

软件加密与解密技术及其应用

实例专辑

中国科学院成都计算机应用研究所情报室



前 言

随着计算机的广泛应用，高技术不断地相互渗透，人们对软件及数据保护已提到议事日程，为了使计算机应用工作者尽快地掌握熟悉软件加密与解密技术知识，我们曾在1987年编了《微型、小型计算机操作系统应用技术与软件加密技术汇编》一书，受到了广大读者的欢迎，为使软件加密与解密技术资料内容更加丰富、系统、实用性更强，我们又大量搜集了近期最新资料，编制成《软件加密与解密技术及其应用实例专集》一书。该书是对前一套书下册的一个补充，以大量应用实例为主，具有较高的实用价值，是一本难得的内部参考性的情报资料。我们希望它能成为广大计算机应用工作者，获取成果捷径不可少的工具。

该书主要内容有：加密与解密基本原理与技术、方法与实践、文件保护与恢复等。

在资料的汇集中，因时间关系未一一征求作者的意见，有不妥的地方，敬请作者原谅，并对作者积极支持此工作表示衷心感谢。

由于水平有限及时间关系，搜集的资料还不很完全，错误难免，敬请读者指正。

该书在编审的过程中受到了高树清、吴浣尘、罗淳、孟晓玲、刘素琴、徐敏如、张薇薇、梁军、郭俊茹等同志的支持表示感谢。

该书由杨明芳、周永培统编。

编者

88, 7, 20

《软件加密与解密技术及其 应用实例专集》

目 录

原理与技术

| | |
|-----------------------------------|--------|
| 计算机与现代密码技术..... | (1) |
| 数据加密的体制及应用..... | (12) |
| 微型计算机应用系统的安全技术..... | (16) |
| 密钥管理探讨..... | (20) |
| 提高微机数据库信息安全保密性的措施..... | (24) |
| 计算机信息保密方法简述..... | (30) |
| 一些加密技术..... | (32) |
| FA 密码体制在数据完整性保护中的应用..... | (36) |
| 磁盘如何“加锁”？..... | (44) |
| 磁盘文件的简单加密与解密..... | (50) |
| IBM—PC 微机磁盘防拷贝和信息加密技术..... | (58) |
| 微型机磁盘信息保密探讨..... | (62) |
| 谈谈软磁盘加密..... | (67) |
| 怎样判定加密算法的和谐性..... | (70) |
| 数据加密/解密器 (DEU) Intel 8294A..... | (72) |
| 软件的加密..... | (78) |
| 计算机信息安全保密及安全保密通讯处理机..... | (82) |
| 关于运行程序的加密与反加密..... | (89) |

方法与实践

| | |
|---------------------|---------|
| 推荐几种加密方法..... | (93) |
| 软件加密及其实现的一种方法..... | (94) |
| 软件加密保护方法..... | (96) |
| 一个反动态跟踪程序的破译方法..... | (99) |
| 微型计算机的序列加密方法..... | (106) |
| 扇区交错保密法..... | (112) |

| | |
|-----------------------------------|---------|
| 用数码形成密文的方法..... | (114) |
| 加密程序的数据输入方法..... | (115) |
| IBM PC软件的加密..... | (117) |
| APPLE—Ⅱ微机磁盘防止复制的一种方法..... | (124) |
| 也谈APPLE—Ⅱ机程序的加密..... | (127) |
| 也谈APPLE BASIC程序的加密..... | (132) |
| 简单可靠的APPLESOFT程序保密法..... | (133) |
| APPLESOFT程序的加密与解密..... | (134) |
| APPLE Ⅱ应用软件的加密方法..... | (135) |
| COMX—PC机 机上程序加密..... | (140) |
| R1机BASIC程序的加密..... | (141) |
| 浅谈 BASIC程序的加密与解密..... | (142) |
| 用LIST执行DOS命令程序保密又一法..... | (143) |
| DOS3.3下文件的简易加密与解密..... | (146) |
| 修改DOS命令达到加密目的..... | (147) |
| APPLE Ⅱ CP/M软盘加密方法..... | (147) |
| 用BASIC在dBASE Ⅲ程序中加密..... | (151) |
| 一种较为可靠的磁盘加密系统电磁软盘加密系统介绍..... | (152) |
| APPLE—Ⅱ操作系统技术分析与磁盘加密方法..... | (154) |
| APPLE—Ⅱ磁盘加密一法..... | (160) |
| 对汉字码加密的解密..... | (161) |
| 为dBASE Ⅲ增加共享文件加锁功能..... | (162) |
| dBASE Ⅲ数据库软件的加密和解密..... | (165) |
| 计算机网络中的数据加密和密钥分配..... | (166) |
| COPYWRITE的威力..... | (169) |
| 如何去掉PROLOK的保护..... | (175) |
| 如何破解SOFTGUARD 2.00版的保护..... | (179) |
| “坏”软盘起死回生..... | (182) |
| PROLOK激光加密系统分析与解密方法..... | (182) |
| 浅谈加密软磁盘的解密与拷贝..... | (186) |
| 一种加密dBASE—Ⅲ软件的解密过程与方法..... | (188) |
| 对IBM PC/XT上一些游戏程序的解密方法..... | (191) |
| 用2字节的短程序解密..... | (193) |
| 过程对称的制、解密程序..... | (194) |
| APPLE—Ⅱ BASIC程序加密后的一种解密方法..... | (195) |
| 一种解密加“P”BASIC程序的方法..... | (197) |
| “P”BASIC源程序的解密方法..... | (198) |
| 加密BASIC程序的一种简单破译方法..... | (201) |
| 用PASCAL编写加“P”保护的BASIC程序的解密程序..... | (203) |

