MD5 算法研究

张裔智[1] 赵 毅[2] 汤小斌[1]
(重庆三峡学院网络中心 重庆 404000)[1] (重庆交通大学教育技术中心 重庆 400074)[2]

摘 要 随着网络技术的迅速发展,信息加密技术已成为保障网络安全的一种重要手段,加密算法已经成为人们的一个研究热点。本文对 MD5 算法进行了深入研究,介绍 MD5 算法的产生背景、应用及其算法流程,并提出了 MD5 算法的一个改进方案。

关键词 MD5,算法,加密技术

MD5 Algorithm

ZHANG Yi-zhi[1] ZHAO Yi[2] TANG Xiao-bin[1]
(Chongqing Three Gorges University, Chongqing 404000, China)[1]
(Educational Technology Center, Chongqing Jiaotong University, Chongqing 400074, China)[2]

Abstract With the rapid development of Internet technology, information encryption technology has become an important means to ensure network security and encryption algorithm has become a hotspot. This paper introduces the MD5 algorithm background, applications and processes algorithm, proposes improvement scheme.

Keywords MD5, Algorithm, Encryption technology

1 引言

从 Rivest1989 年开发出 MD2 算法开始就揭开了人们对加密算法的新一轮研究,进而推出了 MD3,MD4 算法,为了加强算法的安全性,满足信息发展对网络安全的要求提出了趋近成熟的 MD5 算法。

MD5 的全称是 Message-Digest Algorithm 5(信息-摘要算法)[1],在 20 世纪 90 年代初由 MIT Laboratory for Computer Science 和 RSA Data Security Inc 的 Ronald L. Rivest 开发出来,经 MD2,MD3 和 MD4 发展而来。它的作用是让大容量信息在用数字签名软件签署私人密匙前被"压缩"成一种保密的格式(就是把一个任意长度的字节串变换成一定长的大整数)。不管是 MD2、MD4 还是 MD5,它们都需要获得一个随机长度的信息并产生一个 128 位的信息摘要。虽然这些算法的结构或多或少有些相似,但 MD2 的设计与 MD4 和 MD5 完全不同,那是因为 MD2 是为 8 位机器做过设计优化的,而 MD4 和 MD5 却是面向 32 位的电脑。

2 MD5 算法应用

MD5 的典型应用是对一段信息(Message)产生信息摘要(Message-Digest),以防止被篡改。比如,在 UNIX 下有很多软件在下载的时候都有一个文件名相同,文件扩展名为. md5 的文件,在这个文件中通常只有一行文本,大致结构如:

MD5 (tanajiya. tar. gz) = 0ca175b9c0f726a831d895e269332461

这就是 tanajiya. tar. gz 文件的数字签名。MD5 将整个文件当作一个大文本信息,通过其不可逆的字符串变换算法,产生了这个唯一的 MD5 信息摘要。如果在以后传播这个文件的过程中,无论文件的内容发生了任何形式的改变(包括人为修改或者下载过程中线路不稳定引起的传输错误等),只要你对这个文件重新计算 MD5 时就会发现信息摘要不相同,由此可以确定你得到的只是一个不正确的文件。如果再有一个第三方的认证机构,用 MD5 还可以防止文件作者的"抵赖",这就是所谓的数字签名应用。

MD5 还广泛用于加密和解密技术上。例如:Cisco 的 Enable 的密码和 Unix 系统中用户的密码都是以 MD5(或其它类似的算法)经加密后存储在文件系统中。当用户登录的时候,系统把用户输入的密码计算成 MD5 值,然后再去和保存在文件系统中的 MD5 值进行比较,进而确定输入的密码是否正确。通过这样的步骤,系统在并不知道用户密码的明码的情况下就可以确定用户登录系统的合法性。这不但可以避免用户的密码被具有系统管理员权限的用户知道,而且还在一定程度上增加了密码被破解的难度。

