安全性策略,由类型实施(TE)[10]、基于角色的访问控制(RBAC)和可选的多级别安全性(MLS)组成。对于目前可用的 Linux 安全模块来说,SELinux 是功能最全面的,而且也是测试最充分的。但是,SELinux 的策略配置语言非常复杂。虽然能够很好地保护系统的完整性,但是最终用户和系统管理员使用起来非常不方便。因此,在灵活的安全配置和正确的用户设置之间存在一种矛盾,现有的安全策略往往不能两者兼顾。事实上,用户把安全视为一种支持工作而不是最终目的。所以,设计一种容易被广大开发者和用户理解与掌握的安全机制是研究的关键问题。本文正是基于这种现状,根据现有的密码访问控制机制结合自主访问和强制访问控制,设计出一种易于配置和使用的密码安全策略。

2　CBAC 的策略和机制

2.1　密钥管理

基于密码的访问控制机制的基本思想是为每一类资源提供一个密钥(类密钥或节点密钥)。每个用户都有一个所属类以及相对应的密钥,用户访问资源时,必须提供密钥。密钥的生成算法可以选择 MD5、SHA-1 等现有的比较成熟的算法。要保证密钥只能被用户本人读取就要制定密钥安全策略。CBAC 考虑了两种密钥存储方案:a)存储在 USB 设备中;b)存储在系统内核中,两种方案可以结合使用。本文主要讨论了后者,将密钥进行加密并以标签的形式和客体相关联,生成的密钥则和主体关联存放在内核中。用户的密钥除了系统安全分级设定的权限外其他设置仅由用户本身管理。如用户可以自行生成密钥、更改密钥、设置密钥共享以及撤销密钥。

2.2　访问控制策略

对文件的强制访问控制是通过文件的安全级别和访问密钥来实现的。首先,根据多级安全思想对主客体进行分级。参考 BLP 模型,按 UID 将主体(用户)分为多个安全等级,根据 UID 取值按级别分段,如超级用户、一般用户、远程登录用户,三种用户安全许可级别由高到低 UID 值由低到高。用 UID 标记客体(文件)的属主和安全级别。这里与 BLP 不同的是仅仅限制向上读。设 $L(s)=l_s$ 是主体的安全许可级别,$L(o)=l_o$ 是客体的安全级别,二者统称为密级。对任意安全密级 $l_i=0,\cdots,k-1$,有 $l_i<l_i+1$。如果 $L(s)<L(o)$,则不允许该主体对客体的任何访问。对于相同密级之间的访问以及高密级主体访问低密级的客体时,都要进行密码认证才允许访问。对于远程登录用户(如 PPP 登录和 FTP 登录)要设置较低的安全级别以限制其对资源的访问。此外,安全级别也可以通过密钥的信息来体现。

CBAC 安全策略的目的是对所有受保护的文件进行访问都通过密钥验证。将文件的访问分为三种:R(读)、W(写)、X(执行)。首先,由主体指定用于读、写和执行的三个密钥,这三个密钥是默认的用于三种访问的认证密钥。当用户对所拥有的文件设置访问权限时文件会关联这几个密钥。同时,文件也会标记属主的 UID 和安全密级。最后为了提高密钥的机密性,可以将三种访问密钥分别用口令加密,然后再存储到文件的扩展属性里并提示用户记录口令,扩展属性设置为不可读取。对于本地用户,访问时需要用户输入相应的口令,系统会根据输入的口令解密访问密钥然后进行密钥匹配。如果输入口令不正确即使是管理员用户也不能访问受保护的文件。对于非本地登录用户,生成的密钥被口令加密后发送并保存在远程用户主机上,访问时需要提交密钥或口令来进行文件访问。其中,口令的任何形式都不会在系统中存储,只是临时用来加/解密。安全等级验证和密钥验证是在强制访问控制阶段进行的,即在 DAC 自主访问控制检查之后的强制访问控制阶段。