| | |
|--------------------------------|---------|
| 一种简便的 IBM—PC BASIC 程序的解密 | (206) |
| 用DEBUG解密IBM PC BASIC 程序 | (208) |
| 用IBM—PC BASIC对P 参数文件解密 | (210) |
| APPLE 微机解密小程序 | (211) |
| 自制 BASIC 快速解密程序 | (212) |
| 对 BASIC 程序加密的一种新方法 | (213) |
| 制表软件OFFIC 1.00A 的解密 | (214) |
| 对加密汉字操作系统(9 针小字)的去密 | (215) |
| 五笔字型操作系统探讨 | (217) |
| APPLE—II 超级软汉字系统 DOS 2.0 的一个错误 | (219) |
| 使用27916KEPROM实现软件加密 | (213) |
| 汉字码文件名解密方法 | (221) |

文件保护与恢复

| | |
|----------------------------|---------|
| MICROVMS和 VMS 的文件保护系统 | (225) |
| IBM—PC 软磁盘文件的恢复方法 | (238) |
| IBM PC软磁盘文件恢复技术 | (243) |
| BM—PC 磁盘文件的恢复及文件属性的修改 | (251) |
| UNIX系统中误删文件的恢复 | (259) |
| 文件的属性及其修改应用 | (261) |
| 数据库系统中恢复管理的设计与实现 | (267) |
| 一种文件加、解密方法的实现 | (273) |
| 也谈APPLE程序的保密 | (276) |
| 文件加密与解密 | (278) |
| PC—1500 机程序的恢复 | (280) |
| LASER 机程序中部分程序段的删除方法 | (281) |
| 用 PC—TOOLS 工具盘拷贝加密盘 | (282) |
| 二字节文件解除各种版本 BASIC 的“P” | (283) |
| 恢复内存中(或加密) BASIC 程序的一种简易方法 | (284) |
| IBM—PC/XT 磁盘文件的一种保护方法 | (284) |
| 探讨关于修复dBASE III中数据文件的方法 | (285) |
| APPLE II 机的dBASE II 命令文件保密 | (286) |
| 如何修改磁盘文件的卷名 | (287) |
| 大批量文件在硬盘上的保护方法 | (288) |
| COMX 机数据保护妙法 | (290) |
| 多功能磁盘管理软件(COPY)(PLUS4.3) | (10) |
| APPLE—II DOS3.3磁盘文件的删除与恢复 | (291) |

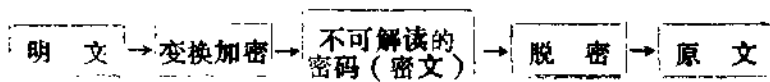
计算机与现代密码技术

复旦大学 李为鉴 鲍振东

摘要: 本文简要地介绍了加密的基本概念和加密的一般方法。在此基础上较具体地介绍了二进制的加密方法。文中着重介绍的DES体制、RSA体制和渐缩体制都是现在密码技术的研究成果,是在计算机上对信息进行加密的具有代表性的几个加密体制,很有应用价值。

在数据处理中,为了保密起见,可设法将数据进行适当的交换,使之成为“面目全非”的密码信息,这个过程通常称为数据加密,这是数据保密的一种重要手段,有关加密的各种方法不断地出现,讨论十分活跃。特别是计算机用于军事、经济、信贷、储蓄、管理等各个领域后,人们对数据信息的保密要求越来越重视。与之相应地就有一个如何将加密后的信息还原成原来数据的“本来面目”这个过程称为信息脱密。很明显,如果一个加密后的信息,很容易被别人找到脱密的办法,那么,这就是一个价值不大的加密系统,所以,研究加密系统,使得加密后的信息很难被别人找到脱密的办法,就形成一门新的学科——密码学。

我们可将加密到脱密的过程用下面的简图表示:



上述过程所研究的原理,手段和方法,就是密码学的基本内容。

密码学可以追溯到4000年前的象形文字时期。在数千年的密码历史中,真正用密码机进行实际使用,这还仅仅是进入二十世纪以后的事,A·S·Vernam于1920年所发明的电传打字机中的加密方式就可算是最早的“机械化密码”。

加密的一般方法

加密的方法大体上有以下两种:

1. 代码法

设明文中所用到的所有词所组成的集合为A,密码字符或数字的集合B,我们确定一种——对应的关系。

$$f: A \rightarrow B$$

对于每一个 $a \in A$

$$f(a) = b \quad b \in B$$

且 $f^{-1}(b) = a$

加密和脱密就是依靠着这种对应关系所编制的密码词典。显然,用这种方法进行加密,机动性很差,工作效率不高,如果利用电子计算机,那就要占用大量的存储空间以存放该密码词典。因此,这在实际上是被认为不适宜应用于计算机的加密方法。

2. 密 法

密码法通常有代替法、置换法和乘积密码法。

代替法：将明文中的每个字母用别的字母或符号来代替，从而达到加密的一种方法。

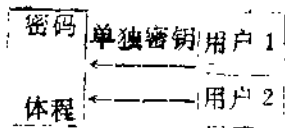
置换法：将明文中的字母的排列顺序进行改变，从而达到加密的一种方法。

乘积密码法：通过混合采用代替法和置换法，使明文变换成难于破译的密文的一种方法。

事实表明，特别是在第二次世界大战以后，关于密码方法的研究表明，乘积密码法是一种较可靠的加密方法。

密码体制的概述

如上所述，乘积密码法是混合地使用代替法和置换法所得到的一种比较好的加密方法。用不同的混合法所得的加密方法就形成不同的密码体制。



一般地说，加密体制就是用各种加密方法组合起来所形成的一种算法，并且通过该体制的用户所选定的单独密钥的作用来决定算法的具体实现。

在这种情况下，如果整个算法也能保密的话，这对密码的用户来说，当然是再好也没有了，然而，密码体制是由电子线路、计算机程序来实现的，因此，就密码体制来说，在一般情况下是难以保密的。可见，密码体制的设计必须把注意力集中在这样一点上：使企图破译密码的人，在不知道密钥的情况下，难以实现对密文的破译。这样，保护整个数据的机密，就变成保护密钥的机密就可以了。

