第3章 信息安全技术基础

3.1 密码学概论

密码学(cryptography)是一门古老而深奥的学科,它以认识密码变换为本质,以加密与解密基本规律为研究对象。Cryptography 一词来源于古希腊的 crypto 和 graphen,意思是密写。保密通信的思想和方法早在几千年前就已经有了,埃及人是最先使用特别的象形文字作为信息编码的人。随着时间推移,巴比伦、美索不达米亚和希腊都开始使用一些方法来保护他们的书面信息。对信息进行编码曾被 Julias Caesar(凯撒大帝)使用,也曾用于历次战争中,包括美国独立战争、美国内战和两次世界大战。最广为人知的编码机器是 German Enigma,在第二次世界大战中德国人利用它创建了加密信息。此后,由于 Alan Turing 和 Ultra 计划以及其他人的努力,终于对德国人的密码进行了破解。当初,计算机的研究就是为了破解德国人的密码,人们并没有想到计算机给今天带来的信息技术革命。近年来,密码学研究之所以十分活跃,主要是它与计算机科学的蓬勃发展密切相关;此外,还有在电信、金融领域和防止日益广泛的计算机犯罪的需要。在互联网出现之前,密码技术已经广泛应用于军事和民用方面。现在,密码技术在计算机网络中的应用实例越来越多。

3.1.1 密码学的历史

密码学的发展历程大致经历了三个阶段:古代加密方法、古典密码和近代密码。

1. 古代加密方法(手工阶段)

这一时期的密码技术源于战争需求,可以说是一种艺术,而不是一种科学。存于石刻或史书中的记载表明,许多古代文明,包括埃及人、希伯来人、亚述人都在实践中逐步发明了密码系统。从某种意义上说,战争是科学技术进步的催化剂。人类自从有了战争,就面临着通信安全的需求,密码技术历史悠久、源远流长。古代加密方法大约起源于公元前 440 年,出现在古希腊战争中的隐写术,当时为了安全传送军事情报,奴隶主剃光奴隶的头发,将情报写在奴隶的光头上,待头发长长后将奴隶送到另一个部落,再次剃光头发,原有的信息复现出来,从而实现这两个部落之间的秘密通信。

我国古代也早有以藏头诗、藏尾诗、漏格诗及绘画等形式,将要表达的真正意思或“密语”隐藏在诗文或画卷中特定位置的记载,一般人只注意诗或画的表面意境,而不会去注意

或很难发现隐藏其中的“话外之音”。比如，我画蓝江水悠悠，爱晚亭上枫叶愁。秋月溶溶照佛寺，香烟袅袅绕轻楼（藏头诗）。

最早的密码技术，来源于公元前 2000 年。希伯来人的一种加密方法是把字母表调换顺序，这样的字母表的每一个字母就被映射成调换顺序后的字母表中的另一个字母，这种加密方法被称为 atbash。例如，单词 security 就被加密成 hvxfirgb，这是一种代换密码，因为一个字母被另一个字母所代替。这种代换密码被称为单一字母替换法，因为它只使用一个字母表，而其他加密方法一次用多个字母表，则成为多字母替换法。公元前 400 年，斯巴达人就发明了“塞塔式密码”，即把长条纸螺旋形地斜绕在一个多棱棒上，将文字沿棒的水平方向从左到右书写，写一个字旋转一下，写完一行再另起一行从左到右写，直到写完。解下来后，纸条上的文字消息杂乱无章、无法理解，这就是密文，但将它绕在另一个同等尺寸的棒子上后，就能看到原始的消息。

后来，朱丽叶斯·凯撒发明了一种近似于 atbash 替换字母的方法。当时，没多少人能够第一时间读懂，这种方法提供了较高的机密性。中世纪，欧洲人在不断利用新的方法、新的工具和新的实践优化自己的加密方案。在 19 世纪晚期，密码学已经被广泛地用作军事上的通信方法。

2. 古典密码（机械阶段）

古典密码的加密方法一般是文字置换，使用手工或机械变换的方式实现。古典密码系统已经初步体现出近代密码系统的雏形，它比古代加密方法复杂，其变化较小。随着机械和电子技术的发展，电报和无线通信的出现，加密装置得到了突飞猛进的提高，转子加密机是军事密码学上的一个里程碑，这种加密机是在机器内用不同的转子来替换字母，它提供了很高的复杂性，从而很难攻破。

德国的 Enigma 机是历史上最著名的加密机，如图 3-1 所示。这种机器有三个转子、一个线路连接板和一个反转转子。在加密过程开始之前，消息产生者将 Enigma 机配置成初

图 3-1　Enigma 机

始设置，操作员把消息的第一个字母输入加密机，加密机用另一个字母来代替并把这个字母显示给操作员看。它的加密机制是：通过把转子旋转预定的次数用另一个不同的字母来代替原来的字母。因此，如果操作员把 T 作为第一个字符敲入机器中，Enigma 机可能会把 M 作为密文，操作员把就字母 M 写下来，然后他可以加快转子的速度再输入下一个字符，每加密一个字符操作员就加快转子的速度作为一个新的设置。继续这样下去，直到整个消息被加密。然后，加密的密文通过电波传输，大部分情况是传到潜水艇。这种对每个字母有选择性地替换依赖于转子装置，因此这个过程的关键和秘密的部分（密钥）在于在加密和解密的过程中操作员是怎样加速转子的。两端的操作员需要知道转子的速度增量顺序以使得德国情报部门能够正确地通信。尽管 Enigma 机的装置在当时非常复杂，但还是被一组波兰密码学家攻破，从而使得英国知道了德国的进攻计划和军事行动。有人说，Enigma 机的破译使第二次世界大战缩短了两年。

3. 近代密码（计算机阶段）

前面介绍了古代加密方法和古典密码，它们的研究还称不上是一门科学。直到 1949 年香农发表了一篇题为“保密系统的通信理论”的著名论文，该文首先将信息论引入了密码，从而把已有数千年历史的密码学推向了科学的轨道，奠定了密码学的理论基础。该文利用数学方法对信息源、密钥源、接收和截获的密文进行了数学描述和定量分析，提出了通用的密钥密码体制模型。

需要提出的是，由于受历史的局限，20 世纪 70 年代中期以前的密码学研究基本上是秘密地进行，而且主要应用于军事和政府部门。密码学的真正蓬勃发展和广泛的应用是从 20 世纪 70 年代中期开始的。1977 年美国国家标准局颁布了数据加密标准 DES 用于非国家保密机关。该系统完全公开了加密、解密算法。此举突破了早期密码学的信息保密的单一目的，使得密码学得以在商业等民用领域的广泛应用，从而给这门学科以巨大的生命力。

在密码学发展的进程中的另一件值得注意的事件是，在 1976 年，美国密码学家迪菲和赫尔曼在一篇题为“密码学的新方向”一文中提出了一个崭新的思想，不仅加密算法本身可以公开，甚至加密用的密钥也可以公开。但这前不意味着保密程度的降低。因为如果加密密钥和解密密钥不一样。而将解密密钥保密就可以。这就是著名的公钥密码体制。若存在这样的公钥体制，就可以将加密密钥像电话簿一样公开，任何用户当它想经其他用户传送一加密信息时，就可以从这本密钥簿中查到该用户的公开密钥，用它来加密，而接收者能用只有它所具有的解密密钥得到明文。任何第三者不能获得明文。1978 年，由美国麻省理工学院的里维斯特，沙米尔和阿德曼提出了 RSA 公钥密码体制，它是第一个成熟的、迄今为止理论上最成功的公钥密码体制。它的安全性是基于数论中的大整数因子分解。该问题是数论中的一个困难问题，至今没有有效的算法，这使得该体制具有较高的保密性。

按照人们对密码的一般理解，密码是用于将信息加密而不易破译，但在现代密码学中，除了信息保密外，还有另一方面的要求，即信息安全体制还要能抵抗对手的主动攻击。所谓主动攻击指的是攻击者可以在信息通道中注入他自己伪造的消息，以骗取合法接收者的相信。主动攻击还可能窜改信息，也可能冒名顶替，这就产生了现代密码学中的认证体制。该体制的目的就是保证用户收到一个信息时，他能验证消息是否来自合法的发送者，同时还能验证该信息是否被窜改。在许多场合中，如电子汇款，能对抗主动攻击的认证体制甚至比信

息保密还重要。

在密码学的发展过程中，数学和计算机科学至关重要。数学中的许多分支如数论、概率统计、近世代数、信息论、椭圆曲线理论、算法复杂性理论、自动机理论、编码理论等都可以在其中找到各自的位置。

密码形成一门新的学科是受计算机科学蓬勃发展刺激和推动的结果。快速电子计算机和现代数学方法一方面为加密技术提供了新的概念和工具，另一方面也给破译者提供了有力武器。计算机和电子学时代的到来给密码设计者带来了前所未有的自由，他们可以轻易地摆脱原先用铅笔和纸进行手工设计时易犯的错误，也不用再面对用电子机械方式实现的密码机的高额费用。总之，利用电子计算机可以设计出更为复杂的密码系统。

3.1.2 密码系统的概念

密码系统又称为密码体制，是指能完整地解决信息安全中的机密性、数据完整性、认证、身份识别及不可抵赖等问题中的一个或几个的一个系统。其目的是人们能够使用不安全信道进行安全的通信，如图 3-2 所示。

图 3-2 密码系统

一个密码系统由算法以及所有可能的明文、密文和密钥组成。因此，一个完整的密码体制要包括如下五个要素$\{M,C,K,E,D\}$：

(1) M，明文(Plain-text)是明文的有限集，称为明文空间；

(2) C，密文(Cipher-text)是密文的有限集，称为密文空间，是对明文变换的结果；

(3) K，密钥(Key)是一切可能的密钥构成的有限集，称为密钥空间；

(4) E，加密算法(Encrypt)是一组含有参数的变换；

(5) D，解密算法(Decrypt)加密的逆变换。

密码体制的设计要求应符合早在 1883 年由科克霍夫斯(A. Kerchoffs)提出的一个重要原则：密码系统中的算法即使为密码分析者所知，也无助于用来推导出明文和密文。