2.3　文件共享

为了实现灵活的文件管理,CBAC 制定了密钥共享机制以实现文件的共享。密钥共享主要遵循以下原则:a)共享必须获得文件属主授权,并由文件属主设置共享密钥;b)共享密钥对非属主用户为不可读,即只允许用户有使用权;c)共享密钥遵循最小权限原则,尽量不共享写密钥;d)组内密钥原则,即如果文件可以被某个群组所有用户共享,需要创建者分发密钥以允许组内所有用户共享该组的共享文件;e)密钥共享遵循多级安全的规则。

CBAC 的密钥共享不是传统意义上的共享,共享密钥只是为其他用户提供临时的链接,除了共享密钥的属主外,共享用户无法访问共享密钥,也无法对其进行更改等操作。共享密钥一般都由创建者设置生存期,生存期过后,密钥链接失效。共享文件一般安全级别较低,所以一般不再需要输入口令。

3　CBAC 的设计与实现

3.1　CBAC 的系统结构

本文基于密码的访问控制主要是参考已有的密码访问控制[11]策略,在操作系统级别上进行设计,结合了 DAC 和 MAC 两种访问控制策略的思想,给用户一定的自主权限,遵循了最小权限原则。当一个主体访问客体时(以用户访问文件为例),相关系统调用会触发安全决策机制,安全决策机制首先根据安全请求的类型确定应采用的安全政策,再把决策任务转交给对应的政策支持机制,最后将决策结果返回给决策实施机制去执行。CBAC 策略分为三个主要部分:a)密钥访问策略,主要规定了文件的各种访问方式以及密钥的管理;b)分级安全策略,将文件按安全等级进行分类,主要是防止向上的访问;c)文件共享策略,制定了文件共享的机制,并根据不同的应用场景采用不同的策略。以上三个策略通过构建 LSM 模块以及在新的模块上重新编写 CBAC 的钩子函数来实现。访问控制框架如图 1 所示。

3.2　CBAC 的实现模块

3.2.1　模块

基于密码的访问控制实现模块主要考虑以下几个问题:

a)密钥的生成和管理。由于是在 Linux 内核实施访问控制,因此首先要将用户生成的密钥打入内核。可以采用 pam 模块的方法[12],编写 pam 动态库文件将用户的密钥打入内核密钥环,根据系统调用 keyctl 对用户的密钥进行管理。

b)根据密码访问控制策略实现 LSM 安全模块和钩子函数的设计。用户对文件进行访问时触发系统调用,首先会进行一

些基础的认证和检查,最后由 CBAC 的模块的安全钩子进行安全检查。这些钩子函数是根据 CBAC 的策略重新定义的。例如,对文件进行读访问时需要输入读的口令来解密读密钥,解密后的密钥跟指定的读密钥进行匹配,在其他检查通过后,如果提供的密钥也是正确的,访问才会被许可;否则不允许对该文件进行读操作。以上内容可以通过 cbac_file_permission 具体实现:security_operations. file_permission = cbac_file_permission。

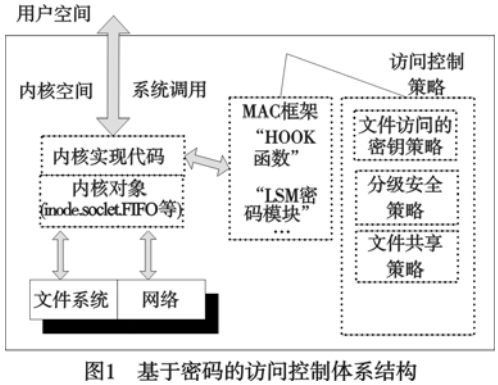

图1　基于密码的访问控制体系结构

c)共享。用户可以将访问密钥临时连接到其他用户的密钥环以使得文件可以被共享。用户自己可以设定密钥的有效期来控制共享的时间范围,当共享密钥到期后连接自动撤销。当文件的非属主用户通过共享密钥的方式来访问文件时一般不需要输入口令。为了加强安全,可以在共享密钥之前对访问主体进行身份认证。

