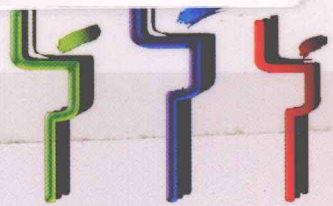
AnQuan

普通高校信息安全系列教材
北京市重点学科共建项目：计算机应用技术

黎妹红　韩　磊　编著

身份认证技术及应用

SHENFEN RENZHENG
JISHU JI YINGYONG



北京邮电大学出版社
www.buptpress.com

普通高校信息安全系列教材

北京市重点学科共建项目：计算机应用技术

身份认证技术及应用

黎妹红　韩磊　编　著

北京邮电大学出版社
www. buptpress. com

内容简介

如今，身份认证技术已经从早期的口令技术发展到使用智能卡、动态口令、生物识别技术和USB Key技术等，从软件认证到硬件认证，从单因子认证到双因子认证，从静态认证到动态认证，各种技术层出不穷。

为适应新的人才培养的要求，结合信息安全专业的自身特点，本书全面介绍了身份认证技术，包括口令技术、智能卡技术和生物认证技术的基本原理和方法，以及常用的认证协议和应用，并对目前的认证技术的防攻击性做了系统的分析和研究，内容由浅入深，循序渐进。

本书可作为高等学校信息安全专业高年级本科生和研究生课程的指导教材，也可用于具有一定密码学知识的工程技术人员使用。

图书在版编目(CIP)数据

身份认证技术及应用/黎妹红，韩磊编著. --北京：北京邮电大学出版社，2012.3
ISBN 978-7-5635-2934-6

Ⅰ.①身… Ⅱ.①黎…②韩… Ⅲ.①密码—理论 Ⅳ.①TN918.1

中国版本图书馆CIP数据核字(2012)第035054号

书　　名：身份认证技术及应用
著作责任者：黎妹红　韩磊
责任编辑：付兆华
出版发行：北京邮电大学出版社
社　　址：北京市海淀区西土城路10号(邮编：100876)
发　行　部：电话：010-62282185　传真：010-62283578
E-mail：publish@bupt.edu.cn
经　　销：各地新华书店
印　　刷：北京联兴华印刷厂
开　　本：787 mm×960 mm　1/16
印　　张：10.75
字　　数：230千字
印　　数：1—3 000册
版　　次：2012年3月第1版　2012年3月第1次印刷

ISBN 978-7-5635-2934-6　　定　价：24.00

前言

建立信息安全体系的目的就是要保证存储在计算机及网络系统中的数据只能够被有权操作的人访问，所有未被授权的人无法访问到这些数据。这里说的是对“人”的权限的控制，即对操作者物理身份的权限控制。如果把信息安全体系看作一个木桶，那么大家熟悉的如防火墙、入侵检测、VPN、安全网关、安全目录等这些安全产品就是组成木桶的一块块木板，则整个系统的安全性取决于最短的一块木板。这些模块在不同的层次上阻止了未经授权的用户访问系统，这些授权的对象都是用户的数字身份。而身份认证模块就相当于木桶的桶底，由它来保证物理身份和数字身份的统一，如果桶底是漏的，那桶壁上的木板再长也没有用。因此，身份认证是整个信息安全体系最基础的环节，身份安全是信息安全的基础。

据统计，针对如此重要的身份认证课程，国内学校的信息安全专业一般都没有专门开设，即使有的话也是作为某一门课程的章节来讲授，缺少系统的理论知识和实验平台。实际上，信息安全专业里边也根本找不到专门的书籍来介绍身份认证的知识，充其量也只是一本书中的一个章节而已。

为适应新的人才培养的要求，结合信息安全专业的自身特点，本书全面介绍了身份认证技术，包括口令技术、智能卡技术和生物认证技术的基本原理和方法，以及常用的认证协议和应用，并对目前的认证技术的防攻击性作了系统的分析和研究，内容由浅入深，循序渐进。

在本书的编写过程中，作者参考了互联网上公布的研究论文和相关资料，主要源于各大学、科研机构、安全网站、安全公司及一些研究身份认证技术的个人，在此向他们表示感谢。由于资料较多，无法一一注明出处。写作过程中所参考的这些资料，其原文版权属于原作者，特此声明。