密码体制的好坏，应该有一些准则，按照C.E.Shannon的提法，应该具备以下五个准则，

- (1) 破译密码需要极大的工作量。
- (2) 密钥的长度很小。
- (3) 加密和脱密所进行的操作比较简单。
- (4) 即使产生错误，错误的扩散也很小。
- (5) 信息被加密后并不改变原信息的长度。

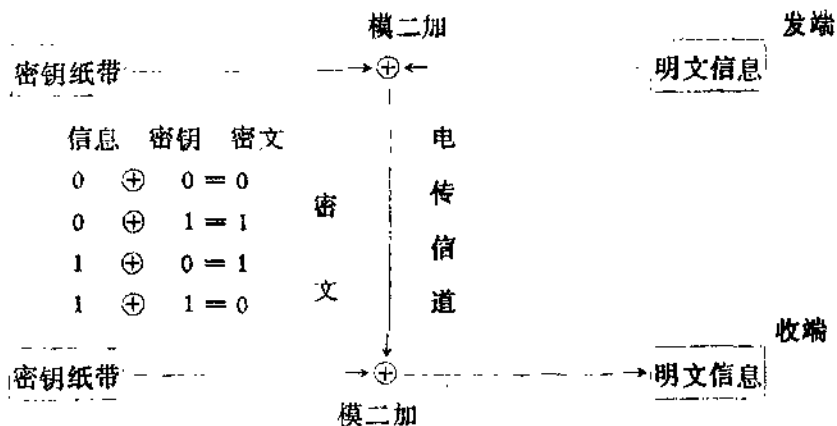
由于现代计算机的发展，算法可以组合到硬件中去，还可以用微编码和微程序，因此，在不降低容量的范围内，算法可以搞得复杂化些。

计算机都是二进制运算的，所以有关二进制信息的加密方法，很自然地引起人们的重视。

早在1920年弗纳姆就提出过一个加密方案，如下图所示：

这个方案的实现要求发端与收端有相同的二进制数码序列的密钥纸带，且双方要保持同步。对于发端来说，将明文信息与密钥序列模二相加后形成密文，并通过电传信道将密文发送出去，对于收端来说，与发端同步，并将收到的密文与密钥序列通过模二加法而得到原来的明文。

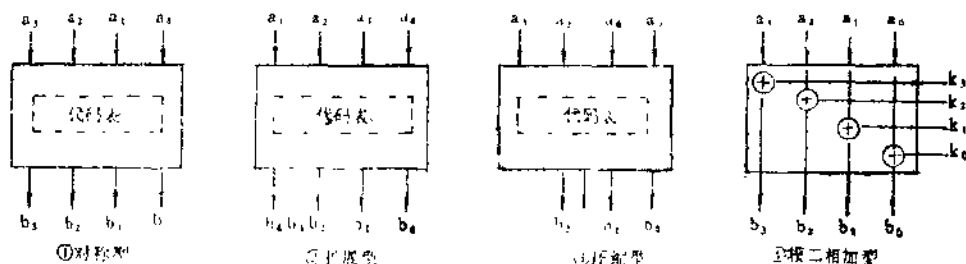
在这个方案中，特定的密钥序列只用一次，因此，即使密文被窃，也是难以破译的，密钥只用次一换，这就称为“一次一密体制”。显然，当明文信息量很大时，密钥序列也就相



应地很长。

对于二进制信息，也可进行代替和置换，简述如下：

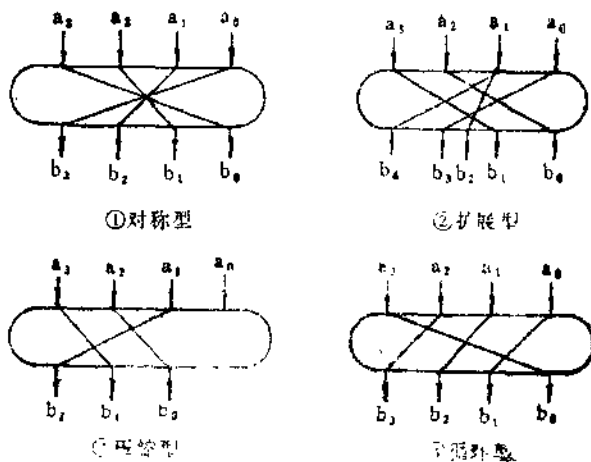
代替：设 a 为输入， b 为输出。



| | | | | | | | | | | | | | | | | |
|---|----|----|----|---|----|----|----|----|----|----|----|----|----|----|----|----|
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 0 | 14 | 4 | 13 | 1 | 2 | 15 | 11 | 8 | 3 | 10 | 6 | 12 | 5 | 9 | 0 | 7 |
| 1 | 0 | 15 | 7 | 4 | 14 | 2 | 13 | 11 | 10 | 6 | 12 | 11 | 9 | 5 | 3 | 8 |
| 2 | 4 | 1 | 14 | 8 | 13 | 6 | 2 | 11 | 15 | 12 | 9 | 7 | 3 | 10 | 5 | 0 |
| 3 | 15 | 12 | 8 | 2 | 4 | 9 | 1 | 1 | 5 | 11 | 3 | 14 | 10 | 0 | 6 | 13 |