根据以上内容,CBAC 包含两个主要模块,即密钥管理模块和强制访问认证模块。具体如表 1、2 所示。

表 1　设置密钥

分类	文件权限设定模块
1. 文件属主验证	验证是否为文件的合法拥有者(创建者)
2. 密钥指定	指定 R、W、X 访问分别对应的密钥
3. 授权	用密钥关联受保护文件
4. 设置共享	设置共享密钥并连接到可用的密钥环

由表 1 可知,该模块实现了文件属主对文件的保护设置以及对文件共享的设置,可以实现较细粒度的访问控制。如要设置某个文件只允许读操作,只要将读密钥关联该文件:set-r file_name R_key。

表 2　强制访问验证

分类	访问验证模块
1. 账户验证模块	验证用户账号是否符合安全规则
2. 验证安全级别	是否越级访问,防止非法向下访问
3. 文件访问验证	是否为文件属主,验证口令
4. 密钥共享模块	验证是否允许共享

由表 2 可知,该模块主要实现各种安全认证,确保访问是合法操作。

3.2.2　*控制流程*

当用户 usr 要把某个文件共享给 usr1 时需要根据 usr1 的 UID 将文件的访问密钥链接到 usr1 的密钥环,或者重新设定文件的访问密钥作为共享密钥。共享密钥或者共享链接的生存期需要限定在一个相对较短的时间内。如果 usr 通过密钥设置允许 usr1 在时间段 *P* 内读取文件 file_name,则模块主要实现以下功能:

a)设定新的 *R* 密钥,不能用口令加密,将其连接到 *B* 的密钥环。

```
reset file_name R_share_key;
Link R_share_key usrB_keyring;
```

b)设定密钥的生存期。

```
R_share_key. time_t = p;
```

c)撤销过期的密钥。

```
ulink R_share_key usrB_keyring;
```

Usr1 在访问文件时,首先会被检查文件属主是否进行越级访问,其次检查文件是否允许共享,然后再搜索密钥环中可用的密钥直到找到匹配的密钥。为了加强安全强度,可以设置让 usr1 输入自己的用户名和密码来通过认证,认证通过后再允许用户共享密钥。该密码访问控制机制下,主体对客体的主要访问流程如图 2 所示。

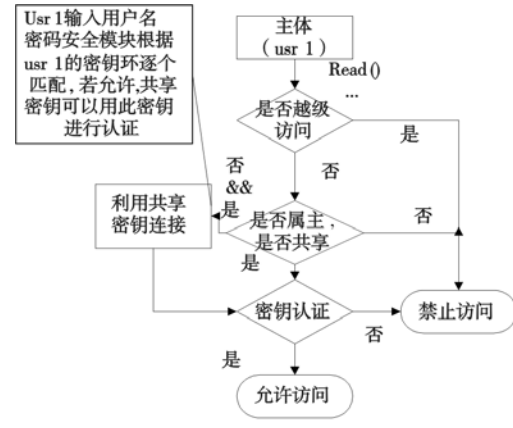

图2　CBAC文件访问控制流程

4　安全性分析

4.1　登录认证及内核密钥的安全性

用户的密钥必须保证是用户自己生成,因此每次生成密钥必须是在输入用户名和密码之后一段时间内。对于本地登录用户,用户输入账户名和密码后,系统根据其 UID 链接属于该账号的内核密钥环。同时,PAM 模块对账号和密码强制验证并生成用户的内核密钥。对于远程登录用户,可以先利用 PAM 认证和 access 机制来实现控制允许某些用户在特殊时段通过 SSH 登录(如通过 IP 网段限制),然后在本地存储的用户列表中查询远程用户的可用密钥并产生链接。远程用户生成的密钥通过 SSH 发送给用户主机,远程用户通过身份认证登录以及各种安全检查后,还需要提供对应的密钥才能访问本地资源。对文件进行加密虽然可以更好地保证数据的机密性和完整性,却大大增加了实现的复杂度,尤其是在对大量的文件进行保护的情况。CBAC 采用密钥认证的方法只对密钥进行加密,降低了加密的时间复杂度,同时也使机密性得到保证。