本书受到北京市重点学科共建项目——计算机应用技术(XK100040519)、中央高校基本科研业务费(2011JBM227、2011JBM230)的资助，在此表示感谢。本书在编写过程中难免出现纰漏，恳切希望各使用单位和个人对本书提出宝贵意见，以便修订时加以完善。

作　者

目 录

第1章 密码学基础

1.1 信息安全的基本性质

信息安全领域有几个基本安全性质，即保密性、认证性、数据完整性、可用性和不可否认性。其中除可用性以外的4个性质都可以用密码学工具来实现。加密是实现保密性的工具，而认证技术是实现认证性、数据完整性和不可否认性的工具。

- 数据完整性：一篇文章，不要求保密，要求没有改动，如何实现？
- 消息认证：如何确定收到的消息是发自某某？
- 身份认证：如何确定一个用户真是某某？

这些都属于认证问题。认证就是搞清“谁真是谁吗？”、“消息可靠吗？”的问题。

假设参与通信的人或者计算机叫做实体(Entity)，认证(Authentication)就是采取一些措施保证实体所宣称的身份是真实的，或者保证他们(它们)发出的消息不被非法者修改。因此存在两类基本的认证，即身份认证(Identification or Entity Authentication)和消息认证(Message Authentication)或称数据源认证(Data Origin Authentication)。

身份认证是确认通信中对方的身份，而消息认证或者数据源认证就是确认收到的消息完整地来自于正确的数据源。这里我们将信息、消息和数据视为同一概念，这一概念不是狭义的表示不确定性的香农信息，而是广义的信息概念。

本质上说，身份也可以视为一种消息，身份认证可视为一种消息认证。但是，身份认证和消息认证是应用于不同场合的两种基本认证形式，它们之间有明显不同的特征。身份认证一般具有实时性，需要证实身份的通信双方A和B都是在线的。如果A想要确认B的身份，A可以向B提出一些问题，也就是所谓的挑战，而只有真实的B可以做出正确的回答，因此A可以由B的应答来判断B的身份；B也可采用相同的方式验证A的身份，这也就是常见的挑战-应答类型的身份认证。

消息认证是确认一个产生于过去的消息的真实数据源，也就是确认消息是从规定的

数据源发出的。消息认证不需要通信双方同时在线，因为交换消息时一般有延迟，甚至通信仅是单方向的，例如 A 向 B 发送电子邮件，过了一些时间 B 才收到。A 和 B 通常不是直接通信，因此 B 收到宣称是 A 发送的消息，B 需要一些手段验证收到的消息确实来自于 A。这时采用挑战-应答方式是不方便的，为了认证，要求 A 提供给 B 一个消息时伴有附加信息，B 通过这些附加信息能够确定消息的真实来源。

与身份认证不同，消息认证不提供时间上何时产生一个消息的保证，因此不提供消息的实时性和唯一性。身份认证需要通过与验证者实际通信，确认宣称者的身份，因此是实时性的，而且一次身份认证只有当时有效，下次认证还需要进行新的认证，这即身份认证的唯一性；身份认证除了身份以外，不涉及其他有意义的消息，而消息认证中重要的正是所传输的消息，而且需要保证消息的完整性。所以说消息认证和身份认证具有不同的性质和要求。

和消息认证密切相关的一个概念就是数据完整性。数据完整性(Data Integrity)就是保持数据在其产生、传输、存储过程中不被非法改动。它是信息安全的一个基本要求。从上面的定义上看，消息认证应当提供数据完整性，也就是说消息认证不仅确认消息的来源，还要保证消息的完整性。因此数据完整性与消息认证这两个问题，一般是不可分割的，即被改变的数据会有一个新的数据源；如果一个数据的数据源不能被确定，那么数据是否被改变也就不能确定。所以提供数据完整性的机制意味着提供数据源认证或消息认证，反之数据源认证必然提供数据完整性。(对于完整性机制提供消息认证这一点，有的学者认为数据完整性仅保证消息不被窜改，并不一定要求确认消息的来源。因此这两个概念的含义需要根据具体使用情况加以确认。)

实际应用中，消息认证和数据完整性要求能够防止对消息的假冒、窜改、插入、删除、重排等主动攻击。

在网络通信中，除了对消息的窜改、伪造、假冒等攻击，还存在对消息重放、延时等形式的攻击，因此还需要验证消息的顺序性和时间性。前面我们已经知道，消息认证一般不提供时间性和唯一性。解决的办法是在基本的消息认证基础上，加上时间戳、唯一数等参数，构成所谓的记录认证 (Transaction Authentication)。因此，记录认证就是消息认证的扩展，附带提供数据的唯一性和时间性(实效性)保证，以阻止不易察觉的消息重放攻击。数字签名和 Hash 函数是实现认证的基本工具，其安全性和构造有丰富内容，具体算法很多。