密码体系的加密过程描述：$C= K_E(M)$

密码体系的解密过程描述：$M= K_D(C)$

3.1.3 密码的分类

从不同的角度根据不同的标准，可以把密码分成若干类。

1. 按应用技术或历史发展阶段划分

(1) 手工密码。以手工完成加密操作或者以简单器具辅助操作的密码，称为手工密码。第一次世界大战前主要是这种操作形式的密码。

(2) 机械密码。以机械密码机或电动密码机来完成加解密操作的密码，称为机械密码。

这种密码从第一次世界大战出现到第二次世界大战中得到普遍应用。

(3) 电子机内乱密码。通过电子电路以严格的程序进行逻辑运算,以少量制乱元素生产大量的加密乱数,因为其制乱是在加解密过程中完成的而无须预先制作,所以称为电子机内乱密码。从20世纪50年代末期出现到20世纪70年代被广泛应用。

(4) 计算机密码。以计算机软件编程进行算法加密为特点,适用于计算机数据保护和网络通信等广泛用途的密码。

2. 按保密程度划分

(1) 理论上保密的密码。不管获取多少密文和有多大的计算能力,对明文始终不能得到唯一解的密码,称为理论上保密的密码,也称理论不可破的密码。如客观随机一次一密的密码就属于这种类型。

(2) 实际上保密的密码。在理论上可破,但在现有客观条件下,无法通过计算来确定唯一解的密码,称作实际上保密的密码。

(3) 不保密的密码。在获取一定数量的密文后可以得到唯一解的密码,叫作不保密密码。如早期的单表代替密码,后来的多表代替密码,以及明文加少量密钥等密码,现在都成为不保密的密码。

3. 按密钥方式划分

(1) 对称式密码。收发双方使用相同密钥的密码,称作对称式密码。传统的密码都属此类。

(2) 非对称式密码。收发双方使用不同密钥的密码,称作非对称式密码。如现代密码中的公共密钥密码就属此类。

4. 按明文形态划分

(1) 模拟型密码。用以加密模拟信息,如对动态范围之内,连续变化的语音信号加密的密码,称作模拟式密码。

(2) 数字型密码。用于加密数字信息,对两个离散电平构成0、1二进制关系的电报信息加密的密码称作数字型密码。

5. 按编制原理划分

可分为移位、代替和置换三种以及它们的组合形式。古今中外的密码,不论其形态多么繁杂,变化多么巧妙,都是按照这三种基本原理编制出来的。移位、代替和置换这三种原理在密码编制和使用中相互结合,灵活应用,形成了各种不同的密码算法。

3.2 常用加密技术

加密技术是对信息进行编码和解码的技术,编码是把原来可读信息(又称明文)译成代码形式(又称密文),其逆过程就是解码(解密)。常用加密技术主要分为对称加密算法和非对称加密算法。

3.2.1 对称加密算法

对称加密算法(Symmetric Algorithm)也称为传统密码算法,指加密和解密使用相同密钥的加密算法,就是加密密钥能够从解密密钥中推算出来,同时解密密钥也可以从加密密钥中推算出来。而在大多数的对称算法中,加密密钥和解密密钥是相同的,所以也称这种加密算法为秘密密钥算法或单密密钥算法。对称算法的安全性依赖于密钥的保密,泄漏密钥就意味着任何人都可以对他们发送或接收的消息解密,所以密钥的保密性对通信性至关重要。此外,每对用户每次使用对称加密算法时,都需要使用其他人不知道的唯一密钥,这会使得发信双发所拥有的密钥数量成几何级数增长,密钥管理成为用户的负担。对称加密算法在分布式网络系统上使用较为困难,主要是因为密钥管理困难,使用成本较高。在计算机专网系统中广泛使用的对称加密算法有 DES、IDEA 和 AES。

1. DES 算法

美国国家标准局(NBS)于 1977 年公布了由 IBM 公司研制的一种加密算法,并批准把它作为非机要部门使用的数据加密标准(Data Encryption Standard,DES)。自从公布以来,它一直超越国界成为国际上商用保密通信和计算机通信的最常用的加密算法。当时规定 DES 的使用期为 10 年。后来美国政府宣布延长它的使用期,其原因大概有两条:一是 DES 尚未受到严重的威胁;二是一直没有新的数据加密标准问世。DES 超期服役了很长时间,在国际通信保密的舞台上活跃了 20 年。

DES 是 1972 年美国 IBM 公司研制的对称密码体制加密算法。明文按 64 位进行分组,密钥长 64 位,密钥事实上是 56 位参与 DES 运算(第 8、16、24、32、40、48、56、64 位是校验位,使得每个密钥都有奇数个 1)分组后的明文组和 56 位的密钥按位替代或交换的方法形成密文组的加密方法。

1) DES 工作的基本原理

入口参数有三个:key、data、mode。key 为加密解密使用的密钥,data 为加密解密的数据,mode 为其工作模式。当模式为加密模式时,明文按照 64 位进行分组,形成明文组,key 用于对数据加密,当模式为解密模式时,key 用于对数据解密。实际运用中,密钥只用到了 64 位中的 56 位,这样才具有高的安全性。DES 工作的基本原理如图 3-3 所示。

图 3-3 DES 基本原理

2) DES 算法的主要流程

DES 算法把 64 位的明文输入块变为 64 位的密文输出块,它所使用的密钥也是 64 位。整个算法的主流程如图 3-4 所示。DES 算法大致可以分成四个部分:初始置换、迭代过程和逆置换,迭代过程中又涉及置换表、函数 f、S 盒以及子密钥生成等。

(1) 置换规则表。

初始置换和逆置换均是按照一张置换表的置换规则进行置换。初始置换主要有输入的 64 位数据按 IP 置换表进行重新组合,并把输出分为 L_0、R_0 两部分,每部分各长 32 位,其 IP

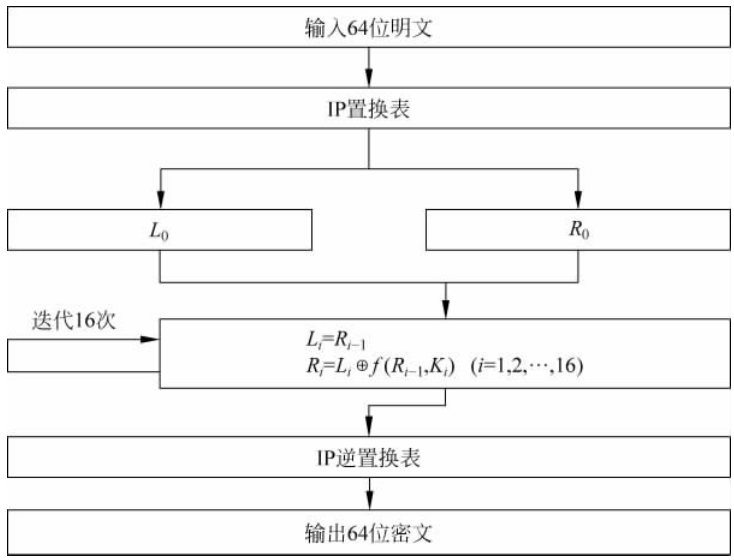

图 3-4　DES 算法流程图

置换表如表 3-1 所示。

表 3-1　IP 置换表

58	50	12	34	26	18	10	2	60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6	64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1	59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5	63	55	47	39	31	23	15	7

即将输入的 58 位换到第 1 位，第 50 位换到第 2 位，以此类推，最后一位是原来的第 7 位。L_0、R_0 则是换位输出后的两部分，L_0 是输出的左 32 位，R_0 是右 32 位。例如，设置换前的输入值为 $D_1D_2D_3\cdots D_{64}$，则经过初始置换后的结果为 $L_0=D_{58}D_{50}\cdots D_8$；$R_0=D_{57}D_{49}\cdots D_7$。

经过 16 次迭代运算后，得到 L_{16}、R_{16}，将此作为输入，进行逆置换，即得到密文输出。逆置换正好是初始置换的逆运算。例如，第 1 位经过初始置换后，处于第 40 位，而通过逆置换 IP-1，又将第 40 位换回到第 1 位，其逆置换 IP-1 规则如表 3-2 所示。

表 3-2　IP-1 逆置换表

40	8	48	16	56	24	64	32	39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30	37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28	35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26	33	1	41	9	49	17	57	25

（2）迭代过程。

在 16 轮迭代的过程中，将每轮 64 比特的输入分成 32 比特的左右两半，分别记为 L 和

R，每轮变换可用下列公式表示：

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

如上述公式所示，第 i 轮迭代的左半部分直接即为第 $i-1$ 轮的右半部分，而第 i 轮右半部分为第 $i-1$ 轮左半部分异或 $f(R_{i-1}, K_i)$。

(3) 函数 f。

函数 f 有两个输入：32 位的 R_{i-1}和 48 位 K_i，f 函数的处理流程如图 3-5 所示。

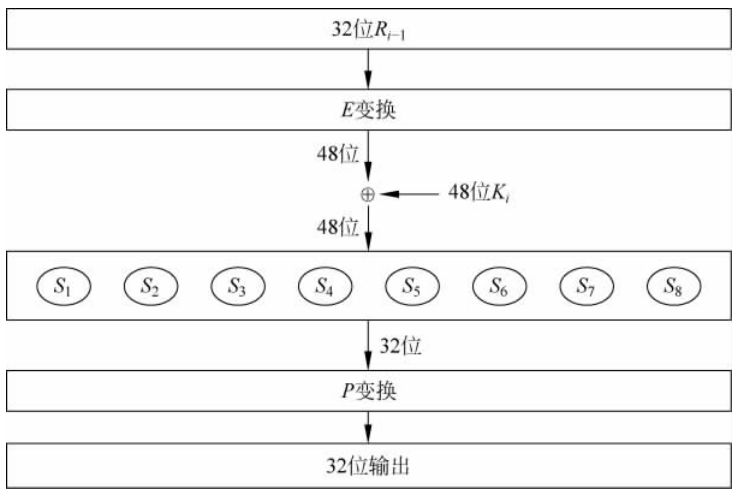

图 3-5 f 函数的处理流程图

(4) 换位表。

图 3-5 中 E 变换的算法是从 R_{i-1}的 32 位中选取某些位，构成 48 位。即 E 将 32 比特扩展变换为 48 位，变换规则根据 E 置换表，如表 3-3 所示。