3 MD5 算法原理

对任意长度的信息输入,MD5 都将产生一个长度为 128 比特的输出。这一输出可以被看作是原输入报文的"报文摘要值 (Message Digest)"。MD5 以 512 位分组来处理输入的信息,且每一分组又被划分为 16 个 32 位子分组,经过了一系列的处理后,算法的输出由四个 32 位分组组成,将这四个 32 位分组级联后将生成一个 128 位散列值。

在 MD5 算法中,首先需要对信息进行填充,使其字节长度对 512 求余的结果等于 448。因此,信息的字节长度(Bits Length)将被扩展至 $N*512+448$,即 $N*64+56$ 个字节(Bytes),N 为一个正整数。填充的方法如下,在信息的后面填充一个 1 和无数个 0,直到满足上面的条件时才停止用 0 对信息的填充。然后,再在这个结果后面附加一个以 64 位二进制表示的填充前信息长度。经过这两步的处理,现在的信

张裔智 工学硕士,副教授,研究方向为计算机网络。

息字节长度=$N*512+448+64=(N+1)*512$，即长度恰好是512的整数倍。这样做的原因是为满足后面处理中对信息长度的要求。

3.1 MD5算法详细描述

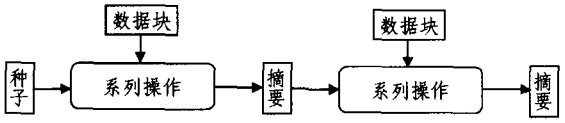

图1 MD5分组摘要算法

假设有一个b位长度的输入信号，希望产生它的报文摘要，此处b是一个非负整数，b也可能是0，不一定必须是8的整数倍，它可能是任意大的长度。设想信号的比特流如下所示：

$m\_0, m\_1, \cdots, m\_\{b-1\}$

(1)补位。MD5算法是对输入的数据进行补位，使得如果数据位长度LEN对512求余的结果是448，则数据扩展至$K\times512+448$位，即$K\times64+56$个字节，K为整数。补位操作始终要执行，即使数据长度LEN对512求余的结果已经是448。

具体补位操作：补一个1，然后补0至满足上述要求。最少要补一位，最多补512位。补位后报文信息长度如图2所示。

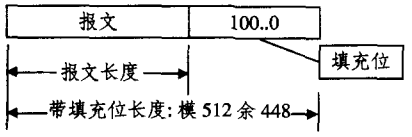

图2 补位后报文信息长度1

(2)补数据长度。用一个64位的数字表示数据的原始长度b，把b用两个32位数表示。那么只取B的低64位。当遇到b大于2^{64}这种极少遇到的情况时，此时，数据就被填补成长度为512位的倍数，也就是说，此时的数据长度是16个字(32位)的整数倍数。用$M[0,\cdots,N-1]$表示此时的数据，其中的N是16的倍数。具体情况如图3所示。

(3)初始化MD缓冲器。用一个4个字的缓冲器(A,B,C,D)来计算报文摘要，A,B,C,D分别是32位的寄存器，初始化使用的是十六进制表示的数字，MD5中这四个32位被称作链接变量(Chaining Variable)的整数参数。它们分别为：

A=0X01234567　　B=0X89abcdef

C=0Xfedcba98　　D=0X76543210

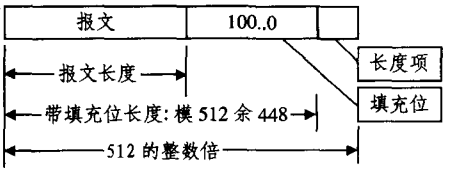

图3 补位后报文信息长度2

当设置好这四个链接变量后，就开始进入算法的四轮循环运算。循环的次数是信息中512位信息分组的数目。

将上面四个链接变量复制到另外四个变量中：A到a，B到b，C到c，D到d。

主循环有四轮(MD4只有三轮)，每轮循环都很相似。第一轮进行16次操作。每次操作对a,b,c和d中的其中三个作一次非线性函数运算，然后将所得结果加上第四个变量，文本的一个子分组和一个常数。再将所得结果向右环移一个不定的数，并加上a,b,c或d中之一。最后用该结果取代a,b,c或d中之一。以下是每次操作中用到的四个非线性函数(每轮一个)。

