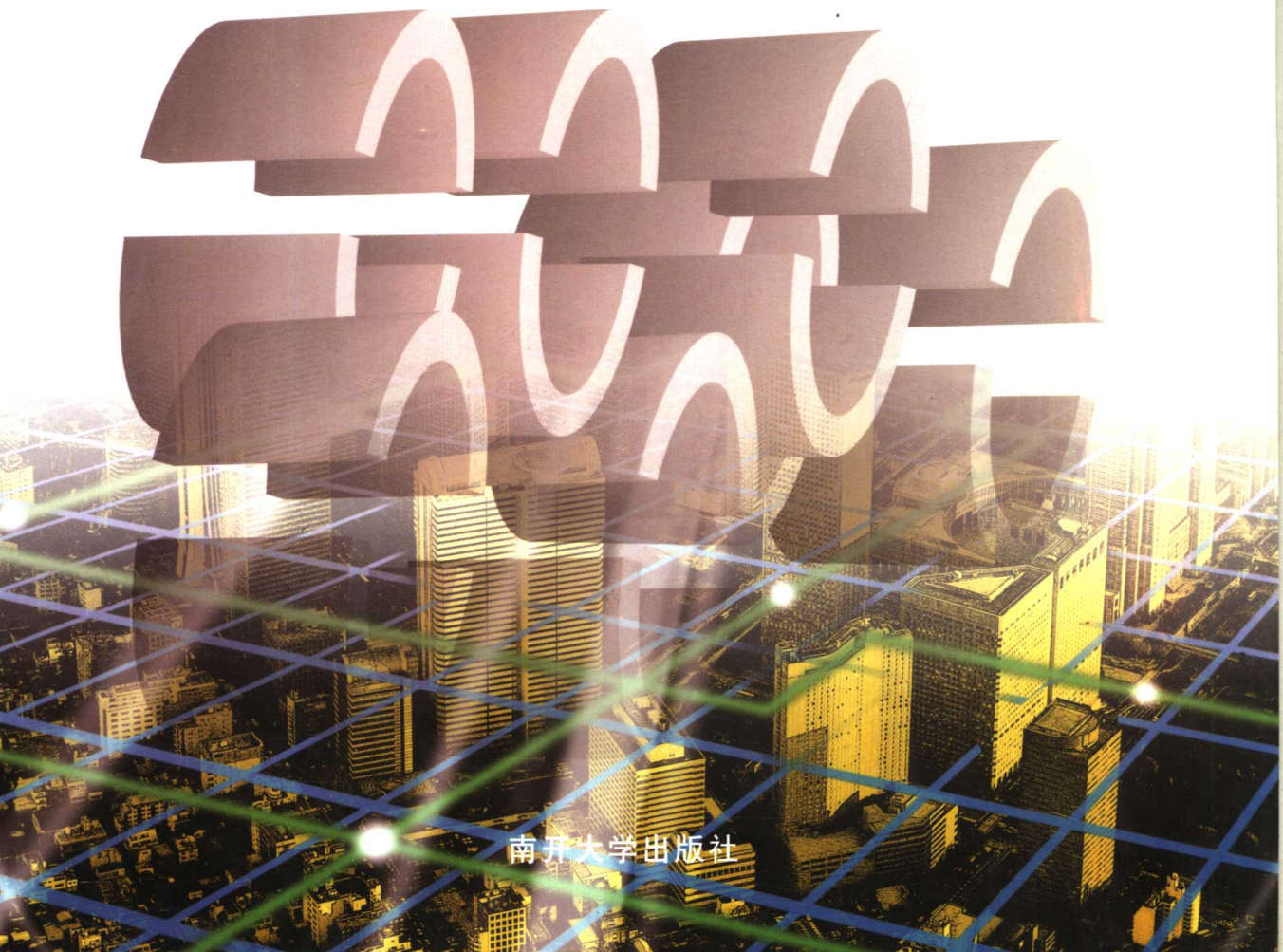
高等院校信息安全专业系列教材

信息安全实验教程

主编 高敏芬 贾春福



南开大学出版社

高等院校信息安全专业系列教材

信息安全实验教程

主编　高敏芬　贾春福

南开大学出版社
天津

图书在版编目(CIP)数据

信息安全实验教程 / 高敏芬，贾春福主编. —天津：南开大学出版社，2007.5

（高等院校信息安全专业系列教材）

ISBN 978-7-310-02701-9

Ⅰ.信… Ⅱ.①高…②贾… Ⅲ.信息系统—安全技术—高等学校—教材 Ⅳ.TP309

中国版本图书馆 CIP 数据核字(2007)第 053091 号

南开大学出版社出版发行

出版人：肖占鹏

地址：天津市南开区卫津路 94 号 邮政编码：300071

营销部电话：(022)23508339 23500755

营销部传真：(022)23508542 邮购部电话：(022)23502200

*

河北省迁安万隆印刷有限责任公司印刷

全国各地新华书店经销

*

2007 年 5 月第 1 版 2007 年 5 月第 1 次印刷

787×1092 毫米 16 开本 19.5 印张 491 千字

定价：31.00 元

如遇图书印装质量问题，请与本社营销部联系调换，电话：(022)23507125

内容简介

本实验教程共设计了 5 组 26 个实验，以及一个综合实验。这 5 组实验包括：密码学实验，如 DES 和 RSA 密码体制实验、MD5 和 DSA 加密算法实验等；计算机系统与网络配置实验，如 Windows 和 Linux 操作系统的安全配置、Windows 和 Linux 系统中 Web 和 FTP 服务器安全配置、Windows 域服务器和活动目录（Active Directory）配置，以及数据库系统安全配置等实验；网络攻防实验，如缓冲区溢出攻击与防范、网络监听技术、计算机和网络扫描技术、DoS 攻击与防范，以及欺骗类攻击与防范实验等；计算机病毒实验，如 COM 病毒、PE 病毒、宏病毒和脚本病毒等实验；网络安全设备使用实验，包括路由器的配置与使用、防火墙的配置与使用、入侵检测系统（IDS）的配置与使用、VPN 密码机的配置与使用、网络安全隔离网闸的配置与使用实验等。并在前面各部分实验的基础上设计了一个构建网络系统安全整体解决方案的综合实验，使学生能够从整体的角度考虑系统和网络的安全防护手段。