表 3-3 E 置换表

32	1	2	3	4	5	4	5	6	7	8	9	8	9	10	11
12	13	12	13	14	15	16	17	16	17	18	19	20	21	20	21
22	23	24	25	24	25	26	27	28	29	28	29	30	31	32	1

图 3-5 中 P 变换的算法是从 S 盒的输出作为 P 变换的输入，P 的功能是对输入进行置换，P 置换表如表 3-4 所示。

表 3-4 P 置换表

16	7	20	21	29	12	28	17	1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9	19	13	30	6	22	11	4	25

(5) 子密钥 K_i。

假设密钥为 K，长度为 64 位，但是其中第 8、16、24、32、40、48、64 用作奇偶校验位，实际上密钥长度为 56 位。K 的下标 i 的取值范围是 1～16，用 16 轮变换来构造。

子密钥 K_i 的构造过程如图 3-6 所示。

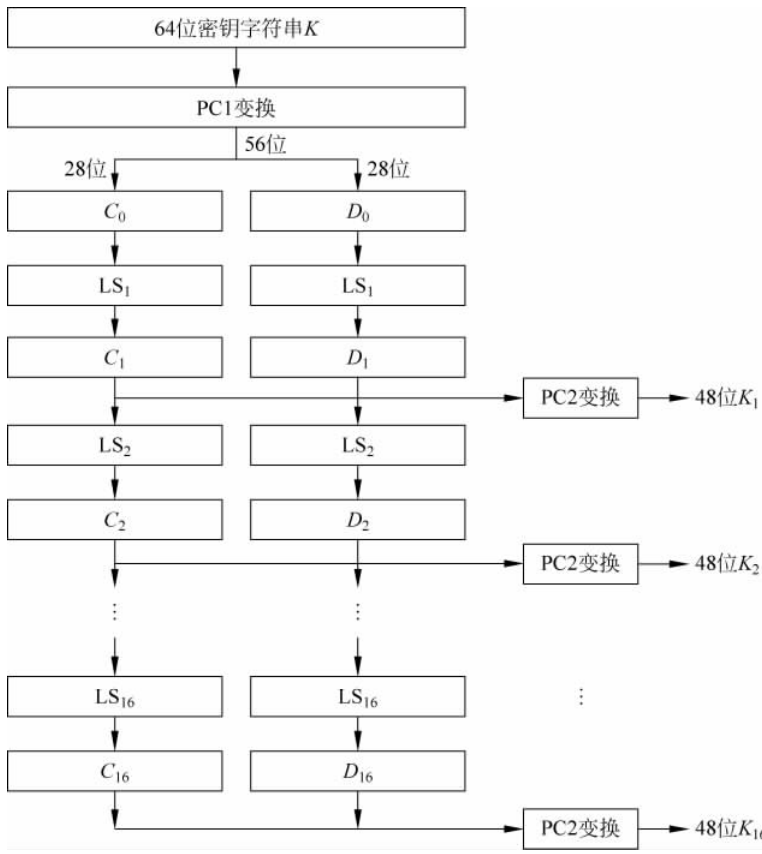

图 3-6　子密钥 K_i 生成过程

首先，对于给定的密钥 K，应用 PC1 变换进行选位，选定后的结果是 56 位，设其前 28 位为 C_0，后 28 位为 D_0。PC1 选位如表 3-5 所示。

表 3-5　PC1 变换表

57	49	41	33	25	17	9	1	58	50	42	34	26	18
10	2	59	51	43	35	27	19	11	3	60	52	44	36
63	55	47	39	31	23	15	7	62	54	46	38	30	22
14	6	61	53	45	37	29	21	13	5	28	20	12	4

第一轮：对 C_0 作左移 LS1 得到 C_1，对 D_0 作左移 LS1 得到 D_1，对 C_1D_1 应用 PC2 进行选位，得到 K_1。其中 LS1 是左移的位数，如表 3-6 所示。

表 3-6　LS 左移表

1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

表 3-6 中的第一列是 LS1，第二列是 LS2，以此类推。左移的原理是所有二进位向左移

动，原来最右边的比特位移动到最左边。其中 PC2 如表 3-7 所示。

表 3-7 PC2 变换表

14	17	11	24	1	5	3	28	15	6	21	10
23	19	12	4	26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40	51	45	33	48
44	49	39	56	34	53	46	42	50	36	29	32

第二轮：对 C_1、D_1 作左移 LS2 得到 C_2 和 D_2，进一步对 C_2、D_2 应用 PC2 进行选位，得到 K_2。如此继续，分别得到 K_3，K_4，…，K_{16}。

（6）S 盒的工作原理。

S 盒以 6 位作为输入，而以 4 位作为输出，现在以 S_1 为例说明其过程。假设输入为 $A=a1a2a3a4a5a6$，则 $a2a3a4a5$ 所代表的数是 0～15 之间的一个数，记为：$k=a2a3a4a5$；由 $a1a6$ 所代表的数是 0～3 间的一个数，记为 $h=a1a6$。在 S_1 的 h 行，k 列找到一个数 B，B 在 0～15 之间，它可以用 4 位二进制表示，为 $B=b1b2b3b4$，这就是 S_1 的输出，如图 3-7 所示。

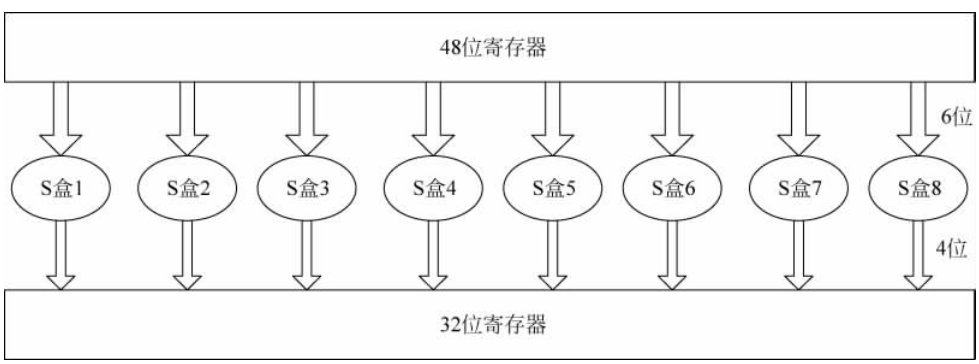

图 3-7 S 盒变换

如 6 位输入的第 1 和第 6 位组合构成了 2 位二进制数，可表示十进制数 0～3，它对应着表中的一行；6 位输入的第 2 到第 5 位组合构成了 4 位二进制数，可表示十进制数 0～15，它对应着表中的一列。假设 S_1 盒的 6 位输入是 110100，其第 1 位和第 6 位组合为 10，它对应 S_1 盒的第 2 行；中间 4 位组合为 1010，它对应 S_1 盒的第 10 列。S_1 盒的第 2 行第 10 列的数是 9，其二进制数为 1001（行和列的计数均从 0 开始而非从 1 开始）。1001 即为输出，则 1001 就代替了 110100。

DES 算法的解密过程是一样的，区别仅仅在于第一次迭代时用子密钥 K_{15}，第二次 K_{14}，最后一次用 K_0，算法本身并没有任何变化。DES 的算法是对称的，既可用于加密又可用于解密。

3）DES 算法的安全性

DES 算法是安全性比较高的一种算法，目前只有一种方法可以破解该算法，那就是穷举法。采用 64 位密钥技术，实际只有 56 位有效，8 位用来校验。譬如，有这样的一台 PC，

它能每秒计算一百万次，那么对于256位空间，要穷举它的时间为2285年，所以这种算法还是比较安全的一种算法。

2. 三重DES(3DES)

DES的主要密码学缺点就是密钥长度较短。为了解决使用DES技术56位时密钥日益减弱的强度问题，其办法之一是采用三重DES算法(Triple DES或3DES)，即使用两个独立密钥 K_1 和 K_2 对明文运行DES算法三次，得到112位有效密钥强度，若对3DES的穷举攻击需要2112次，而不是DES的264次。其算法的步骤如下：

(1) 用密钥 K_1 进行DES加密；

(2) 用步骤(1)的结果使用密钥 K_2 进行DES解密；

(3) 用步骤(2)的结果使用密钥 K_1 进行DES加密。

这个过程称为EDE，因为它是由加密-解密-加密(Encrypt Decrypt Encrypt)步骤组成的。在EDE中，中间步骤是解密，所以，可以使 $K_1 = K_2$ 来用三重DES方法执行常规的DES加密。

三重DES的缺点是时间开销较大，三重DES的时间是DES算法的3倍。但从另一方面看，三重DES的112位密钥长度在可以预见的将来可认为是适合的。

DES被认为是安全的，这是因为要破译它可能需要尝试256位密钥直到找到正确的密钥。

3. IDEA加密

1) IDEA加密算法概述

国际数据加密算法(International Data Encryption Algorithm，IDEA)是瑞士的James Massey、Xuejia Lai等人提出的加密算法，在密码学中属于数据块加密算法(Block Cipher)类。IDEA使用长度为128位的密钥，数据块大小为64位。从理论上讲，IDEA属于“强”加密算法，至今还没有出现对该算法的有效攻击算法。

早在1990年，Xuejia Lai等人在EuroCrypt'90年会上提出了分组密码建议(Proposed Encryption Standard，PES)。在EuroCrypt'91年会上，Xuejia Lai等人又提出了PES的修正版IPES(Improved PES)。目前IPES已经商品化，并改名为IDEA。IDEA已由瑞士的Ascom公司注册专利，以商业目的使用IDEA算法必须向公司申请许可。