只对压缩型举例说明：对于输入 $(a_3 a_2 a_1 a_0)_2$ 按照以下的代码表中的 $(a_5 a_0)_2$ 行与 $(a_4 a_3 a_2 a_1)_2$ 列交叉位置上的数的二进制数作为输出。

例如：以 $(110111)_2$ 作为输入，则输出应为：



$(11)_2 = 3$ 行与 $(1011)_2 = 11$ 列交叉位置上的数14

故输出应为 $(1110)_2 = 6$

置换：同样，设a为输入，b为输出

只对扩展型举例说明：由下表可知，将输入的第32位作为输出的第一位；输入的第一位作为输出的第二位；……；输入的第4位既作为输出第五位，又作为输出的第七位；……；输入的第32位作为输出的第47位；输入的第一位作为输出的第48位。

| | | | | | | | | | | | |
|---|---|---|---|---|---|--|--|--|--|--|--|
| ① | ② | ③ | ④ | ⑤ | ⑥ | | | | | | |
| ⑦ | ⑧ | ⑨ | ⑩ | ⑪ | ⑫ | | | | | | |
| ⑬ | ⑭ | ⑮ | ⑯ | ⑰ | ⑱ | | | | | | |
| ⑲ | ⑳ | ㉑ | ㉒ | ㉓ | ㉔ | | | | | | |
| ㉕ | ㉖ | ㉗ | ㉘ | ㉙ | ㉚ | | | | | | |
| ㉛ | ㉜ | ㉝ | ㉞ | ㉟ | ㊱ | | | | | | |
| ㊲ | ㊳ | ㊴ | | ㊵ | ㊶ | | | | | | |
| ㊷ | ㊸ | ㊹ | | ㊺ | ㊻ | | | | | | |

例如，对于输入

| | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 13 | 14 | 15 | 16 | 17 |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | | |
| 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 28 | 29 | 30 | 31 | | |
| 0 | | | | | | | | | | | | | |
| 32 | | | | | | | | | | | | | |

根据上表，经扩展后将输出

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | | | | | | | | | | | |

我们知道，密码体制的要害是难以破译的特征，因此，一个密码体制，总应该有某种关键所在。设计陷门函数，并在密码体制中应用陷门函数，这就是一个很重要的手段。

所谓陷门函数是指具有以下性质的函数F：

对于定义域X，值域Y，若

(1) 计算函数F的算法存在，即对于 $x_i \in X$ ， $F(x_1, x_2, \dots, x_n) = y \in Y$ ，是容易实现的。

(2) 对于几乎所有的 $y \in Y$ ，要求出 x_i 所需要化费的计算时间或占用的存储空间，实际上都是不可能实现的。

例如，一个大的合整数n，分解它为素因数的问题。在每次运算操作时间为 $1 \mu s$ 的机器上，当 $n=30$ 位十进制时，约化费3.9小时；当 $n=100$ 位时，需74年；当 $n=200$ 位时，要用 3.8×10^9 年。

因此，若X是由100位以下的所有的素数对组成的集合，则对于 $(x_1, x_2) \in X$ ，若定义函数 $F(x_1, x_2) = x_1 \cdot x_2$ ，由于一个大的合数分解成两个素数的乘积是一个要化费很多时间的问题，所以，这里所定义的F便可看作是一个陷门函数。

几个具体的加密体制

加密体制有很多，这里就几个具有代表性的加密体制作出介绍。

一、DES体制

DES是Data EncryPtion Standard的缩写，是美国国家标准局的数据加密标准。这个体制于1971—1972初由IBM公司的沃尔特·塔奇曼(W. Tuchman)和卡尔耶迈斯(C. Meyers)研究成功的，在1973—1974年，美国国家标准局曾两次公开征求能用电子计算机的保密方案，经过挑选采用了DES系统。经过几年的研究和讨论于1977年1月批准，1977年7月15日生效。后来，花了将近17年人的讨论没有找到简捷的方法来攻击。如果穷举法来攻击的话，即使一个微秒穷举一个密钥，共有 2^{56} 个密钥，就要花费约2283年。时间。

DES体制是将加密的信息按每64比特分成组，然后用密钥 $k_i (i=1, 2, \dots, 16)$ 经过16次迭代加密运算，从而得到64比特的密文作出输出。

DES体制的加密算法，如图1所示：

这里，作三点说明：

(1) 对于 $f(R, K)$ ，采用如下的体制(图2)

(2) 这个体制经过如此复杂的16轮操作，其目的就是为使明文尽可能增大其混乱性和扩散性，使得输出不残存统计规律，以达到破译者不能从反向推算出密钥。

(3) 在算法中所使用的密钥 $K_1—K_{16}$ ，是用户从给定的64bit的主密钥经过一定运算而产生的，并不是对用户直接给出16种密钥。

由于在主密钥的64bit中，有8bit是用作奇偶检验码，剩下56bit才作为有效密钥使用，因此，用户可选用密钥可有 2^{56} 种。

因此，使即在给定输入(明文)和输出(密文)的情况下，要想从中推出密钥 K_1, K_2, \dots, K_{16} ，必须用穷举法去搜索 2^{56} 种情况，这在实际上是难以完成的，尽管对这个系统有不同看法，但是塔奇曼和耶迈斯认为：DES系统在可以预见的将来(约5年到10年时间内)是保密的，用计算机来破译DES密码，将花费极大的费用和力量，因而也将是不现实的。