数字签名的功能强大，除了可以实现认证的功能外，还有不可否认性；还可以附加不同的功能以实现某些特殊的应用。数字签名和 MAC 相比，数字签名可以被任何参与者验证，而 MAC 只能拥有共享密钥者才能验证。但 Hash 函数相比数字签名，有其自身的简单、灵活的特点。数字签名一般是利用公钥体制实现的，而消息认证码 MAC 一般是利用对称密钥体制实现的，我们知道公钥体制需要的计算量一般比对称密钥体制大得多，因此 MAC 实现消息认证速度快，使用灵活、方便。但是另一方面，对称密钥技术也存在密

钥分配的困难，所以实际中应当根据不同的应用场合，使用或者结合使用不同的认证技术。

认证技术是信息安全领域的主要工具。以往的信息安全技术多侧重于加密实现的保密性，随着互联网的普及，越来越多的应用需要确认对方身份和消息来源的真实性，因此认证技术日益显现其重要性。在各类网络安全协议中，在电子商务和电子政务中，在智能卡系统、监控系统、访问控制中，认证是必备的环节。数字签名作为认证技术之一，还有其自身的多功能性质，附加功能的数字签名，在一些领域例如电子现金、电子选举等，起着至关重要的作用，在各种证书、PKI 技术等，以及近来兴起的可信计算技术，依赖的主要是认证和签名技术。

1.2 Hash 技术基础

数字签名实现认证，具有高安全性，但是由于公钥体制的速度相对较慢，而在某些场合需要更加快速灵活的认证方式，这就需要用到实现认证的另一个主要工具——Hash（哈希）函数。Hash 函数是实现消息认证和数据完整性功能的一个主要工具。它是在 20 世纪 70 年代伴随着数字签名技术，为了实现信息安全目的应运而生的。

Hash 函数的基本思想是作为输入串的简明表示或代表，可以用作确认输入串的唯一凭证。形象地说，Hash 函数的目的就是要产生文件、消息或其他数据块的“指纹”。所以 Hash 函数也被叫做消息摘要（Message Digest）、指纹（Finger-print）或印记（Imprint），等等。因此，Hash 函数将大数据量的消息或文件，压缩成短的 Hash 函数值，不是为恢复原有信息，而是为了提供一种附加信息，当原来消息中任意一个二进制位发生变化时，都将引起 Hash 值的变化，因此可以实现消息的认证性和数据完整性。例如，文件所有者可保留原文件的 Hash 值，作为检查数据完整性的一个参数。使用文件时再计算一次文件的 Hash 值，如果与原 Hash 值不一样，则说明文件被改动过了。另外实用的数字签名方案中，一般都是将消息首先经过一个 Hash 函数压缩，之后对 Hash 值进行签名，这样不仅节省空间和提高效率，还可以防止一些类型的攻击。

Hash 函数是一个可有效计算的函数，它将任意有限长度的消息压缩到固定长的较短的值。即 Hash 函数为

$$h: D \to R, |D| > |R|$$

它作用于一个任意长度的消息 $x \in D$，返回一个固定长度的函数值 $h(x)$，$h(x)$ 称为 Hash 值，其长度为固定值 n。上面的绝对值符号表示集合的元素个数。上述 Hash 函数形式也常写做二进制形式为

$$h: \{0,1\}^* \to \{0,1\}^n$$

由定义可以看出，Hash 函数至少有以下两个基本性质。

① 压缩性。即将一个任意有限长的输入，映射为一个固定长的短的输出。

② 容易计算。给出 h 和输入 x，计算 $h(x)$是容易的。

由于定义域 D 的元素个数大于值域R 的元素个数，Hash 函数是多对一的映射，所以存在碰撞是不可避免的，也就是多个不同的输入可能对应同一个输出。如果输入串的二进制长度为 t，输出串的二进制长度为 $n(t>n)$，并且假定所有输出的串以相等的概率出现(即均匀分布)，则 2^{t-n} 个输入映射为相同的输出。如果发生碰撞，则意味着两个或多个不同的消息产生同一个 Hash 值，这时 Hash 函数则无法保证数据完整性。当存在碰撞的 Hash 函数用于数字签名时，将使不同的消息产生相同的签名。因此应用中除了具有前述的两个基本性质以外，还要求 Hash 函数具有其他性质。