$F(X,Y,Z)=(X\&Y)|((\sim X)\&Z)$

$G(X,Y,Z)=(X\&Z)|(Y\&(\sim Z))$

$H(X,Y,Z)=X^Y^Z$　　$I(X,Y,Z)=Y^(X|(\sim Z))$

(说明：& 是与，| 是或，～是非，^是异或)

这四个函数的说明：如果X,Y和Z的对应位是独立和均匀的，那么结果的每一位也应是独立和均匀的。F是一个逐位运算的函数。即，如果X，那么Y，否则Z。函数H是逐位奇偶操作符。

设Mj表示消息的第j个子分组(从0到15)，$\langle\langle\langle w$表示循环左移w则四种操作为：$FF(a,b,c,d,Mj,w,ti)$表示$a=b+((a+(F(b,c,d)+Mj+ti)\langle\langle\langle w)$

$GG(a,b,c,d,Mj,w,ti)$表示$a=b+((a+(G(b,c,d)+Mj+ti)\langle\langle\langle w)$

$HH(a,b,c,d,Mj,w,ti)$表示$a=b+((a+(H(b,c,d)+Mj+ti)\langle\langle\langle w)$

$II(a,b,c,d,Mj,w,ti)$表示$a=b+((a+(I(b,c,d)+Mj+ti)\langle\langle\langle w)$

这四轮(64步)是：

第一轮	第三轮
FF(a,b,c,d,M0,7,0xd76aa478)	HH(a,b,c,d,M5,4,0xfffa3942)
FF(d,a,b,c,M1,12,0xe8c7b756)	HH(d,a,b,c,M8,11,0x8771f681)
FF(c,d,a,b,M2,17,0x242070db)	HH(c,d,a,b,M11,16,0x6d9d6122)
FF(b,c,d,a,M3,22,0xc1bdceee)	HH(b,c,d,a,M14,23,0xfde5380c)
FF(a,b,c,d,M4,7,0xf57c0faf)	HH(a,b,c,d,M1,4,0xa4beea44)
FF(d,a,b,c,M5,12,0x4787c62a)	HH(d,a,b,c,M4,11,0x4bdecfa9)
FF(c,d,a,b,M6,17,0xa8304613)	HH(c,d,a,b,M7,16,0xf6bb4b60)
FF(b,c,d,a,M7,22,0xfd469501)	HH(b,c,d,a,M10,23,0xbebfbc70)
FF(a,b,c,d,M8,7,0x698098d8)	HH(a,b,c,d,M13,4,0x289b7ec6)
FF(d,a,b,c,M9,12,0x8b44f7af)	HH(d,a,b,c,M0,11,0xeaa127fa)
FF(c,d,a,b,M10,17,0xffff5bb1)	HH(c,d,a,b,M3,16,0xd4ef3085)
FF(b,c,d,a,M11,22,0x895cd7be)	HH(b,c,d,a,M6,23,0x04881d05)
FF(a,b,c,d,M12,7,0x6b901122)	HH(a,b,c,d,M9,4,0xd9d4d039)
FF(d,a,b,c,M13,12,0xfd987193)	HH(d,a,b,c,M12,11,0xe6db99e5)
FF(c,d,a,b,M14,17,0xa679438e)	HH(c,d,a,b,M15,16,0x1fa27cf8)
FF(b,c,d,a,M15,22,0x49b40821)	HH(b,c,d,a,M2,23,0xc4ac5665)
第二轮	**第四轮**
GG(a,b,c,d,M1,5,0xf61e2562)	II(a,b,c,d,M0,6,0xf4292244)
GG(d,a,b,c,M6,9,0xc040b340)	II(d,a,b,c,M7,10,0x432aff97)
GG(c,d,a,b,M11,14,0x265e5a51)	II(c,d,a,b,M14,15,0xab9423a7)
GG(b,c,d,a,M0,20,0xe9b6c7aa)	II(b,c,d,a,M5,21,0xfc93a039)
GG(a,b,c,d,M5,5,0xd62f105d)	II(a,b,c,d,M12,6,0x655b59c3)
GG(d,a,b,c,M10,9,0x02441453)	II(d,a,b,c,M3,10,0x8f0ccc92)
GG(c,d,a,b,M15,14,0xd8a1e681)	II(c,d,a,b,M10,15,0xffeff47d)
GG(b,c,d,a,M4,20,0xe7d3fbc8)	II(b,c,d,a,M1,21,0x85845dd1)
GG(a,b,c,d,M9,5,0x21e1cde6)	II(a,b,c,d,M8,6,0x6fa87e4f)
GG(d,a,b,c,M14,9,0xc33707d6)	II(d,a,b,c,M15,10,0xfe2ce6e0)
GG(c,d,a,b,M3,14,0xf4d50d87)	II(c,d,a,b,M6,15,0xa3014314)
GG(b,c,d,a,M8,20,0x455a14ed)	II(b,c,d,a,M13,21,0x4e0811a1)
GG(a,b,c,d,M13,5,0xa9e3e905)	II(a,b,c,d,M4,6,0xf7537e82)
GG(d,a,b,c,M2,9,0xfcefa3f8)	II(d,a,b,c,M11,10,0xbd3af235)
GG(c,d,a,b,M7,14,0x676f02d9)	II(c,d,a,b,M2,15,0x2ad7d2bb)
GG(b,c,d,a,M12,20,0x8d2a4c8a)	II(b,c,d,a,M9,21,0xeb86d391)