二、RSA体制(即由Rivest Shamir和Adleman三人提出的体制)

1976年由W. Diffie和M. Hellman [1]在“密码学新方向”中提出了公开密钥密码的设想。所谓“公开”就是加密密钥可以公开，而解密密钥予以保密。这样，关于密钥分发问题就很方便了，公开密钥就可以像电话号码本那样地公开。譬如，A发给B，那么A只要查一下密钥本上有关B的加密密钥，然后，A就可按这个公开密钥对明文进行加密后发送给B，最后，B就根据自己的解密密钥而将密文变换成原来的明文。

对于任何一个明文P，如果明文很长的话，可以将它分段，每一段信息N可以看成0到n-1的一个数。这里n是一个很大的合数，取数r，将N加密为M，使得 $N^r \equiv M \pmod{n}$ ，称 (r, n) 为加密密钥，它可以公开。可以选取一个适当的s，使得 $M^s \equiv N \pmod{n}$ ，称 (s, n) 为脱密密钥，将它进行保密。

对于n, r, s的选取，可以如下进行：

1. 先取两个非常大的随机素数P, q(p, q位)均为几百位且这两个数之间相差近一百，令

862159

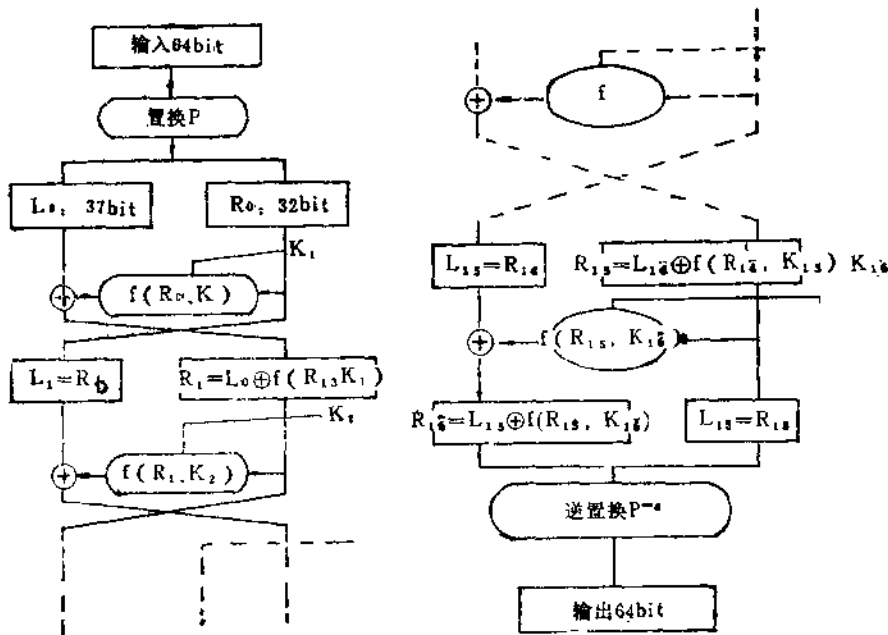


图 1

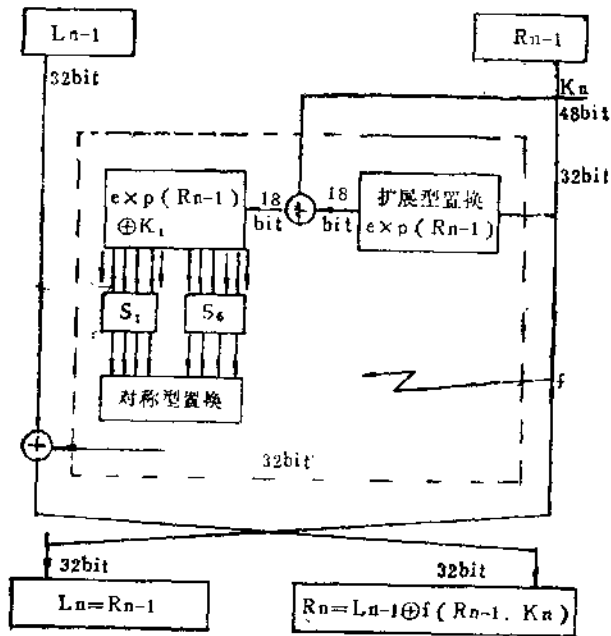


图 2

$n = P \cdot q$.

2. 随机地取一个大整数 r , 使它带足, $\text{GCD}(r, P-1) = 1$, $\text{GCD}(r, q-1) = 1$.

3. 由 $s \cdot r \equiv 1 \pmod{(p-1) \cdot (q-1)}$, 求出整数 s .

上述办法选取的 n, r, s , 由于 n 很大, 在不路道 P, q 的情况下, 要从 n 出发进行分解是很难的, 因此, 在公开 (r, n) 的情况下, 要求得 s 是困难的。这就达到了保密的目的。