对一个无密钥的 Hash 函数，应当有以下 3 个附加的性质。

① 原象阻止性(Pre-image Resistance)，也称为单向性(One-way)。对所有事先确定的输出 y，发现任何输入 x，使 $h(x)=y$ 计算困难；

② 第二原象阻止性(2nd Pre-image Resistance)，也称为弱碰撞阻止性。对任何指定的输入 x，发现任何第二个输入 $x'\neq x$，使得 $h(x')=h(x)$ 计算困难；

③ 碰撞阻止性(Collision Resistance)，也称为强碰撞阻止性。发现两个任意选择的不同的输入 x,x'，使得 $h(x')=h(x)$ 计算困难。

一个具有原象阻止性和第二原象阻止性的 Hash 函数，称为单向性 Hash 函数 (One Way Hash Function，OWHF)，有时也被称为弱单向性 Hash 函数；一个具有第二原象阻止性和碰撞阻止性的 Hash 函数，称为碰撞阻止性[①] Hash 函数(Collision Resistant Hash Function，CRHF)，有时也被称为强单向性 Hash 函数。

消息认证码(Message Authentication Code，MAC)就是带有密钥的 Hash 函数，此时计算输入消息的 Hash 值，需要具有密钥。

广义上说，大致存在两种类型的 Hash 函数，即带有密钥(Keyed)的 Hash 函数和无密钥(Unkeyed)的 Hash 函数。带有密钥的 Hash 函数主要就是消息认证码 MAC，而无密钥的 Hash 函数主要为修改检测码(Manipulation Detection Code，MDC)。带密钥的 Hash 函数在计算和检验 Hash 函数时，必须拥有相应的密钥；而无密钥的 Hash 函数不需如此，任何人都可以计算消息的 Hash 函数值。

需要注意的是，有些学者将 Hash 函数直接定义为无密钥形式的 Hash 函数，而将有密钥的消息认证码 MAC 进行单独分类。这样做是为了强调 MAC 和无密钥 Hash 函数的不同，MAC 是专门为了消息认证而设计的。因为只有拥有共享密钥者才能计算 MAC，所以单独使用 MAC 就能确定数据的来源和保证其完整性，也就是可实现消息源认证；而单独使用 MDC 可实现数据完整性检测，但实现数据源认证需要附加可靠信道，用于保证 MDC 的来源可靠性。为了方便，我们对 Hash 函数的这两种分类不作区分，读

① 碰撞阻止性原来被称为无碰撞的(Collision Free)，这是不合适的，因为碰撞不是没有，而是需要避免，所以改为现在的名词。

者可以从上下文和有无密钥上容易判断出所使用的 Hash 函数这一概念属于哪一种定义。

上述 Hash 函数的一般形式是针对无密钥的形式。对于有密钥的 Hash 函数，其定义与 MDC 有所不同。

有密钥的 Hash 函数是由密钥 k 控制的函数 h_k，它具有以下性质。

① 压缩性。h_k 将一个任意有限长度的输入 x，映射成固定长度 n 的输出 $h_k(x)$；

② 容易计算。对一个已知的函数 h_k，给定密钥 k 和一个输入 x，$h_k(x)$ 是容易计算的，$h_k(x)$ 叫做 MAC 值或 MAC；

③ 计算阻止性(Computation Resistance)。未知或者已知多个文本(text)-MAC 对 $(x_i, h_k(x_i))$，对任何新的输入 $x(x \neq x_i)$（x 也不等于那些满足 $h_k(x') = h_k(x_i)$ 的 x'），在未知密钥 k 时，计算任何文本-MAC 对 $(x, h_k(x))$ 是困难的。

如果不满足计算阻止性，将产生伪造的 MAC。

根据不同的分类方法，Hash 函数可以有以下不同的划分。

(1) 按有无密钥分类，Hash 函数可分为以下两类。

① 无密钥的 Hash 函数。此时仅有一个输入，就是输入消息；

② 有密钥的 Hash 函数。此时有两个输入，分别是输入消息和密钥。

(2) 按实现的功能分类，Hash 函数可大致分为以下两类。

① 修改检测码(Manipulation 〈Or Mdification〉 Detection Codes，MDC)。MDC 是无密钥的 Hash 函数，它提供一个表示性的 Hash 值，用于检查输入消息是否被改动，实现数据完整性。MDC 还可细分为单向性 Hash 函数和碰撞阻止性 Hash 函数。