常数 ti 可以如下选择：在第 i 步中，ti 是 4294967296 * abs(sin(i))的整数部分，i 的单位是弧度。(4294967296 等于 2 的 32 次方)，所有这些完成之后，将 A,B,C,D 分别加上 a,b,c,d。然后用下一分组数据继续运行算法，最后的输出是 A,B,C 和 D 的级联：从 A 的低字节开始，直到 D 的高字节。输出结果。

$A = A + AA \quad B = B + BB$

$C = C + CC \quad D = D + CC$

当我们按照上面所说的方法实现 MD5 算法以后，就可以用以下几个信息对我们做出来的程序作一个简单的测试，就可以检查程序的正确性。

MD5("")=d41d8cd98f00b204e9800998ecf8427e

MD5("a")=0cc175b9c0f1b6a831c399e269772661

MD5("abc")=900150983cd24fb0d6963f7d28e17f72

MD5("message digest")=f96b697d7cb7938d525a2f31aaf161d0

MD5("abcdefghijklmnopqrstuvwxyz")=c3fcd3d7612e4007dfb496cca67e13b

MD5("ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789") = d174ab98d277d9f5a5611c2c9f419d9f

MD5("12345678901234567890123456789012345678901234567890123456789012345678901234567890")= 57edf4a22be3c955ac49da2e2107b67a

4 MD5 算法的安全性能分析

4.1 MD5 与 MD4 的比较

MD5 较 MD4(1)增加了第四轮；(2)每一步均有唯一的加法常数；(3)为减弱第二轮中函数 G 的对称性从(X&Y)|(X&Z)|(Y&Z)变为(X&Z)|(Y&(~Z))；(4)第一步加上了上一步的结果，这将引起更快的雪崩效应；(5)改变了第二轮和第三轮中访问消息子分组的次序，使其更不相似；(6)近似优化了每一轮中的循环左移位移量以实现更快的雪崩效应，各轮的位移量互不相同。

MD5 算法在 32 位机器上能以很快的速度运行，MD5 算法不需要任何大型的置换列表，此算法编码简洁。MD5 算法是 MD4 报文摘要算法的扩展，MD5 算法稍慢于 MD4 算法，但是在设计上比 MD4 算法更加"保守"。设计 MD5 是因为 MD4 算法被采用的速度太快，以至于还无法证明它的可靠性，因为 MD4 算法速度非常快，处在遭受成功秘密攻击的"边缘"。MD5 则后退了一步，舍弃了一些速度以求更好的安全性。从某种意义上，MD5 也可以看作是一种"校验和"(checksum)。只是 MD5 算法要比一般的校验和的计算复杂得多，并且所得到的输出也比一般的校验和长得多，以至于：