4.2　多级安全和密码认证混合策略

CBAC 在密码认证之前先采用多级安全机制阻止非法的越级访问,这样可以大大减少密钥的设置操作。如系统文件通常只需要超级用户访问,这些文件不必设置密钥也可以阻止一般用户访问。超级用户之间也可以采用密钥的方法防止相互之间对秘密文件的访问。对于远程登录用户,对其设置较低的安全级别并禁止超级用户的远程登录。与多级安全不同的是 CBAC 除了不允许向上越级访问外,对于向下的访问也默认是要通过密钥认证的。

(下转第 315 页)

验结果可见,加密算法本身对椒盐噪声和剪切攻击有一定的抵抗能力,而抗攻击算法对这两种攻击有较强的抵抗能力。

7　结束语

本文提出了一种新的基于 Jerk 系统的混沌图像加密算法,Jerk 系统产生的混沌序列表现出理想的伪随机特性。仿真实验结果表明,该算法可以实现对彩色数字图像的加、解密,并且达到很好的加密效果。从算法安全性的分析中可以看出,该算法密钥空间大,有极强的密钥敏感度和非常好的统计特性,并且抗攻击算法具有较强的抗椒盐噪声和剪切攻击能力。

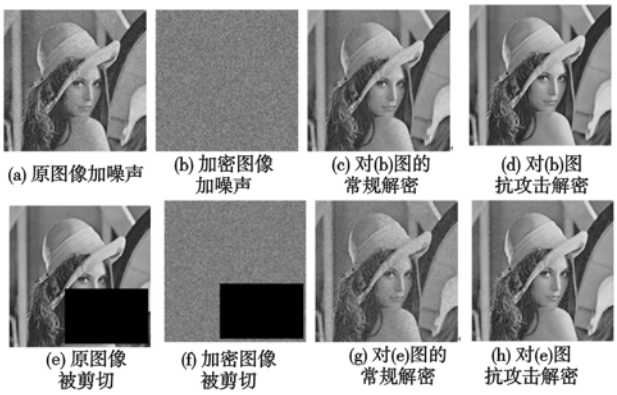

图7　加密图像受干扰的分析

参考文献:

[1] MATTHEWS R A J. On the derivation of chaotic image encryption algorithm[J]. Cryptologia,1989,13(1): 29-42.

[2] FRIDRICH J. Image encryption based on chaotic maps[C]//Proc of IEEE International Conference on Systems, Man and Cybernetics. 1997:1105-1110.

[3] WONG K W, KWOK B S H, YUEN C H. An efficient diffusion approach for chaos-based image encryption[J]. Chaos, Solitons and Fractals,2009,41(5):2652-2663.

[4] FAN Jiu-lun, ZHANG Xue-feng. Image encryption algorithm based on chaotic system[C]//Proc of the 7th International Conference on Computer-Aided Industrial Design and Conceptual Design. 2006: 113-117.

[5] CHEN G, MAO Y, CHUI C K. A symmetric image encryption scheme based on 3D chaotic cat maps[J]. Chaos, Solitons and Fractals, 2004,21(3):749-761.

[6] 邱燕,陈斌峰,王艳阳. 基于二维超混沌系统的数字图像加密算法[J]. 陕西国防工业职业技术学院学报,2007, 17(4):17-21.

[7] ZHU Zhi-liang, ZHANG Wei, WONG K W, *et al*. A chaos-based symmetric image encryption scheme using a bit-level permutation[J]. Information Sciences,2011,181(6):1171-1186.

[8] TAHA B A, FOURNIER-PRUNARET A K, BOUALLEGUE D R. A fast color image encryption scheme based on multidimensional chaotic maps[C]//Proc of the 2nd International Global Information Infrastructure Symposium. 2009:1-4.

[9] GMBSS K, CHANDRASEKARAN V. A novel image encryption scheme using Lorenz attractor[C]//Proc of the 4th IEEE Conference on Industrial Electronics and Applications. 2009:3653-3657.

[10] CHEN Xi-ming, ZHOU Ping. Tracking control and synchronization for two-dimension discrete chaotic systems[J]. The Journal of China Universities of Posts and Telecommunications,2002,9(3): 7-10.