② 消息认证码(Message Authentication Codes，MAC)。MAC 是有密钥的 Hash 函数，一般用对称密码技术实现，有两个不同的输入，即消息和密钥。MAC 用于保证消息源认证和消息完整性。

有密钥的 MAC 可分为基于分组密码的 MAC、基于 MDC 的 MAC，以及其他类型的 MAC。

(3) 按构造方法来分，无密钥的 Hash 函数大致可分为以下几类。

① 基于分组密码的 Hash 函数(block cipher-based Hash function)。

② 基于模运算的 Hash 函数(modular Arithmetic-based Hash function)。(现已改为：基于具有安全证明的代数结构的 Hash。2009 年)

③ 专门设计的 Hash 函数(dedicated Hash function 或 customized Hash function)。

Hash 函数的作用是非常关键的，它是构成实用的密码学方案和协议的不可缺少的工具，其实现方法和安全性分析一直是密码学研究的重要内容，目前已经形成了许多实用的 Hash 函数算法。2004 年以来，密码学界迎来了分析和设计 Hash 函数算法的高潮，其原因是出现了对于作为标准普遍使用的 MD5 和 SHA-1 等杂凑算法的有效攻击方法。这些攻击也说明了如何设计安全的 Hash 函数是密码学的一项迫切任务。

1.3 加密技术基础

1.3.1 加密技术概述

1. 加密技术的基本概念

首先，我们介绍密码学中的几个基本概念。

① 明文、密文和密码算法：被伪装的信息称为“明文”，伪装后的信息称为“密文”，而加密时所采用的信息变换规则称为“密码算法”。

② 加密、解密：加密就是对机密信息加以伪装的过程，也就是把明文转变为密文的过程；而把密文转变为明文的过程称为“解密”。明文用 P 表示。在智能卡中，它表现为比特流或二进制数据。密文用 C 表示，它也是二进制数据，加密函数 E 作用于明文 P 而得到密文 C，公式为 $E(P)=C$；解密函数 D 作用于 C 产生明文 P，其表达式为 $D(C)=P$；而对明文加密解密的过程可表示为 $D(E(C))=P$。

如图 1.1 和图 1.2 所示分别描述了一个密钥（对应于对称密钥算法）和两个密钥（对应于非对称密钥算法）的加解密过程。

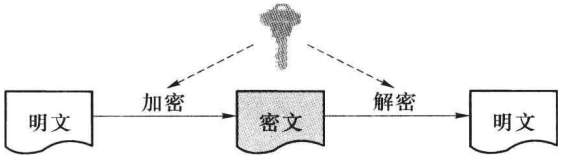


图 1.1 使用一个密钥的加解密过程

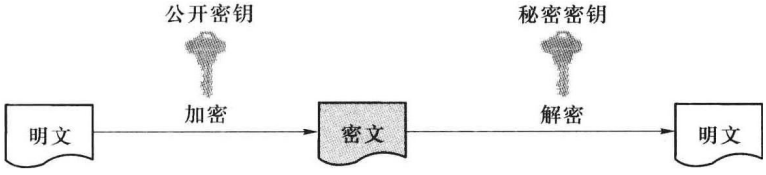


图 1.2 使用两个密钥的加解密过程

2. 现代密码体制的分类

现代密码体制的分类很多，按照密码算法对明文信息的加密方式，分为序列密码体制和分组密码体制；按照加密过程中是否注入了客观随机因素，分为确定型密码体制和概率密码体制；按照是否能进行可逆的加密变换，分为单向函数密码体制和双向函数密码体制。不过人们常用的是按照密码算法所使用的加密密钥和解密密钥是否相同，能不能由加密过程推导出解密过程（或者反之，由解密过程推导出加密过程）而将密码体制分为对

称密码体制和非对称密码体制。在对称密钥加密中，明文的加密过程是这样的——将一个固定的文本，即“密钥”，和明文一起提交给加密算法，然后加密算法返回密文。密文的解密则通过将密钥和密文提交给一个解密算法来完成。由于加密和解密都使用相同的密钥，因此这种算法被称作“对称密钥加密”。对称密钥加密有时也被称为秘密密钥加密，因为密文的安全取决于密钥能否被保密。常见的对称加密算法比较如表 1.1 所示。

