

基于 USBKEY 的网络存储用户数据保护的研究与实现

喻潇¹, 田里¹, 刘喆^{2,3}, 王捷¹

(1. 国网湖北省电力有限公司电力科学研究院, 湖北 武汉 430077;

2. 武汉大学国家网络安全学院, 湖北 武汉 430072;

3. 武汉大学空天信息安全与可信计算教育部重点实验室, 湖北 武汉 430072)

摘要: 网络硬盘和云存储等网络存储服务运用面广泛, 但是由于网络硬盘与云存储自身的相对灵活性, 造成其需要面对诸多安全风险。提出基于 USBKEY 的数据加密防护机制, 数据加密与存储服务独立, 把密钥存储于 USBKEY 内, 最终将“服务层面提供安全”细化到“实体掌握安全”, 达到用户掌控数据安全的目的。

关键词: USBKEY; 云存储; 网络硬盘; 数据隐私保护

中图分类号: TP393

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2018018

Research and realization on USBKEY based network storage user data protection

YU Xiao¹, TIAN Li¹, LIU Zhe^{2,3}, WANG Jie¹

1. State Grid Hubei Electric Power Research Institute, Wuhan 430077, China

2. School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China

3. Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education, Wuhan University, Wuhan 430072, China

Abstract: Network storage services such as network hard disk and cloud storage are widely used. However, due to the relative flexibility of network hard disk and cloud storage, it has to face many security risks. Based on the data encryption protection mechanism of USBKEY, data encryption and storage service, the key is stored in USBKEY, will eventually “service level provide security” elaboration to the “real master security” to achieve the purpose of user control of data security.

Key words: USBKEY, cloud storage, online storage, data privacy protection

1 引言

随着互联网与计算机的发展, IT 技术的不断演进, 用户数据呈爆炸式增长, 越来越多的用户依靠网络等渠道进行数据的处理、传递和存储, 网络硬盘、云存储等用户数据的服务产品能够向用户提供数据、文件等的存储、备份、共享和管

理服务, 具有方便快捷的优点。但是, 网络硬盘与云存储等服务只是提供了存储空间, 并没有承诺对存储数据的隐私保护, 安全问题也越来越突出, GoogleDocs、The Linkup 等多家云服务提供商都曾被曝出由于安全问题导致数据泄露等严重后果^[1]。具体来说, 一方面来自外部威胁与框架安全问题, 通过云服务器用户接口与云内部系统

收稿日期: 2018-04-25; 修回日期: 2018-05-20

通信作者: 刘喆, liangshijie@sina.com

基金项目: 国家自然科学基金资助项目 (No.61772384)

Foundation Item: The National Natural Science Foundation of China (No.61772384)

进行交互, 外部攻击者利用软硬件的漏洞实施入侵, 这主要是系统在开发过程中就存在安全隐患, 目前针对安全框架已有相当多的研究, 云计算的安全问题也已经是一大热门话题^[2]; 另一方面是来自内部威胁与用户信任的问题, 目前阻碍网盘和云存储等服务的重要制约因素就是用户对服务提供商的信任问题^[3], 云存储的外包存储服务模式导致了特权用户的存在, 后者具有非授权访问用户数据的能力, 易导致数据信息和隐私泄露等内部攻击问题^[4], 内部管理员或者特权用户的误操作、恶意企图都将造成用户数据发生泄露, 而系统本身的一些操作行为也会侵犯用户的隐私。用户对网络硬盘和云存储之类的存储服务不够信任, 同时存储服务提供商也无法证明自己绝对的安全可信, 所以用户的敏感数据和文件不敢随便上传到服务端。为了实现数据的保护, 对数据进行加密是最基本的方法, 但一般的加密处理方式往往较为简单, 而且缺乏完整的密钥管理体系, 不涉及较安全的加密算法。

目前, 在数据加密方面已有框架安全、加密算法等诸多研究^[5], 但在客户端进行加密的研究相对较少。在客户端进行加密涉及很多问题, 考虑到服务使用者大部分是非专业人员, 因此在加密过程和密钥管理上存在较多问题, 本文提出了一个用户自主保护数据的安全方案, 以 USBKEY 为密钥载体, 建立完整的密钥管理体系。该体系将前序安全体系中的“服务层面提供安全”细化到“实体掌握安全”, 以 USBKEY 为基础, 实现安全快捷的加解密服务。