[11] 郑凡,田小建,范文华,等. 基于 Henon 映射的数字图像加密[J]. 北京邮电大学学报,2008,31(1):66-70.

[12] 胡学刚,王月. 基于复合混沌系统的图像加密新算法[J]. 计算机应用,2010,30(5):1209-1211.

[13] 刘云,郑永爱. 基于混沌系统的彩色图像加密新方案[J]. 计算机工程与应用,2011,47(3):90-93.

(上接第 307 页)

4.3　认证强度的保证

在 CBAC 访问过程中要通过账号和口令认证,其中口令是不保存的,这在一定程度上保证了口令的安全。对于共享的文件只能靠身份认证保证安全,因此账号和密码的安全性要用 PAM 模块进行加强。例如,账户名称和密码的设置要符合一定的安全复杂度,而且不能被其他用户甚至 root 用户读取。主体之间的标志只有 UID 是透明的。

5　结束语

本文分析了当前 Linux 下的主要访问控制机制的特点,根据密钥认证和安全分级策略设计了一种文件访问控制机制。这种机制的主要优点是主客体只需关联几个密钥就可以实现灵活的安全访问,其配置和实现都相对容易,实现了较细粒度的访问控制而且给用户一定的自主选择权。但是,文件属主需要进行大量的访问设置和口令输入,这在一定程度上给实际应用造成了不便。此外,在远程用户数量较多的情况下,如分布式网络密钥应用[13]情况下密钥共享也是一项比较复杂的任务。下一步的研究重点是如何提高可用性,如将密钥导入 USB 设备,避免用户多次重复输入密码,从而实现快速访问,以及采用分类安全思想进一步简化安全设置等。

参考文献:

[1] 毕晓普. 计算机安全学[M]. 王立斌,黄征,等译. 北京:电子工业出版社,2005:84-101.

[2] 倪继利. Linux 安全体系分析与编程[M]. 北京:电子工业出版社,2007.

[3] 石文昌. 安全操作系统的发展[J]. 计算机科学,2002,29(6):1-10.

[4] MALLEY S, VANCE C, SALAMON W. Implementing SELinux as a Linux security module, NAI LabsReport JHJ01-043S[R]. 2002.

[5] 赵亮. Linux 安全模块(LSM)简介[EB/OL]. (2003-07-26)[2011-05-10]. http://www.ibm.com/developerworks/cn/linux/l-lsm/part1/.

[6] HARRINGTON A, JENSEN C. Cryptographic access control in a distributed file system[C]//Proc of the 8th ACM Symposiumon Access Control Models and Technologies. New York: ACM, 2003:158-165.

[7] AL-HAMDANI W A. Cryptography based access control in healthcare Web systems[C]//Proc of Information Security Curriculum Development Conference. NewYork: ACM, 2010.

[8] KUMAR A, CHOPDEKAR S. Get started with the Linux key retention service[EB/OL]. (2007-04-11)[2011-05-10]. http://www.ibm.com/developerworks/linux library/l-key-retention/index.html.

[9] SANDHU R S, COYNEK E J, FEINSTEIN H L. *et al*. Role-based-access-control models [J]. Computer,1996,29(2):38-47.

[10] HALLYN S E, KEARNS P. Domain and type enforcement for Linux [C]//Proc of the 4th Aunual Linux Showcase & Conference. Berkeley: USENIX Associafion, 2005.

[11] 袁春,钟玉琢,张基宏,等. 基于密码学的访问控制和加密安全数据库[J]. 电子学报,2006,34(11):2043-2046.

[12] ANDREW G M, THORSTEN K. The Linux-PAM application developers guide[EB/OL]. (2010-08)[2011-05-10]. http://www.kernel.org/pub/linux/libs/pam/linux-PAM-html/linux-PAM-SAG.html.

[13] CHAN H, PERRIG A, SONG D. Random key pre-distribution schemes for sensor networks[C]//Proc of IEEE Symposium on Security and Privacy. 2003:197-213.