这种算法是在DES算法的基础上发展出来的，类似三重DES。发展IDEA也是因为感到DES具有密钥太短等缺点，已经过时。IDEA的密钥为128位，这么长的密钥在今后若干年内应该是安全的。类似于DES，IDEA算法也是一种数据块加密算法，它设计了一系列加密轮次，每轮加密都使用从完整的加密密钥中生成的一个子密钥。与DES的不同处在于，它采用软件实现和采用硬件实现同样快速。由于IDEA是在美国之外提出并发展起来的，避开了美国法律上对加密技术的诸多限制，因此，有关IDEA算法和实现技术的书籍都可以自由出版和交流，可极大地促进IDEA的发展和完善。

目前IDEA在工程中已有大量应用实例，PGP(Pretty Good Privacy)就使用IDEA作为其分组加密算法；安全套接字(Secure Socket Layer，SSL)也将IDEA包含在其加密算法库SSLRef中；IDEA算法专利的所有者Ascom公司也推出了一系列基于IDEA算法的安全

产品，包括基于 IDEA 的 Exchange 安全插件、IDEA 加密芯片、IDEA 加密软件包等。IDEA 算法的应用和研究正在不断走向成熟。

2) IDEA 算法原理

IDEA 是一种由 8 个相似圈(Round)和一个输出变换(Output Transformation)组成的迭代算法。IDEA 的每个圈都包含三种基本运算，即乘法、加法和异或。在加密之前，IDEA 通过密钥扩展(Key Expansion)将 128bit 的密钥扩展为 52Byte 的加密密钥(Encryption Key，EK)，然后由 EK 计算出解密密钥(Decryption Key，DK)。EK 和 DK 分为 8 组半密钥，每组长度为 6Byte，前 8 组密钥用于 8 圈加密，最后半组密钥(4Byte)用于输出变换。IDEA 的加密过程和解密过程是一样的，只不过使用不同的密钥(加密时用 EK，解密时用 DK)。

密钥扩展的过程如下：

(1) 将 128bit 的密钥作为 EK 的前 8byte；

(2) 将前 8byte 循环左移 25bit，得到下一 8byte，将这个过程循环 7 次；

(3) 在第 7 次循环时，取前 4byte 作为 EK 的最后 4byte；

(4) 至此 52byte 的 EK 生成完毕。

3) IDEA 算法的安全性

IDEA 算法的密钥为 128bits(DES 的密钥为 56bits)，设计者尽最大努力使该算法不受差分密码分析的影响，数学家已证明 IDEA 算法在其 8 圈迭代的第 4 圈之后便不受差分密码分析的影响了。假定穷举法攻击有效的话，那么即使设计一种每秒钟可以试验 10 亿个密钥的专用芯片，并将 10 亿片这样的芯片用于此项工作，仍需 1013 年才能解决问题；另一方面，若用 1024 片这样的芯片，有可能在一天内找到密钥，不过人们还无法找到足够的硅原子来制造这样一台机器。目前，尚无一篇公开发表的试图对 IDEA 进行密码分析的文章。因此，就现在来看应当说 IDEA 是非常安全的。并且，IDEA 算法比 RSA 算法快得多，又比 DES 算法要相对安全得多。

3.2.2 非对称加密算法

美国斯坦福大学的两名学者 W. Diffie 和 M. Hellman 于 1976 年在 IEEE Transactions on Information Theory 杂志上发表了文章 New Direction in Cryptography，提出了“公开密钥密码体制”的概念，开创了密码学研究的新方向。公开密钥密钥体制的产生主要有两个方面的原因：一是由于对称密钥密码体制的密钥分配问题，二是由于对数字签名的需求。

非对称密码系统的解密密钥与加密密钥是不同的，一个称为公开密钥，另一个称为私人密钥(或秘密密钥)，因此这种密码体系也称为公钥密码体系。公钥密码体制的算法中最著名的代表是 RSA 系统，此外还有椭圆曲线、背包密码、McEliece 密码、Diffe-Hellman、Rabin 和 ElGamal 算法等。

1. RSA 算法

RSA 算法是最著名、应用最广的公钥密码算法。它是在 1978 年由 Rivest、Shamir 和 Adleman 三个人共同提出的，并以三个发明者的名字的首字母命名。RSA 的安全性取决于大模数的因子分解的困难性，其公开密钥和私人密钥是一对大素数(100 到 200 位的十进制

数或更大)的函数,从一个公开密钥和密文中恢复出明文的难度等同于分解两个大素数之积的难度。从严格的技术角度上来说这是不正确的,在数学上至今还没有证明分解模数就是攻击 RSA 的最佳方法,也未能证明分解大整数就是 NP 问题。事实的情况是,大整数因子分解问题过去几百年来一直是令数学家头疼而又未能有效解决的世界性难题。人们设想了一些非因子分解的途径来攻击 RSA 算法,但这些方法都不比分解模数来得容易。因此,严格地说,RSA 的安全性基于求解其单向函数的逆的困难性。RSA 单向函数求逆的安全性没有真正的因子分解模数的安全性高,而且目前人们也无法证明这两者是等价的。许多研究人员都试图改进 RSA 算法使它的安全性等价于因子分解模数。

RSA 是最具代表性的公钥密码算法,可能也是最知名和最古老的公钥密码算法。由于算法完善(既可以用于数据加密,又可用于数字签名),安全性良好,易于理解和实现,RSA 已经成为了一种应用极为广泛的公钥密码算法。

RSA 算法的思路如下:

1) 密钥生成

(1) 系统产生两个大素数 p,q(保密)。为了获得最大程度的安全性,选两数的长度一样。

(2) 计算模数 $n=p\times q$(公开),欧拉函数 $\Phi(n)=(p-1)\times(q-1)$(保密)。

(3) 随机选取加密密钥 e,使 e 和 $\Phi(n)$ 互素,即满足:$0<e<\Phi(n)$ 且 $\gcd(e,\Phi(n))=1$。

(4) 用欧几里得(Euclidean)扩展算法计算解密密钥 d,d 满足 $e\times d\equiv 1 \bmod \Phi(n)$,即 $d=e-1 \bmod \Phi(n)$。

(5) e 和 n 为公开密钥,d 是私人密钥。两个大数 p 和 q 应该立即丢弃,不让任何人知道。一般选择公开密钥 e 比私人密钥 d 小。最常选用的 e 值有 3 个,即 3、17、65 537。

2) RSA 加密和解密过程

加密消息时,首先将明文分组并数字化,每个数字化分组明文的长度不大于 n(采用二进制数,选到小于 n 的 $2d$ 的最大次幂),设 mi 表示消息分组,ci 表示加密后的密文,它与 mi 具有相同的长度。

对每个明文分组 m 依次进行加解密运算:

(1) 加密运算:使用公钥 e 和要加密的明文 m 进行 $\mathrm{ci}=\mathrm{mi}^e \pmod n$ 运算即得密文。

(2) 解密运算:使用私钥 d 和要解密的密文 c 进行 $\mathrm{mi}=\mathrm{ci}^d \pmod n$ 运算即得明文。RSA 的实现过程如图 3-8 所示。

下面举例说明 RSA 算法的实现过程。

① 取两个质数 $p=11$,$q=13$,p 和 q 的乘积为 $n=p\times q=143$;

② 算出另一个数 $\Phi(n)=(p-1)\times(q-1)=120$;

③ 再选取一个与 $\Phi(n)=120$ 互质的数,例如 $e=7$,则公开密钥$=(n,e)=(143,7)$;

④ 对于这个 e 值,可以算出其逆:$d=103$。因为 $e\times d=7\times 103=721$,满足 $e\times d \bmod \Phi(n)=1$;即 $721 \bmod 120=1$ 成立,则秘密密钥$=(n,d)=(143,103)$。

设张小姐需要发送机密信息(明文)$m=85$ 给李先生,她已经从公开媒体得到了李先生的公开密钥$(n,e)=(143,7)$,于是她算出加密值:$c=me \bmod n=857 \bmod 143=123$ 并发送给李先生。

李先生在收到密文 $c=123$ 后,利用只有他自己知道的秘密密钥计算:$m=cd \bmod n=$

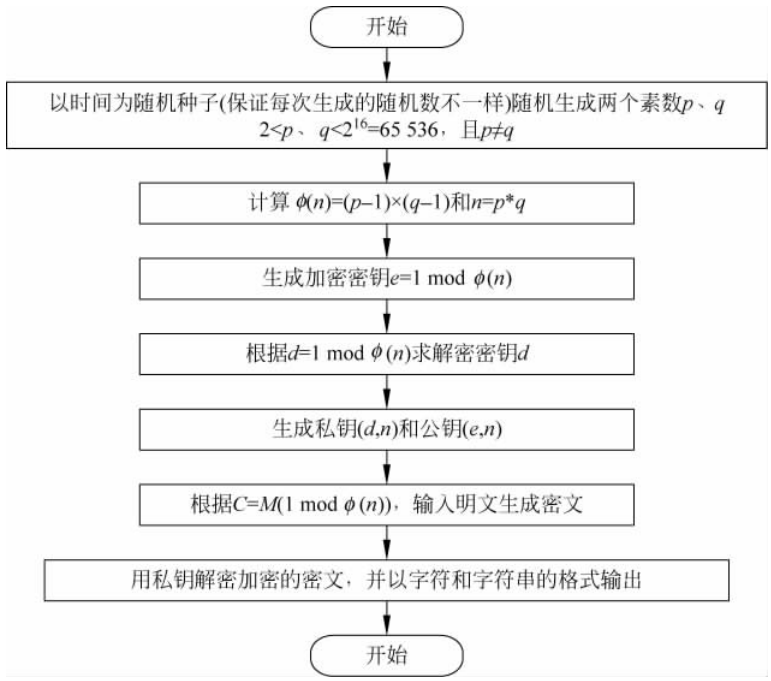

图 3-8 RSA 算法的实现过程

123103 mod 143＝85，所以，李先生可以得到张小姐发给他的真正的信息 $m=85$，实现了解密。

3）RSA 算法的特点及应用

RSA 算法具有密钥管理简单（网上每个用户仅需保密一个密钥，且不需配送密钥）、便于数字签名、可靠性较高（取决于分解大素数的难易程度）等优点，但也具有算法复杂、加密/解密速度慢、难于用硬件实现等缺点。因此，公钥密码体制通常被用来加密关键性的、核心的、少量的机密信息，而对于大量要加密的数据通常采用对称密码体制。