表 1.1 常见对称加密算法比较

算法	密钥长度/k	轮数/轮	数学计算	应用
DES	56	16	异或、固定 S 盒	SET，智能卡，Kerberos
Triple DES	112 或 168	48	异或、固定 S 盒	S/MIME，智能卡，PGP
IDEA	126	8	异或、加法、乘法	PGP
Blowfish	可变至 448	16	异或、变长 S 盒、加法	
RC5	可变至 2 048	可变至 255	加法、减法、异或、旋转	
CAST-128	40～128	16	加法、减法、异或、旋转、固定 S 盒	PGP

在公共密钥加密（也称非对称密钥加密）中，加密和解密过程要用到两个密钥，一个用于加密数据的公钥（Public Key），另一个用于解密数据的私钥（Secret Key）。任何需要将数据安全地发送给私钥持有者的人首先获得公钥。由于只有通过私钥才能解密密文，因此用于加密的密钥通常不需要保密。这比起对称密钥加密来有一个很大的优点，那就是消息的发送方不需要与相应的接收方交换需保密的密钥。只有消息的接收方需要持有一个保密的密钥。常见非对称加密算法比较如表 1.2 所示。

表 1.2 常见非对称加密算法比较

算法	公开密钥/位	私有密钥/位	加密 100 bit 消息所需带宽	加密 2 000 bit 消息所需带宽	存储空间
RSA	1 024	2 048	1 024	2 048	大
DSA	1 024	160	1 024	2 048	大
ECC	161	160	120	1 024	小

公共密钥加密的主要缺点是速度慢。公共密钥算法与对称密钥算法相比非常慢，所以对于需要考虑响应时间的应用程序，采用公共密钥算法是不实际的。因此，用于保护传输中的数据的加密应用程序，例如安全套接字层（Secure Sockets Layer）协议常常结合使用公共密钥算法和对称密钥算法。发送方和接收方使用公共密钥算法进行协商，并交换双方都可以接受的、保密的密钥，然后使用这个密钥加上对称密钥算法在双方之间传输消息。对于保护静态数据的应用程序，公共密钥算法与对称密钥算法相比没有优势，因为解密密钥需要保密，这与对称密钥算法一样。如果被加密的数据范围相对比较小的话，公共

密钥算法也比较容易受到选择明文攻击。

1.3.2 对称密码算法 DES

最早、最著名的对称密钥加密算法(Data Encryption Standard,DES)是由 IBM 公司在 20 世纪 70 年代发展起来的,经过美国政府的加密标准筛选后,于 1976 年 11 月被采用,随后获得了美国国家标准局和美国国家标准协会(American National Standard Institute,ANSI)的承认。三十几年来,DES 一直活跃在国际保密通信的舞台上,扮演了十分重要的角色,被广泛应用在 ATM、磁卡、智能卡,以及加油站等很多领域。

虽然 DES 得到了十分广泛的应用,但是对于它的安全性一直存在着许多争论。这些批评主要集中在两个方面,至今也没有平息。其一,DES 是作为 LUCIFER 算法的改进版本被提出的,但是,LUCIFER 的密钥有 128 位,而 DES 却只有 56 位,批评者认为 56 位的密钥长度不足以抵御穷举攻击;其二,DES 内部结构中至关重要的 S 盒的设计标准是保密的,这样用户就无法确信 DES 的内部结构是否存在隐藏的陷门,如果存在陷门的话,美国国家安全局(National Security Agency,NSA)就能够在不知道密钥的情况下解密报文。

无论这个争论的是非曲直究竟怎样,DES 实际上获得了巨大的成功。除了政府和军事机密等极端敏感的场合外,DES 被广泛应用在金融、商业等各个领域。

1. DES 的加密方案

DES 的总体方案如图 1.3 所示。显然,加密函数需要两个输入,即明文和密钥。在 DES 加密算法中,一个明文分组的长度为 64 bit,一个密钥分组的长度为 56 bit。

从图 1.3 中可以看出,明文的处理过程可以分为三个阶段。首先,64 bit 的明文经过一个初始置换 IP 后,原比特序列被打乱顺序进行重新排列。然后,经过重排的明文要与密钥作用,进行 16 轮的迭代置换。最后一个循环产生的 64 位比特序列,还要再把左右 32 个比特对调,以便产生预输出。最后,预输出还要经过一个逆初始置换 IP^{-1} 处理后,才能产生 64 bit 的密文。