2 基于 USBKEY 的数据保护方案

由于网络硬盘以及云存储等服务功能越来越强大, 可以实现检索、运算等功能。在本方案中, 主要考虑存储和备份服务, 不考虑其他(如计算、检索等)相关功能。数据仅在客户端以明文存在。

2.1 数据加密

数据加密是对数据进行保护的基础, 在用户使用存储服务的过程中, 在哪一个环节对数据进行加密尤为重要。目前很多存储服务提供商宣称对存储空间进行加密, 使数据能够在服务提供商端实现密文存储, 但依然存在安全风险, 一方面数据在传输

的过程中是明文存在, 容易被截获^[6]; 另一方面, 用户对自己的数据安全仍然无法感知^[7], 因此数据的加密还必须由用户完成。由于用户对服务提供商存在信任问题, 若要以用户可感知的方法解决数据保护问题, 则必须由用户完成数据加密工作, 同时, 为了抵御服务提供商内部的恶意攻击, 推荐方式为仅使用网络硬盘和云存储所提供的存储服务, 而加密的细节对存储服务提供商不可见, 这就要求在用户端具有一套自主的数据加解密体系, 同时又能达到对存储服务提供商相对独立的目的^[8-9]。

2.2 USBKEY

USBKEY 是一种 USB 接口的硬件设备, 内置了 CPU、存储器、芯片操作系统(COS, chip operating system), 具有安全数据存储空间, 可以存储数字证书、密钥等秘密数据, 对该存储空间的读写操作必须通过程序实现, 用户无法直接读取, 其中的用户密钥是不可导出的, 这就杜绝了复制用户数字证书或身份信息的可能性, USB-KEY 可以实现加解密和签名的各种算法, 加解密运算在 USBKEY 内进行, 保证了密钥不会出现在计算机内存中, 从而杜绝了用户密钥被黑客截取的可能性^[8]。利用 USBKEY 内置的密码算法还能够对用户身份的认证, 每一个 USBKEY 都具有硬件 PIN 码保护, PIN 码和硬件构成了用户使用 USBKEY 的 2 个必要因素, 用户只有同时取得了 USBKEY 和用户 PIN 码才可以登录系统。如果 PIN 码泄露, 只要用户仍持有 USBKEY, 用户的身份就不会被仿冒; 如果用户的 USBKEY 遗失, 但由于不知道用户 PIN 码, 也无法仿冒用户的合法身份^[10]。

要实现用户自主数据加密且相对存储服务提供商独立, 需要面对的主要问题是: 加密算法的实现和密钥的管理。加密操作必须采用相应的软件来实现, 而密钥管理则更为复杂。从以上 USBKEY 的特点分析来看, USBKEY 非常适合作为加解密与密钥管理的载体^[11-12]。

2.3 基于 USBKEY 的数据加密

用户对敏感数据先加密后上传, 由于上传的数据是密文, 能够同时抵抗针对服务商的外部攻击和内部攻击, 当用户自行加密数据时, 最大问

题就是加密软件和密钥管理。采用 USBKEY 的形式能保证用户的密钥安全,并且使用户能方便地在多种环境下实现随身加解密。

2.4 用户自主掌控数据安全

用户个人掌控安全,意味着仅用户能获得最终加解密文件的密钥,用户也能够独自掌握 USBKEY 的 PIN 码。在本方案中,最终加解密文件的密钥由 USBKEY 内置密钥和 PIN 码生成的密钥组合而成,最终密钥只在运行时临时生成。首先,用户 PIN 码生成一个密钥,并作为最终密钥的一部分,使密钥完全由用户掌控,即使 USBKEY 内密钥意外泄露,也无法得到最终密钥,管理服务端也无法获得最终密钥,使安全完全由用户自己掌握,用户信息的隐私得到完全保护;其次,USBKEY 内部有标准密钥,该密钥存放在安全存储区域中,只作为加解密密钥的一部分,内置密钥进一步加强了密钥的安全性,也解决了 PIN 码产生的密钥空间过小的问题。