RSA 算法的安全性建立在难于对大整数提取因子的基础上，已知的证据都表明大整数因式分解问题是一个极其困难的问题。但是，随着分解大整数方法的进步及完善、计算机速度的提高以及计算机网络的发展，要求作为 RSA 加密/解密安全保障的大整数越来越大。

RSA 算法的保密性，取决于 dui 大素数因式分解的时间。假定用 106 次/秒的计算机进行运算，用最快的公式分解 $n=100$ 位十进制数要用 74 年，分解 200 位数要用 3.8×109 年。可见，当 n 足够大时（p 和 q 各为 100 位时，n 为 200 位），对其进行分解是很困难的。可以说，RSA 的保密强度等价于分解 n 的难易程度。

RSA 算法为公用网络上信息的加密和鉴别提供了一种基本的方法。他通常是先生成一对 RSA 密钥，其中之一是保密密钥，由用户保存；另一个为公开密钥，可对外公开，甚至可在网络服务器中注册。

2. 椭圆曲线密码算法

1）椭圆曲线密码概述

椭圆曲线密码学（Elliptic Curve Cryptography，ECC）是基于椭圆曲线数学的一种公钥

密码的方法。椭圆曲线在密码学中的使用是在 1985 年由 Neal Koblitz 和 Victor Miller 分别独立提出的。ECC 的主要优势是在某些情况下它比其他的方法使用更小的密钥(如 RSA)提供相当的或更高等级的安全。ECC 的另一个优势是可以定义群之间的双线性映射,基于 Weil 对或 Tate 对。双线性映射已经在密码学中发现了大量的应用,例如基于身份的加密。不过它的一个缺点是加密和解密操作的实现比其他机制花费的时间长。椭圆曲线密码学的许多形式稍微有所不同,所有的都依赖于被广泛承认的解决椭圆曲线离散对数问题的困难性上,对应有限域上椭圆曲线的群。

椭圆曲线是用三次方程来表示的,该方程与计算椭圆周长的方程相似,因而称为椭圆曲线。在 ECC 中,我们关心的是某种特殊形式的椭圆曲线,即定义在有限域上的椭圆曲线,椭圆曲线的吸引人之处在于提供了由“元素”和“组合规则”来组成群的构造方式,用这些群来构造密码算法具有完全相似的特性,且没有减少密码分析的分析量。

2) 椭圆曲线国际标准

椭圆曲线密码系统已经形成了若干国际标准,其涉及加密、签名、密钥管理等方面,包括:

(1) IEEE P1363:加密、签名、密钥协商机制。

(2) ANSI X9:椭圆曲线数字签名算法,即椭圆曲线密钥协商和传输协议。

(3) ISO/IEC:椭圆曲线 ElGamal 体制签名。

(4) IETF:椭圆曲线 DH 密钥交换协议。

(5) ATM Forum:异步传输安全机制。

(6) FIPS 186-2:美国政府用于保证其电子商务活动中的机密性和完整性。

3) 椭圆曲线技术实现

ECC 的技术实现可以分成 4 个层次:运算层、密码层、接口层和应用层。运算层最基础、最核心;应用层最接近用户。

(1) 运算层。

运算层的主要功能是提供密码算法的所有数论运算支持,包括大整数加、减、乘、除、模、逆、模幂等。运算层的实现效率将对整个密码系统的效率起决定性作用。因而运算层的编程工作是算法实现最核心、最基础,也是最艰巨的部分。

(2) 密码层。

密码层的主要功能是在运算层的支持上选择适当的密码体制,科学地、准确地、安全地实现密码算法。在相同的运算层的基础上,可以构建起多种密码体制。对于密码体制和具体结构的选择和实现是密码层的核心内容。最终,密码系统的安全性将决定于密码层的实现能力。在密码层中,为了支持公钥密码系统,通常必须提供 5 种操作,即生成密钥对、加密、解密、签名和验证签名。

(3) 接口层。

接口层的主要功能是对各种软硬件平台提供公钥密码功能支持。其工作重点在于对各种硬件环境的兼容、对各种操作系统的兼容、对各种高级语言的兼容、对多种应用需求兼容。其难点主要在于保持良好的一致性、可移植性、可重用性,以有限的资源换取应用层尽可能多的自由空间。

(4) 应用层。

应用层是最终用户所能接触到的唯一层面,它为用户提供应用功能和操作界面。应用功能包括交易、网络、文件、数据库、加解密、签名及验证等。操作界面包括图形、声音、指纹、键盘、鼠标等。

ECC 的实现效率一般表现为 ECC 公钥密码功能的效率。实现效率是被多种因素制约和影响的。下面列举了在实现 ECC 公钥密码功能效率。实现效率是被多种因素制约和影响的。下面列举了在实现 ECC 的过程中遇到的涉及 ECC 实现效率的方面。

① ECC 密码机制。众所周知,任何密码理论都必须在某种密码机制上实现才能完成密码功能(如加密、签名等)。同一种密码理论也可以运用于不同的密码机制上,而且它们的实现效率也不尽相同。我们在自行发明的、拥有自主知识产权的密码机制上实现 ECC,并且容易证明其安全性不低于其他常用密码体制,且效率更高。

② 安全前瞻性。由于公钥系统的安全性建立在数学的困难性上,因此在选择 ECC 参数时,不能一味地追求速度快,而是应该在理论上、实现上都要为安全性留出一定的余量,以保证在密码分析技术进步后,不致受到重大威胁。ECC 安全性的保障是要通过降低一定的效率来换取的。

③ 应用环境。应用环境是 ECC 软硬件实现的约束条件。硬件环境要求空间小、指令简单、高稳定性、低成本;软件环境要求兼容性好、可移植性好、易于维护升级。因此,从高端到低端,从高级语言到汇编、从系统到门电路设计,每个应用环境对 ECC 实现所提供的支持和约束都不相同。所以,ECC 实现效率也依应用环境而异。

④ 算法优化。算法优化始终都是提高效率的根本所在。对 ECC 实现算法的优化主要从这几个方面入手:对数学公式的变形和组合优化;在软件实现中,根据编译系统的特点、CPU 指令集的特点优化;在硬件实现中,根据硬件资源的具体特点优化。

3.3 密码技术的应用

网络安全系统的一个很重要方面是防止非法用户对系统的主动攻击,如伪造信息、篡改信息等。这种安全要求对实际网络系统的应用(如电子商务)是非常重要的。以下介绍的鉴别、数字签名、物联网认证与访问控制以及公钥基础设施等都是基于数据加密的应用技术。

3.3.1 鉴别技术

1. 基本概念

鉴别(authentication,也叫验证)是防止主动攻击的重要技术。鉴别的目的就是验证用户身份的合法性和用户间传输信息的完整性与真实性。

鉴别服务主要包括信息鉴别和身份验证两方面。信息鉴别和身份验证可采用数据加密技术、数字签名技术及其他相关技术来实现。

信息鉴别是为了确保数据的完整性和真实性,对信息的来源、时间性及目的地进行验证。信息鉴别过程通常涉及加密和密钥交换。加密可使用对称密钥加密、非对称密钥加密或两种加密方式的混合。信息经验证后表明,它在发送期间没有经过篡改,发送者经验证后

表明，他就是合法的发送者。

身份验证是验证进入网络系统者是否是合法用户，以防非法用户访问系统。身份验证的方式一般有用户口令验证、摘要算法验证、基于 PKI(公钥基础设施)的验证等。验证、授权和访问控制都与网络实体安全有关。

网络中的通信除需要进行消息的验证外，还需要建立一些规范的协议对数据来源的可靠性、通信实体的真实性加以认证，以防止欺骗、伪装等攻击。例如：A 和 B 是网络的两个用户，他们想通过网络先建立安全的共享密钥再进行保密通信，那么 A 如何确信自己正在和 B 通信而不是和 C 通信呢？这种通信方式为双向通信，因此此时的认证称为互相认证。类似地，对于单向通信来说，认证称为单向认证。

认证中心(Certificate Authority，CA)在网络通信认证技术中具有特殊的地位。例如，电子商务，认证中心是为了从根本上保障电子商务交易活动顺利进行而设立的，主要是解决电子商务活动中参与各方的身份、资信的认定，维护交易活动的安全。CA 是提供身份验证的第三方机构，通常由一个或多个用户信任的组织实体组成。例如，持卡人(客户)要与商家通信，持卡人从公开媒体上获得了商家的公开密钥，但无法确定商家不是冒充的(有信誉)，于是请求 CA 对商家认证。此时，CA 对商家进行调查、验证和鉴别后，将包含商家公钥的证书传给持卡人。同样，商家也可对持卡人进行验证，其过程为持卡人→商家；持卡人→CA；CA→商家。证书一般包含拥有者的标识名称和公钥，并且由 CA 进行数字签名。CA 的功能主要包括接收注册申请、处理、批准/拒绝请求和颁发证书。

在实际运作中，CA 也可由大家都信任的一方担当，例如，在客户、商家、银行三角关系中，客户使用的是由某个银行发的卡，而商家又与此银行有业务关系(有账号)。在此情况下，客户和商家都信任该银行，可由该银行担当 CA 角色，接受和处理客户证书和商家证书的验证请求。又如，对商家自己发行的购物卡，则可由商家自己担当 CA 角色。

2. 信息的验证

从概念上说，信息的签名就是用专用密钥对信息进行加密，而签名的验证就是用相对应的公用密钥对信息进行解密。但是，完全按照这种方式行事也有缺点。因为，同普通密钥系统相比，公用密钥系统的速度很慢，用公用密钥系统对长信息加密来达到签名的目的，并不比用公用密钥系统来达到信息保密的目的更有吸引力。