现将用 (r, n) 作为加密密钥, (s, n) 作为脱密密钥的算法及其正确性证明如下,

$$N^r \equiv M \pmod{n} \quad \text{加密}$$

$$M^s \equiv (N^r)^s \equiv N^{r \cdot s}$$

$$\text{由于 } r \cdot s \equiv 1 \pmod{(p-1) \cdot (q-1)}$$

$$\text{所以 } r \cdot s = k \cdot (p-1) \cdot (q-1) + 1$$

$$\text{故 } N^{r \cdot s} = N^{k \cdot (p-1) \cdot (q-1) + 1}$$

$$\text{由 Fermat 定理可知, } a^{p-1} \equiv 1 \pmod{p}$$

$$\text{所以 } (N^r)^s \equiv N \pmod{p}$$

$$(N^r)^s \equiv N \pmod{q}$$

$$\text{最后即得 } (N^r)^s \equiv N \pmod{n}$$

1979年, 由 Bob Blakley 和 G.R. Blakley 发表的长篇文章 "Security of Number Theoretic Public Key Cryptosystems Against Random Attack" 防止随机攻击的数论公开密钥密码体制的“保密性”, 详细地论述了该体制的现论依据和具体实现。整个体制实质上是 RSA 体制的推广和发展, 关于这个体制的详细论证, 可参阅上面提供的文章。现将该体制的具体算法叙述如下:

对于给定的 g, w, u

1. 随机地选择奇正整数 a , 使得

$$g-1 < \log_2 a < g + \frac{w}{2} - 1$$

2. 对于每个质数 $r < u$ 作

$$\text{GCD}(r, a), \text{GCD}(r, 2a+1), \text{GCD}(a, (u-1)/2)$$

如果这三者中有一个不是 1, 则再重做 1.

3. 判定 a 与 $2a+1$ 是否同时为质数, 若不是则重复做 1,

4. 随机地选择奇正整数 b , 使得

$$g+w-1 < \log_2 b < g + \frac{3}{2}w - 1$$

5. 对于每个质数 $r < u$, 作

$$\text{GCD}(r, b), \text{GCD}(r, 2b+1), \text{GCD}(b, (u-1)/2)$$

如果这三者中有一个不是 1, 则返回到 4.

6. 判定 b 与 $2b+1$ 是否同时为质数, 若不是则再重做 4

7. 作 $\text{GCD}(a, b), \text{GCD}(a, 2b+1)$

$\text{GCD}(2a+1, b), \text{GCD}(2a+1, 2b+1)$, 若其中一个不等于 1, 则返回到 1.

8. 解以下六对同余式

$$(i) A \equiv 0 \pmod{2a+1}$$

$$A \equiv 1 \pmod{2b+1}$$

$$(ii) B \equiv 1 \pmod{2a+1}$$

$$B \equiv 0 \pmod{2b+1}$$

$$iii) C \equiv 0 \pmod{2a+1}$$

$$C \equiv -1 \pmod{2b+1}$$

$$(iv) D \equiv -1 \pmod{2a+1}$$

$$D \equiv 0 \pmod{2b+1}$$

$$(v) E \equiv 1 \pmod{2a+1}$$

$$E \equiv -1 \pmod{2b+1}$$

$$(vi) F \equiv -1 \pmod{2a+1}$$

$$F \equiv 1 \pmod{2b+1}$$

当这六对同余方程的解都不是发文信息时，就转向 9，否则就转向 1。

9. 计算 $P=2a+1$, $q=2b+1$, $m=p \cdot q$

$$v=2ab$$

10. 由 $u \cdot d \equiv 1 \pmod{v}$ 求出其最小正整数 d 。

然后，以 (u, m) 为加密密钥， (d, m) 为脱密密钥，那么，很明显有 $(x^a)^d = x^{ad} = x^{k \cdot 2ab+1}$ ，故有 $(x^a)^d \equiv X \pmod{m}$ 。

为了使得所取的 m 较大，且 P, q 相差也较大，所以，就要求使 m 满足以下的关系式

$$2g < \log_2 m < 2g + 2w$$

如果要求选取的 P, q 的范围为：

$$2^g < P, q < 10 \times 2^g$$

则有

$$2g < \log_2 (P \cdot q) < 2\log_2 10 + 2g$$

即

$$2g < \log_2 m < 2\log_2 10 + 2g$$

按上述范围取 P 和 q ，那么 $w = \log_2 10 = 3.321\dots$ 。称 w 为宽度。

如果取 P, q 为两个十进制的 100 位的质数，那么，以上述范围，可求得 $g = 328.870\dots$ ，称 g 为规格。

为了保证加密密钥中的 u 能使下面的不等式成立

$$x^a > m \quad (\text{对于每一个正整数 } x \geq 1)$$

只须取 $u > 2g + 2^w$ 即可，这是因为

$$x^a > x^{2g+2^w} > x^{10^g 2^m} \geq 2^{10^g 2^m} = m$$

因此，如果取 $g = 350$, $w = 5$ ，那么， u 应取为 $u > 2g + 2^w = 2 \times 350 + 2 \times 5 = 710$

随着计算机网络的发展，公开密钥密码体制较好地解决了通信用户数量极其庞大时密钥的分发问题，可以与数量不确定的大量用户进行保密通信，这显然是它的一大优点。然而，在具体的实施中，对于公开密钥表的使用和管理能否有可信赖的公正第三者这也是一个实际问题，保密信息总是在互相认识的双方为了保守信息的机密而采用的通信方式，在这种情况下，通信双方之间或几方之间当然是经过周密而又慎重的考虑才确定加密和脱密的办法，因此，有关密钥的分发数量也是相当有限并且受到严格限制的。完全的公开究竟又有多大意义呢？有关这些问题，只能在实际使用的过程中获得满意的解答。