3 PIN 码的保护

3.1 PIN 码

在本方案中,USBKEY 提供基本 PIN 码管理、认证等功能,并在此基础上扩充了 PIN 码的功能,以及对历史 PIN 码的管理。

3.2 身份认证

为防止 USBKEY 被盗用,每次使用 USBKEY 时需验证用户身份,验证方式采用 PIN 码,验证通过后方可使用 USBKEY 的功能,具体认证流程图 1 所示。

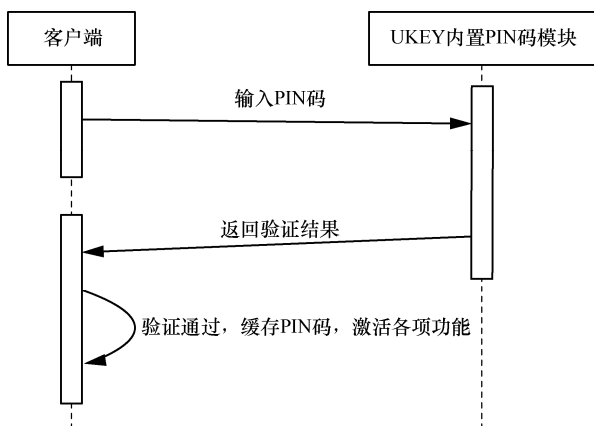


图1 身份认证流程

3.3 密钥生成

为了达到个人掌控安全的目的,最终的加解密密钥由 USBKEY 内置的标准密钥和用户口令共同决定。其中, PIN 码生成的密钥是实现用户掌控安全、保护用户信息隐私的重要手段,内置密钥则为了达到增大密钥空间、加大密钥强度。密钥生成流程如图 2 所示。

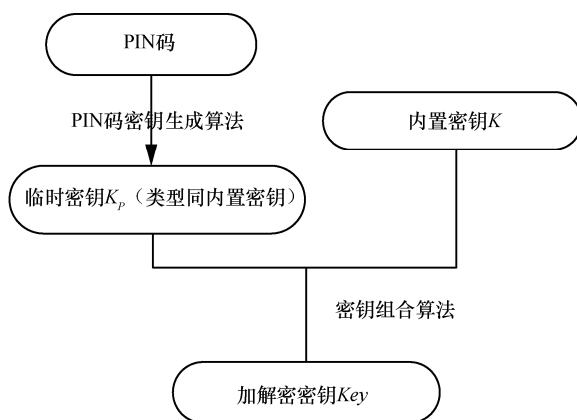


图2 密钥生成流程

3.4 PIN 码修改与管理

3.4.1 PIN 码与密钥更新

为了安全使用 USBKEY,用户可随时修改 USBKEY 的使用 PIN 码,同时,修改 PIN 码也会同步更新加解密密钥,修改操作全部在客户端进行,认证 PIN 码后才能进行修改,修改流程与修改口令类似。

3.4.2 PIN 码标识数

当用户修改 PIN 码时,加解密文件的密钥也会改变,用户能对之前所有文档进行解密操作,就需要对 PIN 码进行管理。为了实现对 PIN 码的管理,会对每个 PIN 码定义一个标识数,标识数的规则如下。

1) PIN 码的标识数产生于该 PIN 码被修改时,值为修改时系统时间距离 1970 年 1 月 1 日 0:0:0 的秒数。

2) PIN 码被修改后,会被新 PIN 码所产生的密钥加密,该 PIN 码的标识数和密文以 XML 格式存储于 USBKEY 安全区域的 PIN 码维护文件中。

3) 加密文件时,将在文件的密文文件的额外信息区域加入时间数信息,即当前加密时间距离

1970年1月1日 0:0:0 的秒数。

4) 解密时, 根据密文文件时间数信息和 PIN 码维护文件确定加密该文件的 PIN 码, 取得该 PIN 码密文后解密并用该 PIN 码解密文件(用 PIN 码加密文件意为用该 PIN 码生成的密钥和内置密钥组合而成的密钥加密, 解密同)。

5) 修改 PIN 码时, 需用修改前的 PIN 码解密所有 PIN 码, 并用新 PIN 码加密所有历史 PIN 码并重新存储, 即每次修改 PIN 码都需重新更新整个文件, 相应的标识数保持不变。