解决方案就是引入另一种普通密码机制，这种密码机制叫做信息摘要或散列函数。信息摘要算法从任意大小的信息中产生固定长度的摘要，而其特性是没有一种已知的方法能找到两个摘要相同的信息。这就意味着，虽然摘要一般要比信息小得多，但是可以在很多用途方面看作是与完整信息等同的。最常用的信息摘要算法叫做 MD5，可产生一个 128 位长的摘要。

使用信息摘要时，对信息签名的过程如下：

(1) 用户制作信息摘要。

(2) 信息摘要由发送者的专用密钥加密。

(3) 原始信息和加密信息摘要发送到目的地。

(4) 目的地接收信息，并使用与原始信息相同的信息摘要函数对信息制作其自己的信息摘要。

(5) 目的地对所收到的信息摘要进行解密。

(6) 目的地将制作的信息摘要同附有信息的信息摘要进行对比,如果相吻合,目的地就知道信息的文本与用户发送的信息文本是相同的,如果二者不吻合,则目的地知道原始信息已经被修改过。

这一过程还有另外一个长处,这个长处可取名为数字签名。由于只有用户知道私用密钥,因而只有用户能够制作加密的信息摘要。任何一个可以获取公用密钥的目的地都可弄清楚签名者的身份。这一技术可用于最流行的程序,用以保护包括 PGP 和 PEM(保密增强邮件)在内的电子邮件。

3. 身份验证

身份认证是在计算机网络中确认操作者身份的过程。身份认证可分为用户与主机间的认证和主机与主机之间的认证。用户与主机之间的认证可以基于如下一个或几个因素:用户所知道的东西,例如口令、密码等;用户拥有的东西,例如印章、智能卡(如信用卡等);用户所具有的生物特征,例如指纹、声音、视网膜、签字和笔迹等。

计算机网络世界中一切信息包括用户的身份信息都是用一组特定的数据来表示的,计算机只能识别用户的数字身份,所有对用户的授权也是针对用户数字身份的授权。如何保证以数字身份进行操作的操作者就是这个数字身份合法拥有者,也就是说保证操作者的物理身份与数字身份相对应,身份认证就是为了解决这个问题。作为防护网络资产的第一道关口,身份认证有着举足轻重的作用。

在真实世界,对用户的身份认证基本方法可以分为三种:

(1) 根据你所知道的信息来证明你的身份(what you know,你知道什么);

(2) 根据你所拥有的东西来证明你的身份(what you have,你有什么);

(3) 直接根据独一无二的身份特征来证明你的身份(who you are,你是谁),如指纹、面貌等;

在网络世界中的手段与真实世界中一致,为了达到更高的身份认证安全性,某些场景会将上面三种认证方法中的两种混合使用,即所谓的双因素认证。

进入电子信息社会,虽然有不少学者试图使用电子化生物唯一识别信息,但是出于其代价高、准确性低、存储空间大和传输速率低,不适合计算机读取和判断,只能作为辅助措施应用。而使用密码技术,特别是公钥密码技术,能够设计出安全性高的识别协议,受到人们的青睐。

过去人们采用通行字作为用户身份识别,通行字短、固定、规律性强、易暴露、安全性差。现在采用密码技术进行交互式询答,只有拥有正确密码的合法用户才能通过询答。目前已经用于身份认证的 IC 卡、数字证书、一次性口令等,它们都采用了密码技术。

3.3.2 数字签名技术

1. 基本概念

数字签名,又称为公钥数字签名、电子签章,是只有信息的发送者才能产生的别人无法伪造的一段数字串,这段数字串同时也是对信息的发送者发送信息真实性的一个有效证明。

数字签名是一种类似写在纸上的普通的物理签名，但是使用了公钥加密领域的技术实现，用于鉴别数字信息的方法。

目前的数字签名大多是建立在公开密钥体制基础上的，这是公开密钥加密技术的另一种重要应用，如基于 RSA 的公开密钥加密标准 PKCS、数字签名算法 DSA、PGP 加密软件等。1994 年美国标准与技术协会公布了数字签名标准，从而使公钥加密技术得到了广泛应用。目前，广泛应用的数字签名算法主要有三种，即 RSA 签名、DSS(数字签名标准)签名和 Hash 签名。这三种算法可单独使用，也可综合在一起使用。数字签名是通过密码算法对数据进行加密/解密变换实现的，用 DES 算法、RSA 算法都可实现数字签名。

用 RSA 或其他公开密钥密码算法的最大方便是没有密钥分配问题(网络越复杂、网络用户越多，其优点越明显)。因为公开密钥加密使用两个不同的密钥，其中有一个是公开的，另一个是保密的(私钥)。公开密钥可以保存在系统目录内、未加密的电子邮件中、电话号码簿或公告牌里，网上的任何用户都可获得公开密钥。而私有密钥是用户专用的，由用户本身持有，它可以对由公开密钥加密的信息进行解密。

一套数字签名通常定义两种互补的运算，一个用于签名，另一个用于验证。数字签名是非对称密钥加密技术与数字摘要技术的应用，其机制需要实现以下几个目的：

(1) 消息源认证：消息的接受者通过签名可以确信消息确实来自声明的发送者。

(2) 不可伪造：签名应是独一无二的，其他人无法假冒和伪造。

(3) 不可重用：签名是消息的一部分，不能被挪用到其他的文件上。

(4) 不可抵赖：签名者事后不能否认自己签过的文件。

DSS 数字签名是由美国国家标准化研究院和国家安全局共同开发的。由于 DSS 是由美国政府颁布实施的，只是一个签名系统，而且美国政府不提倡使用任何削弱政府窃听能力的加密软件，认为这才符合美国的国家利益，因此，DSS 主要用于与美国政府做生意的公司，其他公司则较少使用。

2. 单向散列函数

单向散列函数，又称为单向 Hash 函数、杂凑函数，就是把任意长的输入消息串变化成固定长的输出串且由输出串难以得到输入串的一种函数。这个输出串称为该消息的散列值。一般用于产生消息摘要，密钥加密等。

1) 一个安全的单向散列函数应该满足的几个条件

(1) 输入长度是任意的；

(2) 输出长度是固定的，根据目前的计算技术应至少取 128bits 长，以便抵抗攻击；

(3) 对每一个给定的输入，计算输出即散列值是很容易的；

(4) 给定散列函数的描述，找到两个不同的输入消息杂凑到同一个值是计算上不可行的，或给定杂凑函数的描述和一个随机选择的消息，找到另一个与该消息不同的消息使得它们杂凑到同一个值是计算上不可行的。

2) 常见单向散列函数(Hash 函数)

(1) MD5(Message Digest Algorithm 5)：是 RSA 数据安全公司开发的一种单向散列算法，MD5 被广泛使用，可以用来把不同长度的数据块进行暗码运算成一个 128 位的数据。

(2) SHA(Secure Hash Algorithm)这是一种较新的散列算法，可以对任意长度的数据

运算生成一个 160 位的数据。

(3) MAC(Message Authentication Code)：消息认证代码，是一种使用密钥的单向函数，可以用它们在系统上或用户之间认证文件或消息。HMAC(用于消息认证的密钥散列法)就是这种函数的一个例子。

(4) CRC(Cyclic Redundancy Check)：循环冗余校验码，CRC 校验由于实现简单，检错能力强，被广泛使用在各种数据校验应用中。占用系统资源少，用软硬件均能实现，是进行数据传输差错检测的一种很好的手段(CRC 并不是严格意义上的散列算法，但它的作用与散列算法大致相同，所以也归入此类)。

3. 数字签名过程

数字签名技术是将摘要信息用发送者的私钥加密，与原文一起传送给接受者。接受者只有用发送者的公钥才能解密被加密的摘要信息，然后用 HASH 函数对收到的原文产生一个摘要信息，与解密的摘要信息对比。如果相同，则说明收到的信息是完整的，在传输过程中没有被修改，否则说明信息被修改过，因此数字签名能够验证信息的完整性。

发送报文时，发送方用一个哈希函数从报文文本中生成报文摘要，然后用自己的私人密钥对这个摘要进行加密，这个加密后的摘要将作为报文的数字签名和报文一起发送给接收方，接受方首先用与发送方一样的哈希函数从接收到的原始报文中计算出报文摘要，然后再用发送方的公用密钥来对报文附加的数字签名进行解密，如果这两个摘要相同，那么接收方就能确认该数字签名是发送方的，如图 3-9 所示。

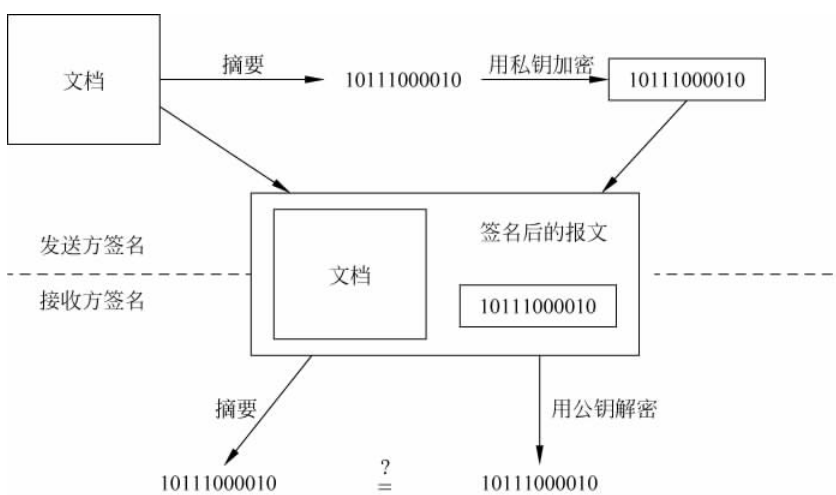

图 3-9 数字签名

数字签名有两方面的作用：一是能确定消息确实是由发送方签名并发出来的，因为别人假冒不了发送方的签名；二是数字签名能确定消息的完整性。因为数字签名的特点是它代表了文件的特征，文件如果发生改变，数字签名的值也将发生变化。不同的文件将得到不同的数字签名。一次数字签名涉及一个哈希函数、发送者的公钥、发送者的私钥。

