Computer Science and Application 计算机科学与应用, 2012, 2, 266-270
http://dx.doi.org/10.12677/csa.2012.25047 Published Online December 2012 (http://www.hanspub.org/journal/csa.html)

An Information Encryption System Based on Block Cipher and Ukey Technology

Juntao Zhang[1], Zhaoyun Chen[2], Zuobing Tian[1], Shaojing Fu[2*]

[1]College of Opto-Electric Science and Engineering, National University of Defense Technology, Changsha
[2]College of Computer, National University of Defense Technology, Changsha
Email: *fu_math@qq.com

Received: Jun. 3rd, 2012; revised: Jul. 11th, 2012; accepted: Aug. 8th, 2012

Abstract: To deal with file security problems, this paper made a comprehensive utilization of identity authentication, file encryption, information hiding, Ukey storage control and other critical safety technologies to develop a safe and easy using information encryption system based on the unique block cipher & file segmentation mode. Based on improving and optimizing the existing encryption algorithms, our system has achieved a result of safe and efficient file encryption and decryption with the use of self-developed encryption module while supporting file safety erasing and different encryption level, which effectively compensates for the deficiencies of existing encryption system.

Keywords: Identity Authentication; Block Cipher Algorithm; File Segmentation; Ukey

一种基于 Ukey 的分组加密信息保护系统

张钧陶[1]，陈照云[2]，田作兵[1]，付绍静[2*]

[1]国防科技大学光电科学与工程学院，长沙
[2]国防科技大学计算机学院，长沙
Email: *fu_math@qq.com

收稿日期：2012 年 6 月 3 日；修回日期：2012 年 7 月 11 日；录用日期：2012 年 8 月 8 日

摘　要：针对文件安全问题，本文综合利用了身份认证、文件加密、信息隐藏、Ukey 存储控制等安全关键技术，并基于独特的“分组加密 + 文件分割”工作模式，开发一套安全易用的信息加密系统。系统在改进和优化已有加密算法的基础上，提出了自主开发的分组加密模块，实现了文件的高效安全加解密，可支持加密原文件深度擦除和不同用户的不同的加密级别需求，有效弥补了目前已有加密系统的不足。

关键词：身份认证；分组加密；文件分割；Ukey

1. 引言

当前，信息产业存在的严重的安全问题，开发安全有效的加密系统和用户登录系统已经迫在眉睫[1-8]。目前信息保护手段的表现并不尽如人意。现有的信息加密手段主要有加密和隐藏两种。其中基于隐藏的信息保护软件处理后的文件隐蔽性高，但只要熟悉其工作原理就可轻易地破解；基于密码算法加密的软件提供了较高的安全性，但其安全性主要靠加密算法的强度和密钥来保证，一旦加密算法被破解或密钥泄露，这种方式就形同虚设，同时效率低下也是这种方式的一大痼疾。

1) 常见的加密软件大部分并不采用加密算法，只是用的是一种在磁盘上建立个特殊文件夹，然后把文件或文件夹转移到这个文件夹里(例如 E 钻文件夹加密大师，E 神文件夹加密，高强度文件夹加密大师，文件加锁王，文件夹加锁王，超级特工秘密文件夹等)。

*通讯作者。

比如只是把文件夹里的文件全部转移到当前文件夹所在分区的根目录下的回收站(RECYCLER)里一般此文件夹为系统隐藏的要找到被加密的文件只需要利用一款常用的文件管理软件 Total Commander 就可以轻松找到被“加密”的文件了。

2) 有一些加密软件采用了加密算法，这些软件的加密强度依赖于加密算法，例如的常见的加密算法有 DES，已被成功破解。加密算法被破解掉后，加密文件便可轻松得到。

综上所述，虽然市场上已有的许多文件保护和用户信息保护系统一定程度上缓解了信息安全产业的一定压力。但没有从根本上解决存在的问题，开发一套更安全、更有效、更易用的安全保密系统具有深刻的现实意义。其具体的设计要求如下：安全性。安全性的重要性对于信息保护系统来说不言而喻，这是用户的第一需求，也是设计者首先考虑的问题。高效性。高效性与安全性在某种程度上是一对矛盾，对于安全性高的安全方案，很难避免加密方案复杂，加密过程耗时相对较大，如何平衡这一矛盾，在保证足够的安全性的同时，加密过程尽可能缩短。易用性。信息保护系统在设计上还需要充分考虑系统的易用性，尽可能向上层屏蔽加密服务的细节以及复杂的安全解决方案，使得系统对于使用者来说简单易用。