3.4.3 PIN 码管理

PIN 码是由标识数管理, 对历史 PIN 码管理规则如下。

1) PIN 码维护文件存放在 USBKEY 安全区域指定目录下, 客户端通过身份认证后才能进行操作。

2) PIN 码维护文件以 XML 格式存储, 每个 PIN 码包含两个字段: 标识数与密文。

3) 每次修改 PIN 码时: 首先用修改后的 PIN 码 P_{new} 加密修改前的 PIN 码 P_{old} , 得到密文 $*P$, 同时生成标识数 NUM ; 再将 NUM 和 $*P$ 组合成

PIN 码维护文件规定的格式, 并添加未修改完成标记, 将以上内容添加进 PIN 码维护文件中; 调用 PIN 码管理模块修改 PIN 码(此 3 步完成后才开始进行之前的 PIN 码维护); 取出 PIN 码维护文件中不带未完成属性(刚加入的节点)的 PIN 码节点的密文, 用 P_{old} 解密、 P_{new} 加密, 并更新对应密文节点的内容; 更新完毕后, 删除未修改完成属性, 修改操作全部结束。修改 PIN 码的流程如图 3 所示。

需要说明的一点是, 由于 PIN 的目标是用户个人掌握安全, 所以 PIN 码仅用户个人知晓。所以针对 PIN 码仅提供修改功能, 不提供管理功能。一旦 PIN 丢失, 无法做到密码找回等管理操作。