三、渐缩体制

对于给定的正整数 a_1, a_2, \dots, a_n 和 S ，求使得

$$S = a_1 x_1 + a_2 x_2 + \dots + a_n x_n$$

成立的 $\{0, 1\}$ 解（即 $x_i \in \{0, 1\}$ ）。这就称为渐缩问题。这个问题当 n 相当大

时，解的选择方式用穷举法去做的话，共有 2^n 种不同的方式。因此，对于一般的 $a_i (i=1, 2, \dots, n)$ 来说，渐缩问题是一个NP问题。

当 a_1, a_2, \dots, a_n 满足以下的关系式时

$$\begin{aligned} a_2 &> a_1 \\ a_3 &> a_1 + a_2 \\ a_4 &> a_1 + a_2 + a_3 \\ &\vdots \\ a_k &> a_1 + a_2 + \dots + a_{k-1} \end{aligned} \quad (3.1)$$

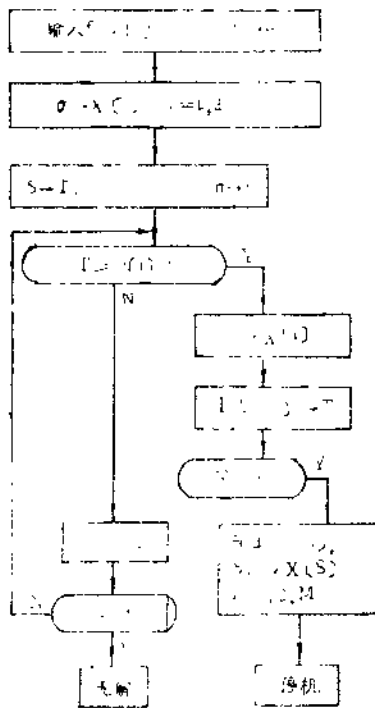


图 3

渐缩问题的解是容易得到的。

例如： $a_1 \ a_2 \ a_3 \ a_4$
 $1 \ 4 \ 6 \ 12$
 $S \ 19 =$

因为： $19 > 12$ 故应取 $x_4 = 1$
 $19 - 12 = 7$
 $7 > 6$ 故应取 $x_3 =$
 $7 - 6 = 1$
 $1 < 4$ 故应取 $x_2 = 0$
 $1 = 1$ 故应取 $x_1 = 1$
 $1 - 1 = 0$ 正好结束

所以，就解得 $x_1 = 1, x_2 = 0, x_3 = 1, x_4 = 1$

对于满足(3.1)的渐缩问题，可以有以下的算法。(图3)

在介绍过渐缩问题后，我们再来建立渐缩体制。

设已知 $a_i (i=1, 2, \dots, n)$ 满足条件(3.1)，任取正整数 m 满足。

$$m > \sum_{i=1}^n a_i$$

取得 w ，使得 $(w, m) = 1$ ，即 w 与 m 互质

由 $w' \cdot w \equiv 1 \pmod{m}$ 解出 w'

作 $a_i' \equiv a_i w_i \pmod{m}$

对于已知的 $\{0, 1\}$ 信息 x_1, x_2, \dots, x_n ，利用 a_i' 来进行加密，得到密文 S

$$S \equiv \sum_{i=1}^n a_i' x_i \pmod{m}$$

接收者收到 S 后，可用以下的方法来脱

$$S w' \equiv S' \pmod{m}$$

$$\begin{aligned}
\text{因为 } S' &\equiv S_{w'} \equiv \sum_{i=1}^n a_i' w' x_i \\
&\equiv \sum_{i=1}^n a_i w w' x_i \\
&\equiv \sum_{i=1}^n a_i x_i \pmod{m}
\end{aligned}$$

所以，就可以解出 x_1, x_2, \dots, x_n 。

在这个体制中， (a', m) 可以分开， w' 作为解密密钥。

参考文献(略)

文献出处：《电脑应用时代》1985年3期1—8页

多功能磁盘管理软件 COPY] [PLUS 4.3

徐凯泉

在APPLE机上进行磁盘作业时，为了处理类似拷贝文件，修复DOS，改写扇区，编辑扇区内容，恢复被删除的文件；检查文件错误，查询起始地址，了解磁盘文件空间分布等多种复杂问题，需要自编多个小程序，使用起来很不方便。在这里有必要推广多功能磁盘管理软件COPY] [PLUS 4.3。它能使你在APPLE磁盘作业中得心应手，事半功倍。

该软件共有16项功能，下面将其经常使用的几个部分重点介绍如下：

将含有COPY] [PLUS 4.3文件的磁盘放入驱动器D中，打入BRUN COPY] [PLUS 4.3，待片刻便出现以下菜单（此时可将该磁盘取出）：

- | | |
|-----------------|-------------------------|
| 1. CATALOG | 9. TRACK/SECTER |
| 2. COPY | 10. VIEW FILES |
| 3. BEYCOPY | 11. FIX FILE SIZES |
| 4. DELETE | 12. CHANGE BOOT PROGRAM |
| 5. LOCK UNLOCK | 13. UNDELETE FILES |
| 6. RENAME FILES | 14. SECTOR EDITOR |
| 7. FORMET | 15. NEW DISK INFO |
| 8. VERIFY | 16. BOOT DISK |