4. 数字签名的原理特点

每个人都有两条“钥匙”(数字身份),其中一条钥匙只有她/他本人知道(密钥),另一条钥匙是公开的(公钥)。签名的时候用密钥,验证签名的时候用公钥。又因为任何人都可以落款声称她/他就是你,所以公钥必须由接受者信任的人(身份认证机构)来注册。注册后身份认证机构给用户发一个数字证书。对文件签名后,把此数字证书连同文件及签名一起发给接受者,接受者向身份认证机构求证是否真的是用你的密钥签发的文件。

在通信中使用数字签名一般基于以下原因:

1) 鉴权

公钥加密系统允许任何人在发送信息时使用公钥进行加密,数字签名能够让信息接收者确认发送者的身份。当然,接收者不可能百分之百确信发送者的真实身份,而只能在密码系统未被破译的情况下才能有理由确信。

鉴权的重要性在财务数据上表现得尤为突出。举个例子,假设一家银行将指令由它的分行传输到它的中央管理系统,指令格式是(a,b),其中 a 是账户的账号,而 b 是账户的现有金额。这时一位远程客户可以先存入 100 元,观察传输的结果,然后接二连三地发送格式为(a,b)的指令。这种方法被称作重放攻击。

2) 完整性

传输数据的双方都希望确认消息未在传输的过程中被修改。加密使得第三方想要读取数据十分困难,然而第三方仍然能采取可行的方法在传输的过程中修改数据。一个通俗的例子就是同形攻击:回想一下,还是上面的那家银行从它的分行向它的中央管理系统发送格式为(a,b)的指令,一个远程客户可以现存 100 元,然后拦截传输结果,再传输(a,b3),这样他就立刻变成百万富翁了。

3) 不可抵赖

在密文背景下,抵赖这个词指的是不承认与消息有关的举动(即声称消息来自第三方)。消息的接受方可以通过数字签名来防止所有后续的抵赖行为,因为接收方可以出示签名给别人看来证明信息的来源。

5. 数字签名与信息加密的区别

数字签名使用的是发送方的密钥对,是发送方用自己的私钥对摘要进行加密,接收方用发送方的公钥对数字签名解密,是一对多的关系,表明发送方公司的任何一个贸易伙伴都可以验证数字签名的真伪性;

密钥加密解密过程使用的是接收方的密钥对,是发送方用接收方的公钥加密,接收方用自己的私钥解密,是多对一的关系,表明任何拥有该公司公钥的人都可以向该公司发送密文,但只有该公司才能解密,其他人不能解密。

3.3.3　物联网认证与访问控制

认证指使用者采用某种方式来证明自己确实是自己宣称的某人,网络中的认证主要包括身份认证和消息认证。身份认证可以使通信双方确信对方的身份后交换会话密钥,消息认证主要是接收方希望能够保证其接收的消息确实来自真正的发送方。

在物联网的认证过程中，传感器网络的认证机制是重要的研究部分，无线传感器网络中的认证技术主要包括基于轻量级公钥的认证技术、预共享密钥的认证技术、随机密钥预分布的认证技术、利用辅助信息的认证和基于单向散列函数的认证等。

访问控制是对用户合法使用资源的认证和控制，目前信息系统的访问控制主要是基于角色的访问控制机制(Role-Based Access Control，RBAC)及其扩展模型。RBAC 机制主要由 Sandhu 于 1996 年提出的基本模型 BRAC96 构成，一个用户先由系统分配一个角色，如管理员或普通用户等，登录系统后，根据用户的角色所设置的访问策略实现对资源的访问。显然，同样的角色可以访问控制方法，是基于用户的访问控制。

对物联网而言，末端是感知网络，可能是一个感知结点或一个物体，采用用户角色的形式进行资源的控制显得不够灵活，主要表现在以下 3 点：

(1) 基于角色的访问控制在分布式的网络环境中已呈现出不相适应的地方，如对具有时间约束资源的访问控制，访问控制的多层次适应性等方面需要进一步探讨。

(2) 结点不是用户，而是各类传感器或其他设备，且种类繁多，基于角色的访问控制机制中角色类型无法一一对应这些结点，因此，使 RBAC 机制难以实现。

(3) 物联网主要是信息的感知互动过程，包含了信息的处理、决策和控制等过程，特别是反向控制是物物相连的特征之一，资源的访问呈现动态性和多层次性，而 RBAC 机制中一旦用户被指定为某种角色，他的可访问资源就相对固定了，所以，寻求新的访问控制机制是物联网，也是互联网值得研究的问题。

基于属性的访问控制(Attribute-Based Access Control，ABAC)是近几年研究的热点，如果将角色映射成用户的属性，可以构成 ABAC 与 RBAC 的对等关系，而属性的增加相对简单，同时基于属性的加密算法可以使 ABAC 得以实现。ABAC 方法的问题是对较少的属性来说，加密解密的效率较高，但随着属性数量的增加，加密的密文长度增加，使算法的实用性受到限制，目前有两个发展方向：基于密钥策略和基于密文策略，其目标就是改善基于属性的加密算法的性能。

3.3.4 公钥基础设施——PKI

1. PKI 概述

公钥基础设施(Public Key Infrastructure，PKI)是一种遵循既定标准的密钥管理平台，它能够为所有网络应用提供加密和数字签名等密码服务及所必需的密钥和证书管理体系，简单来说，PKI 就是利用公钥理论和技术建立的提供安全服务的基础设施。PKI 技术是信息安全技术的核心，也是电子商务的关键和基础技术。

PKI 的基础技术包括加密、数字签名、数据完整性机制、数字信封和双重数字签名等。PKI 是指用公钥概念和技术来实施和提供安全服务的具有普适性的安全基础设施，指任何以公钥技术为基础的安全基础设施都是 PKI，若没有好的非对称算法和好的密钥管理就不可能提供完善的安全服务，不能称为 PKI，即该定义中已经隐含了必须具有的密钥管理功能。

X. 509 标准中，为了区别于权限管理基础设施(Privilege Management Infrastructure，PMI)，将 PKI 定义为支持公开密钥管理并能支持认证、加密、完整性和可追究性服务的基

础设施。这个概念与第一个概念相比，不仅仅叙述 PKI 能提供的安全服务，更强调 PKI 必须支持公开密钥的管理。也就是说，仅仅使用公钥技术还能叫做 PKI，还应该提供公开密钥的管理。因为 PMI 仅仅使用公钥技术但并不管理公开密钥，所以 PMI 就可以单独进行描述，而不至于跟公钥证书等概念混淆。

美国国家审计总署在 2001 年和 2003 年的报告中都把 PKI 定义为由硬件、软件、策略和人构成的系统，当完善实施后，能够为敏感通信和交易提供一套信息安全保障，包括保密性、完整性、真实性和不可否认。

2. 基本组成

完整的 PKI 系统必须具有权威认证机构(CA)、数字证书库、密钥备份及恢复系统、证书作废系统、应用接口(API)等基本构成部分，构建 PKI 也将围绕着这 5 大系统来构建。PKI 技术是信息安全技术的核心，也是电子商务的关键和基础技术。PKI 的基础技术包括加密、数字签名、数据完整性机制、数字信封和双重数字签名等。

(1) 认证机构(CA)。即数字证书的申请签发机关，CA 必须具备权威性的特征。

(2) 数字证书库。用于存储已签发的数字证书及公钥，用户可由此获得所需的其他用户的证书及公钥。

(3) 密钥备份及恢复系统。如果用户丢失了用于解密数据的密钥，则数据将无法解密，这将造成合法数据丢失。为避免这种情况，PKI 提供备份与恢复密钥的机制。但需注意，密钥的备份与恢复必须由可信的机构来完成，并且，密钥备份与恢复只能针对解密密钥，签名私钥为确保其唯一性而不能够作备份。

(4) 证书作废系统。证书作废处理系统是 PKI 的一个必备的组件。与日常生活中的各种身份证件一样，当密钥介质丢失或用户身份变更等时，证书有效期内也可能作废，即 PKI 必须提供作废证书的一系列机制。

(5) 应用接口(API)。PKI 的价值在于使用户能够方便地使用加密、数字签名等安全服务，因此一个完整的 PKI 必须提供良好的应用接口系统，使得各种各样的应用能够以安全一致、可信的方式与 PKI 交互，确保安全网络环境的完整性和易用性。

通常来说，CA 是证书的签发机构，它是 PKI 的核心。众所周知，构建密码服务系统的核心内容是如何实现密钥管理。公钥体制涉及一对密钥(即私钥和公钥)，私钥只由用户独立掌握，无须在网上传输，而公钥则是公开的，需要在网上传送，故公钥体制的密钥管理主要是针对公钥的管理问题，目前较好的解决方案是数字证书机制。

3. 目标

PKI 是一种基础设施，其目标就是要充分利用公钥密码学的理论基础，建立起一种普遍适用的基础设施，为各种网络应用提供全面的安全服务。公开密钥密码为我们提供了一种非对称性质，使得安全的数字签名和开放的签名验证成为可能。而这种优秀技术的使用却面临着理解困难、实施难度大等问题。正如让电视机的开发者理解和维护发电厂有一定的难度一样，要让每一个应用程序的开发者完全正确地理解和实施基于公开密钥密码的安全有一定的难度。PKI 希望通过一种专业的基础设施的开发，让网络应用系统的开发人员从烦琐的密码技术中解脱出来，而同时享有完善的安全服务。

将 PKI 在网络信息空间的地位与电力基础设施在工业生活中的地位进行类比可以更好地理解 PKI。电力基础设施通过伸到用户的标准插座为用户提供能源，而 PKI 通过延伸用户本地的接口为各种应用提供安全的服务。有了 PKI，安全应用程序的开发者可以不用再关心那些复杂的数学运算和模型，而直接按照标准使用一种插座（接口）。正如电冰箱的开发者不用关心发电机的原理和构造一样，只要开发出符合电力基础设施接口标准的应用设备，就可以享受基础设施提供的能源。