3.5 PIN 码生成密钥算法

输入 PIN 码 P 、内置密钥 KU , 输出为 64 位 DES 密钥 KEY , KEY 生成过程如下。

1) 初始化 Hash 函数, 将 P 作为 Hash 函数的输入。

2) 取 Hash 结果并计算校验码, 凑成 64 位 DES 密钥 KP 。

3) 计算 KP 与 KU 异或结果 K 。

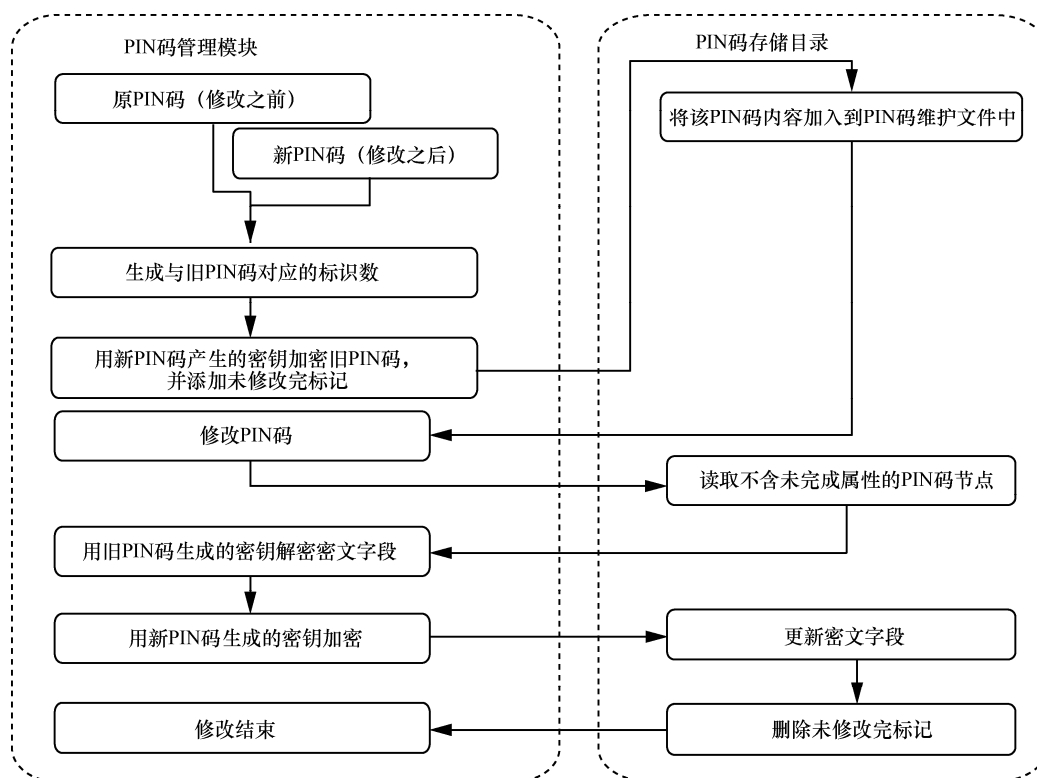


图3 PIN 码修改流程

4) 根据 KP 与 KU 的长度, 以 K 作为输入, 调用对应的密钥生成算法得到组合后的密钥 KEY 。

4 基于 USBKEY 的用户自主数据安全防护方案

在整个安全防护方案中, 加解密分为用户端与服务端两部分。其中, 用户端由 USBKEY 组成, 负责用户数据的加解密以及集成网盘接口, 服务端则负责密钥的管理。

USBKEY 硬件提供一定的存储空间和加解密功能, 硬件部分对软件提供加解密调用接口, 接收上层软件传来的数据进行加解密操作并返回结果数据; 软件的核心部分是文件加解密包, 负责读取文件调用 USBKEY 进行加解密并输出密/明文文件至指定目录, 整个方案中该部分与硬件进行交互并对外提供接口, 供调用实现文件的加解密操作。

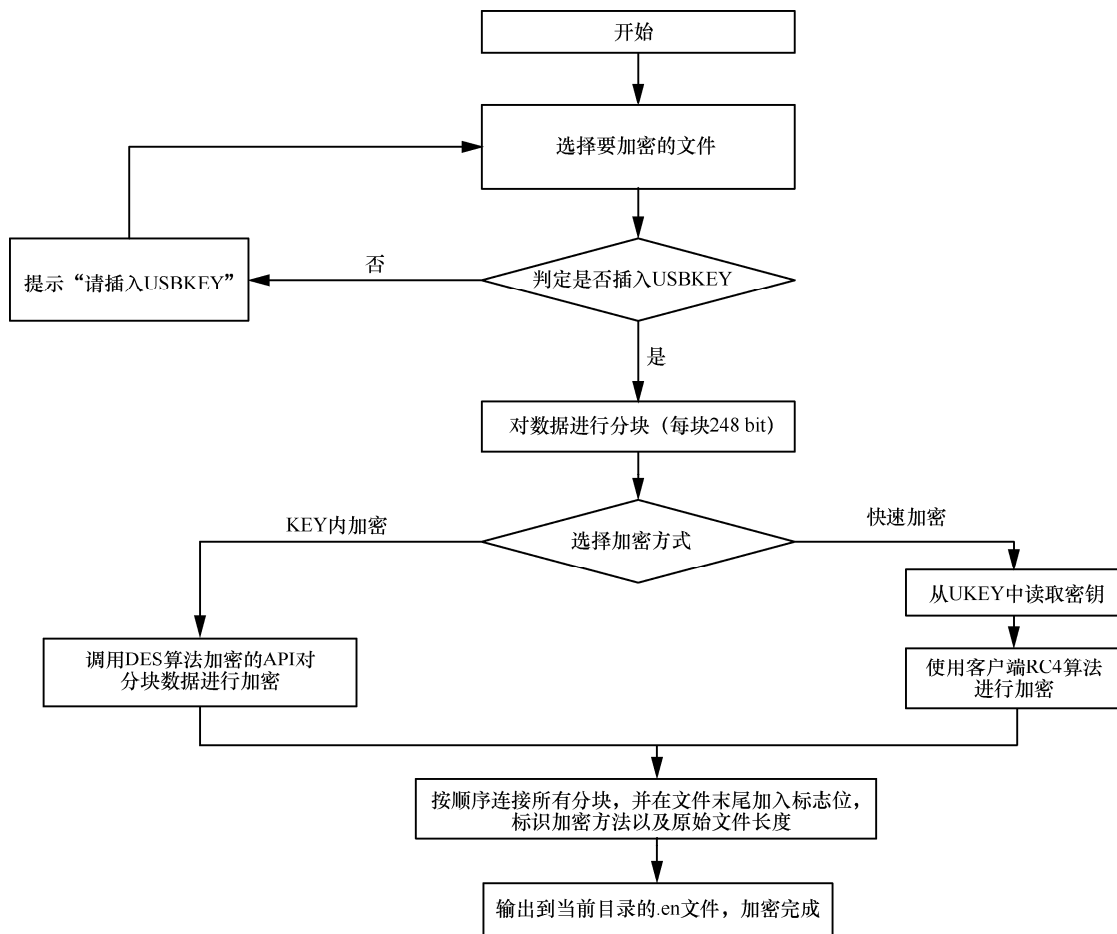
4.1 加密处理

客户端共实现了 2 种密码算法: DES 和 RC4, 其中, USBKEY 加密采用的是 DES 加密算法, 快速加密采用 RC4 算法, 2 种算法的密钥存储于 USBKEY 内的保密区域, 未授权者无法读取。数据源经过分块后根据选择的操作类型 (USBKEY 内部加密或快速加密) 进行相应的加密操作, 2 种方式的区别如下。