2. 系统开发的关键技术以及解决方案

2.1. 自主“分组加密”的算法的设计

由于破坏了密文相关性，使得分组加密后的密文是极难破解的，我们认为保证同样的密文扩散前提下，降低轮数，加快速度，很有意义。针对这一需求，我们自主开发了经过严格测试的分组加密算法——XT-Serpent 算法。我们的系统使用密钥长度为 128 比特位的 TEA 算法以及加密数据块和密钥长度为 256 比特位的改进 AES 算法。TEA 算法的加密强度中等，加密、解密速度较快；改进 AES 算法的加密强度很高，加密、解密速度一般。TEA 算法适用于文件大小比较大，需要高速加密、解密的文件；改进 AES 算法适用于文件大小较小且极为重要的文件。所以，秘密用 16 次迭代的 TEA 算法和机密用 32 次迭代的 TEA，绝密级文件用改进 AES 算法。

S 盒作为 AES 算法惟一的非线性运算，直接决定算法的性能。AES 算法初始 S 盒的仿射变换如下：

$$\begin{bmatrix} y_7 \\ y_6 \\ y_5 \\ y_4 \\ y_3 \\ y_2 \\ y_1 \\ y_0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

该 S 盒的仿射变换周期为 4，迭代输出周期不大于 88，且代数表达式只有 9 项。为了提高 S 盒的安全性，提出了改进方案。改进后 S 盒的仿射变换如下：

$$\begin{bmatrix} y_7 \\ y_6 \\ y_5 \\ y_4 \\ y_3 \\ y_2 \\ y_1 \\ y_0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

改进后的 S 盒仿射变换具有周期为 16，迭代输出周期为 228，而且 S 盒代数表达式项数大于 240 项，这表明改进 S 盒具有具有复杂的代数结构和良好的非线性特性，抗代数攻击的能力更强。

2.2. “文件分割”技术加强安全性

我们将加密文件分割，并将分割后的文件的路径以及小部分文件存放在硬件系统，部分文件可以利用 NTFS 文件流隐藏，部分隐藏于 exe、jpg、或者 bmp 中，其余简单修改文件属性随机分散于硬盘中，以上手段使获得密文难度增大。相应的，软件实现了分割文件的合并功能。

2.3. Ukey 管理密文和密钥模块设计

Ukey 里面存储一个不可读出的固化密钥以及硬盘内隐藏文件的索引位置。对于不同的文件，有个对应的随机密钥，所以 Ukey 里面存储随机密钥，需要加密或解密时，Ukey 里面算法将通过固化密钥和随机密钥运算后得到加密密钥，在低密级时返回给电脑进行加密或解密；高密级时，由 UKey 自行完成加解密。

同时 Ukey 里面存储加密后分割文件的一个小部分，实现部分密文物理隔离，由于分组加密的特性，缺失部分的密文极难破解。

2.4. Ukey 内信息的保护手段

Ukey 内保存着的信息对于本系统来说都是非常重要的。隐藏后文件位置索引、加密算法、密钥，以及部分密文均保存在 Ukey 中。因此 Ukey 内信息的保护是非常重要的，我们充分借鉴了现有的 Ukey 内信息的保护手段，例如：利用数据交换随机噪声技术、迷宫技术对抗逻辑分析仪以及软件在并口监测。此外，我们采用设置时间闸、抑制跟踪中断、封锁键盘输入，优化 Ukey 读写结构等方法，进一步提升跟踪难度以及对抗各种调试跟踪工具的攻击。从而提升系统整体安全性。

2.5. 加密原文件深度擦除技术

如果原文没有完全删除，不法分子可以通过技术手段恢复出部分甚至全部文件，从而绕开我们的文件保护系统，窃取到文件。因此，彻底删除加密原文件是很有必要的。目前市面上的数据销毁软件大都是暴力的将整个磁盘的空间都擦写一遍，或者将磁盘的空余空间擦写一遍，以确保没有已经被删除的信息数据被泄露，本系统将提供硬盘物理位置区域的擦除，直接读取 NTFS 磁盘的原始数据，自主解析 NTFS 的结构，形成文件系统的目录树，定位文件数据存储在磁盘上的具体区域，并对该区域进行多次擦写，保证文件一经擦除将无法被 EasyRecovery 等数据恢复软件恢复。