PKI 与应用的分离也是 PKI 作为基础设施的重要标志。正如电力基础设施与电器的分离一样，网络应用与安全基础实现了分离，有利于网络应用更快地发展，也有利于安全基础设施更好地建设。正是由于 PKI 与其他应用能够很好地分离，才使得我们能够将之称为基础设施，PKI 也才能从千差万别的安全应用中独立出来，才能有效地、独立地发展壮大。PKI 与网络应用的分离实际上就是网络社会的一次“社会分工”，这种分工可能会成为网络应用发展史上的重要里程碑。

4. 内容

PKI 在公开密钥密码的基础上，主要解决密钥属于谁，即密钥认证的问题。通过数字证书，PKI 很好地证明了公钥是谁的，PKI 的核心技术就围绕着数字证书的申请、颁发、使用与撤销等整个生命周期展开，其中，证书撤销是 PKI 中最容易被忽视，但却是很关键的技术之一，也是基础设施必须提供的一项服务。

PKI 技术的研究对象包括数字证书、数字证书认证中心、证书持有者和证书用户，以及为了更好地成为基础设施而必须具备的证书注册机构、证书存储和查询服务器、证书状态查询服务器、证书验证服务器等。

PKI 作为基础设施，两个或多个 PKI 管理域的互连就非常重要。PKI 域间如何互联，如何更好地互联就是建设一个无缝的大范围的网络应用的关键。在 PKI 互连过程中，PKI 关键设备之间，PKI 末端用户之间，网络应用与 PKI 系统之间的互操作与接口技术就是 PKI 发展的重要保证，也是 PKI 技术的研究重点。

5. 优势

PKI 作为一种安全技术，已经深入到网络的各个层面。这从一个侧面反映了 PKI 强大的生命力和无与伦比的技术优势。PKI 的灵魂来源于公钥密码技术，这种技术使得“知其然不知其所以然”成为一种可以证明的状态，使得网络上的数字签名有了理论上的安全保障。围绕着如何用好这种非对称密码技术，数字证书破壳而出，并成为 PKI 中最为核心的元素。

PKI 的优势主要表现在：

（1）采用公开密钥密码技术，能够支持可公开验证并无法仿冒的数字签名，从而在支持可追究的服务上具有不可替代的优势。这种可追究的服务也为原发数据完整性提供了更高级别的担保。支持可以公开地进行验证，或者说任意的第三方可验证，能更好地保护弱势个体，完善平等的网络系统间的信息和操作的可追究性。

（2）由于密码技术的采用，保护机密性是 PKI 最得天独厚的优点。PKI 不仅能够为相互认识的实体之间提供机密性服务，同时也可以为陌生的用户之间的通信提供保密支持。

（3）由于数字证书可以由用户独立验证，不需要在线查询，原理上能够保证服务范围的

无限制扩张，这使得 PKI 能够成为一种服务巨大用户群的基础设施。PKI 采用数字证书方式进行服务，即通过第三方颁发的数字证书证明末端实体的密钥，而不是在线查询或在线分发。这种密钥管理方式突破了过去安全验证服务必须在线的限制。

(4) PKI 提供了证书的撤销机制，从而使得其应用领域不受具体应用的限制，撤销机制提供了在意外情况下的补救措施，在各种安全环境下都可以让用户更加放心。另外，因为有撤销技术，不论是永远不变的身份，还是经常变换的角色，都可以得到 PKI 的服务而不用担心被窃后身份或角色被永远作废或被他人恶意盗用。为用户提供“改正错误”或“后悔”的途径是良好工程设计中必须的一环。

(5) PKI 具有极强的互联能力。不论是上下级的领导关系，还是平等的第三方信任关系，PKI 都能够按照人类世界的信任方式进行多种形式的互联互通，从而使 PKI 能够很好地服务于符合人类习惯的大型网络信息系统。PKI 中各种互联技术的结合使建设一个复杂的网络信任体系成为可能。PKI 的互联技术为消除网络世界的信任孤岛提供了充足的技术保障。

3.4　常用安全协议

3.4.1　Kerberos 协议

Kerberos 协议是一种网络认证的协议，其工作的原理是通过密钥，对客户端或者服务器提供一个认证服务，与传统的认证协议相比，Kerberos 协议不需要物理安全，只要通过认证，就可以随意的读取和修改数据库。因此，这种协议被广泛地应用在第三方认证服务领域，该协议在 TCP/IP 协议栈中，处于 UDP 和 TCP 的上层，与 HTTP 处于同一个级别。在具体认证的过程中，采用数据加密算法进行认证，用户机首先要发出相应的请求，然后安装服务器的证书文件，服务器如果能够读取正确的用户密钥，那么就可以通过相应的认证，这个证书文件还可以为经过认证后的通信进行加密，保证通信内容的安全。目前，很多服务器都采用这个协议进行加密，以此来保证网络的安全，尤其是一些含有重要内容的通信，以及需要相应的身份才能访问数据的系统，Kerberos 协议可以很好地防止连接和窃听，虽然该协议的安全性较高，可以为不同的服务提供单独的认证，但这种认证体系不会验证物理地址，因此无法检验用户的真实性，如果密钥的数量过多，那么对认证服务器的性能，会有很高的要求。

3.4.2　SET 协议

SET 是 Secure Electronic Transaction 的缩写，中文名为安全电子交易协议，该协议是随着电子支付的普及应用，逐渐产生的一种安全协议，由于其可以很好地处理用户、商户和银行之间的关系，在 B2C 等网站上得到了普及应用，经过了多年的使用，已经成为了信用卡网上交易的国际标准。在 TCP/IP 协议栈中，SET 协议处于 HTTP 的上层，在实际的网上交易中，由于用户与商家都是经过网络沟通，具有一定的虚拟性，在用户确定订单之后，商家希望用户可以填写更多真实的信息，而用户则希望一些私密的账户信息等可以保密，但是由

于双方不够了解，经常会出现矛盾甚至是欺诈的现象，而 SET 协议的应用可以很好地解决这个问题，利用这种协议对双方进行认证，用户信用卡等信息就不会被商家知道。SET 协议的安全系数很高，所有参与的用户都必须先安装证书，以此来识别自己的身份，可以很好地防止欺诈现象的发生，但是由于这种算法自身非常复杂，要想使用这种算法需要较高的成本，而且对此加密对服务性能的要求很高，在使用的过程中，必须安装相应的插件和软件。

3.4.3 SSL 协议

SSL 协议是网景公司在开发浏览器的过程中，研发的一种安全协议，因此该协议主要是为了保证网络数据传输的安全，采用数据加密技术，可以很好地防止数据在上行或下行的过程中被窃取。因此，现在的 Web 浏览器，基本上都支持该协议，SSL 协议在 TCP/IP 协议栈中处于 TCP 和 HTTP 之间，SSL 协议可以选择多种加密算法来进行认证。SSL 协议的认证是双向的，首先就是服务器的认证，客户机要向服务发出请求信息，服务器接收到用户的请求后，会返回用于生成密钥的相关信息，用户收到这个信息后就可以生成密钥，完成对服务器的认证，最后服务器还要向客户机发送一个提问，客户机返回相应的数据后，才算完成双方的认证。SSL 协议的应用非常广泛，几乎所有涉及 Web 通信的领域，都可以采用该协议来保证网络的安全，尤其是该协议的设置非常简单，只需要少量的成本，不需要安装任何的插件和软件，但是该协议只能保证传输过程的安全。

3.4.4 SHTTP 协议

SHTTP 是 Secure HyperText Transfer Protocal 的缩写，中文名为安全超文本转换协议，该协议是在传统 HTTP 的基础上，为了保证网络的安全性，研发的一种新的网络安全协议，SHTTP 协议的应用，可以很好地兼容 HTTP 的程序，这种协议可以提供多种安全措施，能够满足互联网上不同用户的需求，SHTTP 与 HTTP 处于同一协议层中。SHTTP 可以通过不同的算法来保证数据的安全，如常见的 RSA、DES 等，在实际的应用中，可以与 SSL 协议共同来保证数据传输的安全，也可以协同 SET 协议等，进行具体功能上的保护，虽然这种协议具有很高的安全性，但是实现起来具有较大的难度，因此目前还没有得到普及应用。

3.5 本章小结

本章介绍了密码学的历史、密码系统的概念、密码的分类，重点介绍了几种常用加密技术原理及其应用，最后介绍了几种安全协议。

密码系统又称为密码体制，是指能完整地解决信息安全中的机密性、数据完整性、认证、身份识别及不可抵赖等问题中的一个或几个的一个系统。其目的是人们能够使用不安全信道进行安全的通信。

用加密技术主要分为对称加密算法和非对称加密算法。对称加密算法指加密和解密使用相同密钥的加密算法，就是加密密钥能够从解密密钥中推算出来，同时解密密钥也可以从

加密密钥中推算出来。在计算机专网系统中广泛使用的对称加密算法有 DES、IDEA 和 AES。非对称加密算法是指加密密钥与解密密钥是不同的，一个称为公开密钥，另一个称为私人密钥(或秘密密钥)，因此这种密码体系也称为公钥密码体系。公钥密码体制的算法中最著名的代表是 RSA 系统，此外还有椭圆曲线、背包密码和 ElGamal 算法等。

基于数据加密的应用技术包括鉴别、数字签名、物联网认证与访问控制以及公钥基础设施等。

常用安全协议有 Kerberos 协议、SET 协议、SSL 协议和 SHTTP 协议等。

复习思考题

1. 简述密码学的定义和作用。
2. 古典密码学主要分成哪几种类型？请详述其中一种。
3. 什么是非对称加密，有哪些特点？请介绍几种非对称加密算法。
4. 什么是公钥加密，有哪些特点？请介绍几种公钥加密算法。
5. 什么是单向散列函数？请举例说出有哪些属于单向散列函数。
6. 简述数字签名技术的原理。
7. 数字签名与加密技术在密钥对的使用上有什么区别？
8. PKI 的优势主要表现在哪些方面？
9. 网络中的认证包括哪些方面？什么是物联网认证？
10. 目前信息系统的访问控制有哪几种？分别简述其特点。
11. 常用的网络协议有哪些？请阐述每一种协议的工作原理。