密钥的使用方式,图 1.3 中也给出了说明,可以分为两个阶段。首先,56 bit 的密钥通过一个初始置换被打乱重新排列。然后,生成子密钥对明文发生 16 轮的作用。每一轮中,密钥都被分成左右两部分,每部分都是 28 bit,分别进行移位;之后,整个密钥经过压缩置换,被处理成 48 位的子密钥,与明文发生作用。对每一个循环来说,压缩置换函数是相同的,但是,由于密钥比特的重复移位,产生的子密钥并不相同。

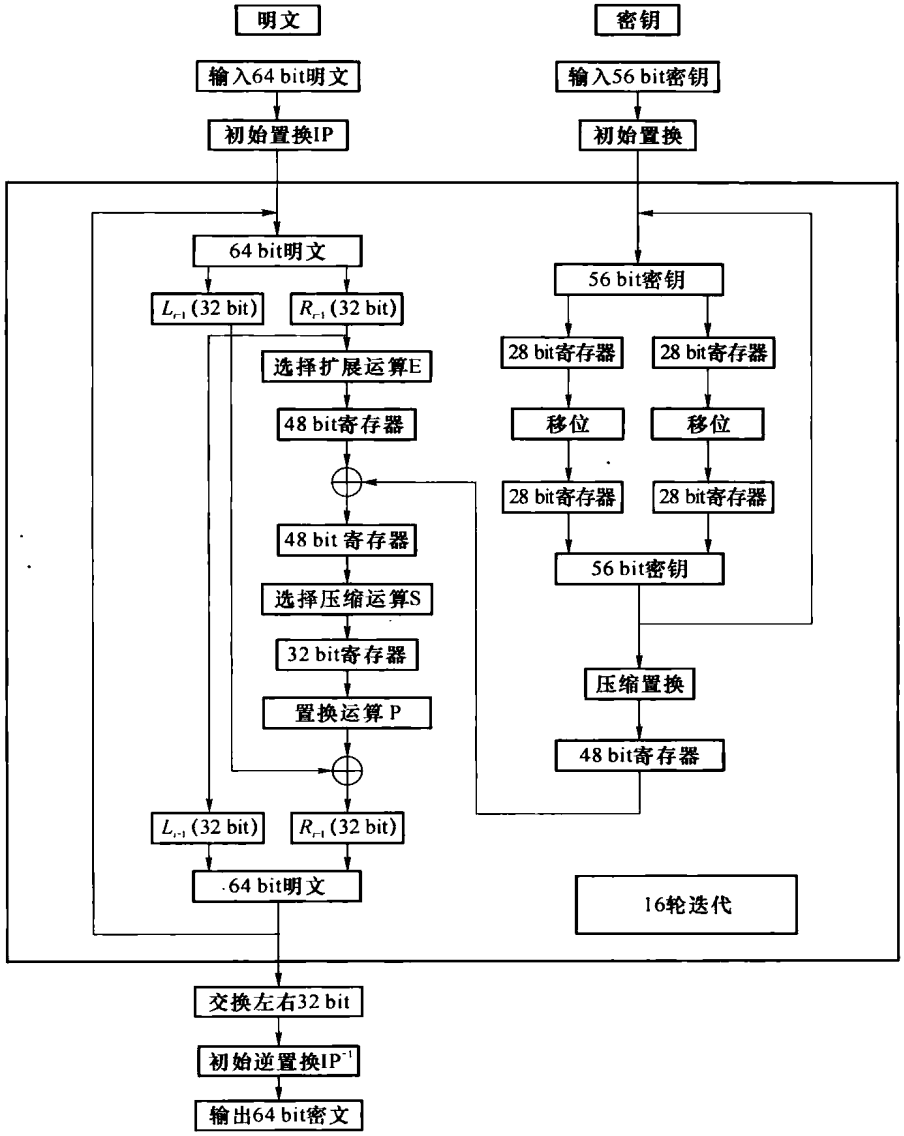


图 1.3 DES 加密流程

(1) 初始置换

初始置换是按表 1.3 所示定义的,表中第 1 个数字(第 1 行、第 1 列)58 的意思是把输入的原序列中第 58 bit 放在这个位置;第 2 个数字(第 1 行、第 2 列)50 的意思是把原序列中的第 50 bit 作为新比特序列的第 2 位;其他依次类推。