1) USBKEY 内部加密的计算由 USBKEY 内的密码芯片完成, 速度较慢。

2) 快速加密的计算由客户端完成, 但密钥需从 USBKEY 读取出来, 具体流程如图 4 所示。

由于存在 2 种加密算法, 解密时要选择相应的解密方式, 具体在加密的过程中将加密方式存入密文, 客户端根据标记字段选择相应的加密方式。由于采用对称算法, 因此密钥的处理方式与加密相同, 快速解密同样需要读出密钥, 整体流程如图 5 所示。



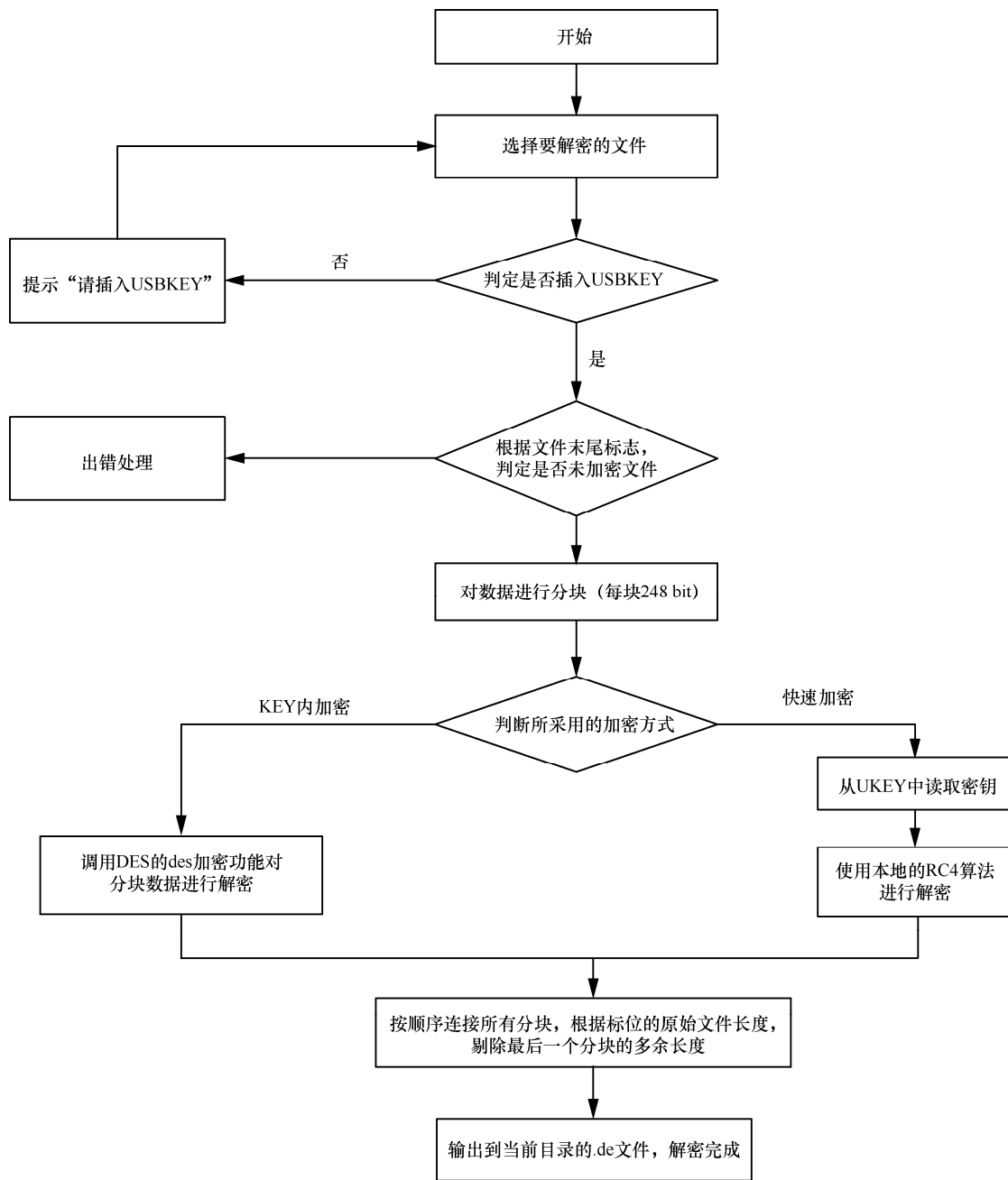


图5 解密流程

4.2 密钥管理

密钥管理由密钥库实现, 密钥库主要用于管理 USBKEY 中的密钥, 具体是有一个独立的非联网数据库以及相关操作程序组成。密钥由非联网服务器的专门程序生成, 生成后写入 USBKEY, 每个 USBKEY 会写入 1 个 DES 密钥和 1 个 RC4 密钥, 密钥写入 USBKEY 后, 会在密钥库中进行备份, 备份数据分组含密钥数据备份与密钥文件备份, 所有备份数据均为密文数据。

密钥数据库中备份表包含以下字段。

- 1) USBKEY_ID: 对应每个 USBKEY 的唯一 DI。
- 2) USBKEY_KEY_DES: DES 加密所用密钥数据 (密文)。
- 3) USBKEY_KEY_RC4: RC4 加密所用密钥 (密文)。
- 4) USBKEY_PATH_DES: DES 密钥密文文件存放目录。

5) USBKEY_PATH_RC4: RC4 密钥密文文件存放目录。

如果用户 USBKEY 遗失, 必须生成相同密钥的 USBKEY 才能确保之前的加密文件能够正常解密, 具体根据遗失 USBKEY 的 ID 查找对应的密钥文件, 解密后写入到新的 USBKEY 中。

4.3 软件版本与安全证明