3. 系统实现

本信息保护系统由软件部分和硬件部分组成，如图 1 所示。

3.1. 软件部分的开发

本系统的软件部分由 visual studio2008 开发，主要实现了实现从 Ukey 中提取密钥，文件加密以及加密文件分割与隐藏，密文的管理，提供多种加密模式以及为用户提供良好的人机交互界面。

在本系统中，我们将加密文件由重要性由低至高

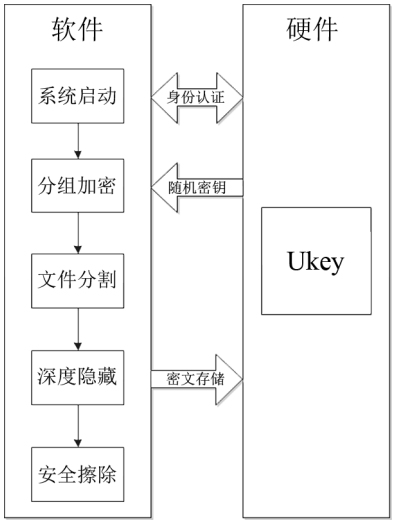

Figure 1. The overall frame figure of system
图 1. 系统总体框架图

分为秘密、机密、绝密三级，针对不同那个的安全需求，我们给出了不同的安全策略，密级低我们就侧重于加解密速度，减少加密的手段或者将部分硬件任务由软件承担。对应的，密级高我们就侧重于安全性。由于加密策略高级包含低级，故这里对秘密和机密级加密方法不在赘述。

加密文件以 Ukey 加密后将密文返回计算机，并进行分割处理，获取文件大小并计算文件分割偏移量。其中一部分利用 jpg 和 exe 将需隐藏的文件写入上述文件，这种方法不会破坏原有的数据，而且隐蔽性较强。并且不影响原来 jpg 的打开以及 exe 的使用。

同时还将文件的一部分字符流添加至 USB 设备中创建的.USBDT 中并隐藏属性。最后将文件分割的偏移量数值，不同部分插入的文件路径，以及加密文件的一部分字符流共同添加至原加密文件路径下的.index 中并隐藏属性，作为还原文件的依据，加密完毕的同时将源文件彻底删除。

解密原理与加密操作相反，根据加密产生的.index 文件读取文件分割的偏移量以及相关隐藏文件(.index 和.USBDT)的路径，并从不同文件尾部读出一定偏移量的数据并进行 AES 算法解密，恢复数据后拼接成源文件，同时彻底删除所有相关隐藏文件(.index 和.USBDT)。

3.2. 硬件部分的开发

UKey 的设计小巧精致、携带方便 UKey 自身所

具备的存贮器可以用来存储一些个人信息或证书等，UKey 的内部密码算法可以为数据传输提供安全的管道，UKey 是适用于单机或网络应用的安全防护产品。Ukey 里面存储一个不可读出的固化密钥。对于不同的文件，有个对应的随机密钥，所以 Ukey 里面存储随机密钥，需要加密或解密时，Ukey 里面算法将通过固化密钥和随机密钥运算后得到加密密钥返回给电脑进行加密或解密；这样一来，加解密在电脑上进行。或者由 Ukey 独立完成加解密。

4. 系统测试与特色分析

4.1. XT-Serpent 分组加密保证安全

XT-Serpent 分组加密模块有良好的混淆和扩散性质，差分分析、线性分析、代数攻击、相关攻击效果较好。本系统将文件进行 XT-Serpent 分组加密后再进行的分割操作，使得分割后的文件完全没有显性意义，极大程度上避免了获得部分密文从而获得部分信息的可能性，加之我们通过多种手段隐藏密文，令破解难度大幅升高。利用分组加密密文扩散的特性，缺失信息的密文将变得极难破解。XT-Serpent 模块是我们首次提出，非法分子无现成破解方法可用，研究本模块的各种加密算法的破解方法也需要大量人力物力。这充分保证了文件的安全性。以上手段保证了我们的加密方式简单高效，实际测试效果良好。但和传统方法一样，面临一样的问题，如果整套加密方法被非法分子知晓，且使用简单有规律的口令、密钥或者泄露了口令、密钥，一切保护 就全部被摧毁。