(1)两条不同的报文具有相同的报文摘要值的可能性极小；

(2)对于预先给定的报文摘要值，要想寻找到一条报文，使得其报文摘要值与某个给定的报文摘值相等，这从计算上是不可能的；

(3)根据报文的摘要值，要想推测出原来的报文是极端困难的。

4.2 常见对 MD5 的攻击方法

(1)穷举法。MD5 的输出是一个 128 位的数。攻击者只知道这个输出结果，他要想得到输入 m 或另一个输出 m'，使得 $H(m) = H(m')$ (H 为 hash 函数)，用一台每秒运算达 10 亿次的机器也要运算 $1.07 * 10^{22}$ 年。因此是不可行的。

(2)生日攻击法。这是应用概率统计法来对 MD5 进行攻击，即试图找到一对不同的输入，经过 hash 函数后产生相同的输出。这样的输入称为匹配对。给定 hash 函数的 n 个输入和 k 个可能的输出。共有 $n(n-1)/2$ 个输入对。对每个输入对来说，两个输入产生相同输出的概率是 $1/k$。这样若输入对数目达到 $k/2$，那么就有 50%的概率得到一个匹配对。在 $n>\mathrm{sqrt}(k)$的情况下就会有较大的概率得到这样的输入匹配对。在 MD5 中就需要尝试 2^64 次。如果用每秒运算达 10 亿次的机器也得花费 58 年。显然这样的做法是不可行的，因此 MD5 算法相对来说是十分安全的。

4.3 对 MD5 的改进算法

MD5 也不能说是考虑得十分周到的，原因是：在第 i 步统一的加法常数是 $2^{32} * \mathrm{abs}(\sin(i))$的整数部分，其中 i 的单位是弧度。而其结果也就是 abs(sin(i))的前 32 位。这一点和 4 个连续的 sin 值之间的关系 $(\sin(i)+\sin(i+2))\sin(i+2)=(\sin(i+1)+\sin(i+3))\sin(i+1)$使任何 4 个连续的加法常量之间存在某种近似的关系。

所以对于每一步的加法常数所引起的关联，这里可以尝试做如下改进：选择 abs(sin(i))二进制扩展值的下一个 32 位，或者每一步都选取不同位置的 32 位来避免。从而增加其安全运算能力。

结束语 当今世界正以前所未有的速度推进社会信息化建设，人们在享受互联网带来的方便快捷的同时，也要面对互联网开放性带来的一系列网络安全的问题，我们已经对信息加密技术进行了详细深入的研究的同时，在不断完成各种相应的法规，通过有利的手段保障人们的利益。虽然 MD5 加密技术还存在着一些弊病，但由于其自身安全等级高，已经广泛应用在社会的各行各业，不久的将来 MD5 可能会完全被其它更安全的算法取代，MD5 加密技术是一种优秀的信息加密算法。

参 考 文 献

[1] Rivest R. The MD5 Message-Digest Algorithm. RFC 1321, April 1992

[2] Rivest R. The MD4 Message-Digest Algorithm. RFC 1320, April 1992

[3] Haller N. The S/KEY One-Time Password System // Proceedings of the ISOC Symposium on Network and Distributed System Security. February 1994, San Diego, CA

[4] Haller N, Atkinson R. On Internet Authentication. RFC 1704, October 1994

[5] Haller N. The S/KEY One-Time Password System. RFC 1760, February 1995

[6] 卢开澄. 计算机密码学. 北京：清华大学出版社，1998,12

[7] 张焕国. 计算机安全保密技术. 北京：机械工业出版社，1997,4

[8] 谢希仁. 计算机网络(第 2 版). 北京：电子工业出版社，1999