软件共分为网络版与本地版共 2 个版本, 其中, 本地版主要实现数据的加解密, 密钥由 USBKEY 内部密钥和 PIN 码形成, 仅有文件加解密的基本功能, 不通过网络交换数据; 网络版则在本地版的基础上增加了存储功能, 集成了网络硬盘和云存储所提供的接口^[13]。

目前, 盗取用户信息的软件主要通过预留后门、植入代码、搜集用户信息或未经允许上传私人信息等方式实现。在加解密软件版本中, 本地版不与外界交流任何信息, 而网络版则调用网盘接口, 数据分组发送地址为网络硬盘或云存储服务商的地址, 可以应对以上常见的安全威胁。

5 结果测试

5.1 加密效率测试

通过测试 USBKEY 内部加密和 USBKEY 外加密速度, 发现 2 种加密耗时还有较大差别, 具体如表 1 所示。

表 1 USBKEY 内部加密和 USBKEY 外加密耗时对比

文件大小	KEY 内加密耗时/s	KEY 外加密耗时/s
513 kB	2	50
2 MB	10	312
9.5 MB	58	>900

5.2 软件监控

本地版无网络数据交换, 对于网络版能够实现客户端提交数据的发送地址进行监控, 具体如图 6 所示。

UDP	127.0.0.1	50145	0.0.0.0	0
TCP	192.168.1.102	10682	61.184.100.231	80

图 6 客户端网络连接

6 结束语

网络硬盘以及云存储等服务为用户提供了诸

多便利, 但其自身存在的安全问题在一定程度上阻碍了发展, 本文提出了一种从客户端着手、以 USBKEY 作为载体, 针对用户数据进行加解密的安全防护方案。该方案改变了以往由服务商提供安全的传统模式, 既适用于本地用户的软件与数据隐私保护, 也针对网络存储系统进行数据安全保障, 并监控软件运行过程, 及时发现异常操作行为, 实现用户自身掌控数据安全, 以用户可感知的方法解决了数据与软件保护的安全挑战, 具有一定的推广应用价值。

参考文献:

- [1] 刘晓建, 王力生, 廖新考. 基于 CP-ABE 和 XACML 多权限安全云存储访问控制方案[J]. 计算机科学, 2016, 43(3): 118-121.
LIU X J, WANG L S, LIAO X K. Multiple permissions secure access control scheme combining CP-ABE and XACML in cloud storage[J]. Computer Science, 2016, 43(3): 118-121.
- [2] 吴吉义, 李文娟, 黄剑平, 等. 移动互联网研究综述[J]. 中国科学: 信息科学, 2015, 45(1): 45-69.
WU J Y, LI W J, HUANG J P, et al. Key techniques for mobile Internet: a survey[J]. Scientia Sinica Informationis, 2015, 45(1): 45-69.
- [3] 牛德华, 马建峰, 马卓, 等. 基于属性的安全增强云存储访问控制方案[J]. 通信学报, 2013, 34(S1): 276-284.
NIU D H, MA J F, MA Z, et al. Enhanced cloud storage access control scheme based on attribute[J]. 2013, 34(S1): 276-284.
- [4] 余江, 万劲波, 张越. 推动中国云计算技术与产业创新发展的战略思考[J]. 中国科学院院刊, 2015, 30(2): 181-186.
YU J, WAN J B, ZHANG Y. Strategy of promoting innovation in China's cloud computing industry[J]. Bulletin of Chinese Academy of Sciences, 2015, 30(2): 181-186.
- [5] 刘帆, 杨明. 一种用于云存储的密文策略属性基加密方案[J]. 计算机应用研究, 2012, 29(4): 1452-1456.
LIU F, YANG M. Ciphertext policy attribute based encryption scheme for cloud storage[J]. Application Research of Computers, 2012, 29(4): 1452-1456.
- [6] 王帅, 常朝稳, 魏彦芬. 基于云计算的 USB Key 身份认证方案[J]. 计算机应用研究, 2014, 31(7): 2130-2134.
WANG S, CHANG C W, WEI Y F. USB key authentication scheme based on cloud computing[J]. Application Research of Computers, 2014, 31(7): 2130-2134.
- [7] 冯登国, 秦宇, 汪丹, 等. 可信计算技术研究[J]. 计算机研究与发展, 2011, 48(8): 1332-1349.
FENG D G, QIN Y, WANG D, et al. Research on trusted computing technology[J]. Journal of Computer Research and Development, 2011, 48(8): 1332-1349.
- [8] 秦中元, 胡爱群. 可信计算系统及其研究现状[J]. 计算机工程, 2006, 32(14): 111-113.
QIN Z Y, HU A Q. Trusted computing system and its current computing

er engineering[J]. Computer Engineering, 2006, 32(14): 111-113.

- [9] 张焕国, 陈璐, 张立强. 可信网络连接研究[J]. 计算机学报, 2010, 33(4): 706-717.

ZHANG H G, CHEN L, ZHANG L Q. Research on trusted network connection[J]. Chinese Journal of Computers, 2010, 33(4): 706-717.

- [10] 于江, 苏锦海, 张永福. 基于 USB-Key 的强口令认证方案设计与分析. 计算机应用, 2011, 31(2): 511-513.

YU J, SU J H, ZHANG Y F. Design and analysis of USB-Key based strong password authentication scheme[J]. Journal of Computer Applications, 2011, 31(2): 511-513.

- [11] Trusted Computing Group. TCG trusted network connect TNC architecture for interoperability[S]. 2009.

- [12] Cisco. Cisco Network Admission Control[R]. 2003.

- [13] Microsoft. Network Access Protection Platform Architecture[R]. 2004.

[作者简介]



喻潇 (1984-), 男, 湖北红安人, 国网湖北省电力有限公司电力科学研究院工程师, 主要研究方向为信息安全。



田里 (1984-), 男, 湖北十堰人, 国网湖北省电力有限公司电力科学研究院工程师, 主要研究方向为信息安全。



刘喆 (1989-), 男, 山东青岛人, 武汉大学博士生, 主要研究方向为信息安全。



王捷 (1987-), 男, 湖北黄冈人, 博士, 国网湖北省电力有限公司电力科学研究院工程师, 主要研究方向为信息安全。

《网络与信息安全学报》2018 年第 5 期刊登的《基于非对称直方图修改的可逆信息隐藏方案》一文的项目基金：国家自然科学基金资助项目 (No.59637050) 应为国家自然科学基金资助项目 (No.61502009)，特此说明。