4.2. 利用 Ukey 保证信息安全

UKey 的硬件是由带有 EPROM 的 CPU 实现的芯片级操作系统，所有读写和加密运算都在芯片内部完成，具有很高的安全度。与通用磁盘介质相比，内置芯片级操作系统，防止被非法复制，保证数据的唯一性，UKey 内数据只在 UKey 内留存，有利于在公共场所使用；与软盘相比，耐用性大大提高；同 IC 卡相比，由于不需要专用的读卡设备，在与电子商务以及各种以 PC 为基础的安全应用上具有其它产品不可替代的优越性。安全性：UKey 提供了比传统口令验证更加安全且更易于使用的网络用户身份认证机制。UKey 使用共享秘密的方式实现网络客户与服务器之间的身份验证，不用暴露任何关键信息就可以实现用户身份的验证。而且 UKey 内置的用户访问控制可以进一步增强验证过程的安全性。UKey 认证系统里面的组件 SecureFile 功能组件提供基于文件的加密，解密等功能。通过 Ukey 内置芯片的运算来进行加密，最大程度上保证了文件在加密过程中和加密的安全性；在我们的系统中，对于不同的文件，有个对应的随机密钥，所以 Ukey 里面存储随机密钥，需要加密或解密时，Ukey 里面算法将通过固化密钥和随机密钥运算后得到加密密钥可以返回给电脑进行加密或解密；这样一来，加解密可以在电脑上进行，在低密级模式下提高加解密速度。采用一系列 UKey 内部信息保护技术来保护 UKey 内信息。

4.3. 安全手段综合利用

将上述两种思想结合的意义在于 1) 加密过程黑箱化，即加密算法以及密钥全部写入硬件系统，文件加密解密过程计算机不参与，且由于烧录入硬件系统内的程序无法被反编译，从而杜绝计算机病毒木马截获信息，加之用户也并不知晓密钥，即便我们将来我们将 XT-Serpent 模块完全公开，非法分子也由于无法获得密钥，只能暴力破解，但耗时难以接受。2) 部分文件物理隔离，存于硬件系统。由于分组加密的特性，缺失部分的密文极难破解，Ukey 体积较小，用户保存起来相对容易且安全，非法分子很难获得，且一旦丢失，我们可以第一时间发现，并紧急摧毁加密文件，使得文件不会外流。

5. 结束语

系统以完善的方案有效地解决了当前信息保护所面临的问题，将为很多需要高安全性加密系统的用户提供一种有效的文件保护手段，能根据用户不同的需要来提供不同的加密级别，同时也为文件在网络传输过程中的安全提供了一种保护的方法。该加密系统中的“分组加密 + 文件分割”思想具有其独创性，其高度相关性特点，将成为破解者的最大障碍，保障了文件的安全性。

6. 致谢

本文承国家自然科学基金(No：61103191)和全国

大学生创业创新训练计划项目资助。

参考文献 (References)

[1] 胡俊, 刘毅. 设计和实现基于UsbKey的透明加解密文件系统[J]. 计算机科学, 2008, 11: 23-29.

[2] 何希平, 朱庆生. 基于混沌映射的 Hash 函数及其在身份标识认证中的应用[J]. 计算机应用, 2006, 26(5): 1058-1060.

[3] M. Bellarea, J. Kilianb and P. Rogawayc. The security of the cipher block chaining message authentication code. Pattern Recognition Letters, 2005, 26(15): 2400-2408.

[4] 程庭, 张明慧, 石国营. 一种基于 DES 和 RSA 算法的数据加密方案及实现[J]. 河南教育学院学报: 自然科学版, 2003, 12(2): 69-71.

[5] 师军, 张福泰, 王耀燕. 高级加密标准 Rijndel 算法中的 S 盒及其实现[J]. 小型微型计算机系统, 2003, 24(7): 1207-1209.

[6] 邢书宝, 李刚, 薛惠锋. 一次一密加密系统设计与实现[J]. 计算机技术与发展, 2007, 17(3): 150-155.

[7] 邵昱, 萧蕴诗. 基于文件系统过滤驱动器的加密软件设计[J]. 计算机应用, 2005, 5: 1151-1152.

[8] 米新家, 张开来, 曹卫兵, 苗胜, 戴冠中. 基于 FPGA 芯片的硬盘数据加密设计与实现[J]. 西北工业大学学报, 2004, 22(2): 12-17.