1) CATALOG列出磁盘目录，可列出四种目录供选择：

①NORMAL普通标准目录，相当于原CATALOG命令

②W/FILE LENGTHS以10进制和16进制两种方式列出当前磁盘上所有文件长度。

其中B型文件，同时列出起始地址。

③W/DELETE FILES列出当前磁盘中曾经建立过而又被删去的文件目录

④W/HIDDEN CHERS列出包含有隐藏字符（如^A、^D等）的文件名。

2) COPY用来拷贝文件，可有三种形式：

①FILES拷贝单个文件。一次可选择多个文件，拷贝到另一磁盘上。

②DISK将一个磁盘上全部文件拷贝到另一磁盘上。

③DOS将一个磁盘上的DOS系统拷入另一磁盘，而不删去其他文件。这对于有些磁盘DOS损坏不能引导时具有修复功能。

4)5)6)7)部分因使用较为容易，在此不再重复。

8) VERIFY检查磁盘

①DISK按磁道检查全盘。若遇到错误，便显示磁道和扇区号。

②FILES按文件逐个检查。当遇到文件不能调入内存时将显示文件名。

③D-SPEED用来检查磁盘速度，并指出与标准速度之差别。

9) TRACK/SECTOR MAP磁道/扇区图。将显示每个文件在磁盘中所在的位置，可供修改时查找文件位置使用。

10) VIEW FILES显示磁盘中文件所在扇区中的内容。其中①按16进制机器码表示②以文本形式显示。

11) FIX FILESIZE压缩文件在磁盘中所占多余扇区。有些文件在反复修改中，在磁盘上所占实际空间发生了变化，有时文件长度变短，而原有空间较长，因此利用功能可使多余空间得到释放，增加磁盘应有的容量。

12) CHANGE BOOT PROGRAM此功能用来改变引导程序名。可在整个磁盘中，随意命名任何一个文件名（可以是A型和B型）为引导程序名（只能选一个）。以后当引导该磁盘时，便立即运行新命名的程序。

13) UNDELETE FILES恢复曾被删除的文件。选择此功能，先列出全部被删文件的名称。你可用回车任意指定哪几个文件需要恢复。再按G即可恢复你失去的文件。若原删去的文件所在空间已被新文件占用，系统将告诉你该文件不可恢复。

14) SECTOR EDITOR扇区编辑。选择此功能后，系统显示，

```
{I} {J} {K} {M} MOVE CURSOR  
{B} BEGINNING, {E} END, {R} READ, {W} WRITE  
{H} EX OR {T} EXTENTER, {ESC} TO EXIT  
{P} PATCH DISK READ WRITE
```

当我们要修改扇区内容时，可先选〔R〕，将某一扇区内容读入内存，然后用〔I〕〔J〕〔K〕〔M〕键来移动光标到需要修改的地方去。再选择〔H〕按16进制机器码修改（显示于屏幕左边）；若选〔T〕则用文本形式修改（显示于屏幕右边）。〔B〕表示将光标移至文首。〔E〕表示将光标移至文尾。修改结束后按〔W〕可将已修改的内容存入原扇区。按ESC键即退出该功能。

15) NEW DISKINFO重新设定初值。通过此功能还可设PRINTER ON状态，对必要的内容进行打印。

16) BOOT DISK退出此软件。在D1驱动器中，插好待起动的软盘，按回车后即可引导。

文献出处：《电子与电脑》1988年4期4—5页

数据加密的体制及应用

哈尔滨工业大学 朱志莹 曹珍富

一、引言

计算机科学和其他科学的交叉渗透,产生了许多离散结构的问题,解决这些离散结构问题所创造的方法,往往产生新的科学分支,突破传统方法的应用范围。

我们熟知,使用计算机的一个基本手段是将所述问题数字化,例如数字信号处理、图象处理等都可归结为数据处理,而数据处理除了数据率庞大需要“压缩”以外,还有数据的加密处理。例如,随着计算机的遍及,“电子信件”将会逐步取代传统的书面信件,人们希望“电子信件”能象传统的书面信件一样,只有合法接收者才能知道信件内容,而任何第三人都只能看到“信封外皮”,无法知道信的内容。这就要求对信的内容(可以数字化)提供加密。再如,计算机中重要数据的加密,重要软件的加密等等,都是数据加密的例子。

因为传统的加密方法是严格保密的,不便在更多的场合下使用(往往只能是政府的军事、外交等部门所专用)。1976年11月,美国的Ciffie和Hellman [1]突破了传统的密码技术,提出了数据加密的一种新的体制,该体制要求每一用户都有自己的两种密钥——加密密钥和解密密钥,其中加密密钥公开,便于通信者用来加密明码,而解密密钥严格保密起来。非法接收者想从公开的加密密钥解出解密密钥几乎是不可能的。这是轰动全球的公开密钥体制(public key cryptosystem),寻找这种体制可以归结为寻找所谓的陷门单向函数,本文将依次介绍陷门单向函数以及目前人们已经找到的一些实现方案。

二、陷门单向函数

首先,我们将Diffie和Hellman提出的公开密钥体制用数学语言来描述:

设用户 x 的加密密钥是 E_x ,解密密钥是 D_x ,则要求 E_x 和 D_x 实现互逆变换,即

$$D_x \cdot E_x = E_x \cdot D_x = I,$$

其中 I 是恒等变换,而 E_x 公开,这里要求两条:① D_x 和 E_x 都容易计算;②仅从公开的 E_x 寻找 D_x 是困难的。

现在我们一般地看一下两个用户 a 和 b 是怎样实现秘密通信的。设用户 Q 开送信息 P (称为明码)给用户 b ,则有

(1) a 用 b 公开的加密密钥 E_b 对 P 进行变换,得到密码:

$$C = E_b(P)$$

(2) b 收到 a 发送的 C 后,用只有他自己知道的解密密钥 D_b 变换 C 恢复明码 P ,

$$D_b(C) = D_b(E_b(P)) = I(P) = P$$

由此可见,寻找适合下面三个条件的函数 $f(n)$ 是构造公开密钥体制的一个重要途径:

(a) 对 $f(n)$ 的定义域中的每一个 n ,均存在函数 $f^{-1}(m)$,使 $f^{-1}(f(n)) = n$;

(b) $f(n)$ 与 $f^{-1}(m)$ 都容易计算;