本实验教程注重实验原理的介绍，使读者能够非常清楚地把握实验目的和实验过程，提高实验效果。本教程适用于本科信息安全及相关专业的实验教学，也可供从事信息安全领域研究和开发的专业技术人员参考。

前 言

由于信息技术的高速发展和广泛应用，尤其是信息技术在政府、国防、金融和电信等国家关键部门中的应用，使得信息安全问题越来越受到人们的关注。信息安全问题已经成为影响国家安全、经济发展和社会稳定至关重要的因素之一。信息安全专门人才的培养是国家信息安全保障体系中的重要支撑。为此，国家明确提出要大力加强信息安全专门人才的培养，以满足社会对信息安全专门人才日益增长的需求。2000 年教育部首次批准武汉大学开办信息安全本科专业，2001～2003 年教育部又相继批准了北京邮电大学、上海交通大学、南开大学等高校开设信息安全本科专业。几年来，国内已有四十多所高校开设了信息安全本科专业，信息安全专门人才的培养已经开始步入正轨。

信息安全专业实验是信息安全本科专业人才培养体系的重要组成部分，目的在于巩固学生所学的内容，提高学生应用所学知识的动手能力，加深对所学知识的认识；同时培养学生独立分析问题和解决问题的能力，及其团结协作的意识和工作态度

本实验教程共设计了 5 组 26 个实验，以及一个综合实验。这五组实验包括：密码学实验，如 DES 和 RSA 密码体制实验、MD5 和 DSA 加密算法实验等；计算机系统与网络配置实验，如 Windows 和 Linux 操作系统的安全配置、Windows 和 Linux 系统中 Web 和 FTP 服务器安全配置、Windows 域服务器和活动目录（Active Directory）配置，以及数据库系统安全配置等实验；网络攻防实验，如缓冲区溢出攻击与防范、网络监听技术、计算机和网络扫描技术、DoS 攻击与防范，以及欺骗类攻击与防范实验等；计算机病毒实验，如 COM 病毒、PE 病毒、宏病毒和脚本病毒等实验；网络安全设备使用实验，包括路由器的配置与使用、防火墙的配置与使用、入侵检测系统（IDS）的配置与使用、VPN 密码机的配置与使用、网络安全隔离网闸的配置与使用实验等。并在前面各部分实验的基础上设计了一个构建网络系统安全整体解决方案的综合实验，使学生能够从整体的角度考虑系统和网络的安全防护手段。

本实验教程的特点是注重实验原理的介绍，使读者能够非常清楚地把握实验目的和实验过程，提高实验效果。一些实验之后，还列了一些思考题，目的是帮助学生加深对实验内容的认识和进一步了解其他相关的知识。本教程的编写目的是希望帮助读者全面了解信息安全方面的知识和技术，提高安全防范的能力和意识，不承担因为技术滥用而产生的连带责任。

本书由高敏芬、贾春福、段雪涛、马勇、付玉冰、梁生吉等编写，贾春福统稿并审校了全书。王晶、王世才等也参与了校稿工作，在此表示衷心的感谢。

本书适于作为信息安全及相关专业本科高年级及研究生实验教材和参考书，也适合于企事业单位的网络管理人员、安全维护人员、系统管理人员和其他相关技术人员阅读参考。

由于作者水平有限，加之时间仓促，书中难免有谬误之处，敬请广大读者批评指正。我们的 E-mail：gaomf@nankai.edu.cn。

编者

2006 年 12 月

目 录

第一章　密码学实验

密码学是信息安全领域非常重要的组成部分，也是信息安全本科专业课程体系中重要的授课内容。密码学实验的目的是在提高学生动手能力的同时，加深学生对所学密码学基础知识的理解，提高学生的学习热情和兴趣及应用所学知识的能力。

第一节 分组密码 DES 算法实验

1.1.1 实验目的

通过实际编程，进一步了解对称分组密码算法 DES 的加密和解密过程，以及 DES 的运行原理和实现方法，加深对所学对称密码算法的认识。

1.1.2 实验原理

1. DES 算法简介

数据加密标准（Data Encryption Standard，DES）是在 1973 年 5 月美国国家标准局[即现在的美国国家标准技术研究所，（NIST）]公开征集密码体制的过程中出现的。DES 由美国 IBM 公司研制，于 1977 年 1 月正式批准并作为美国联邦信息处理的标准（即 FIPS PUB - 46），同年 7 月生效，并且规定每隔 5 年由美国国家保密局（National Security Agency，NSA）做出评估，决定它是否继续作为联邦加密标准。DES 的最后一次评审在 2001 年 1 月。作为迄今为止世界上最为广泛使用和流行的一种分组密码算法，DES 对于推动密码理论的发展和应用起了非常重要的作用。

2. 算法描述

DES 加密算法流程如图 1.1.1 所示，它使用 56 比特的密钥对 64 比特的明文来加密，它是一个 16 轮的迭代型密码。加密和解密的算法一样，但加密和解密时所使用的子密钥的顺序则刚好相反。

（1）DES 加密过程描述

第一步：初始置换 IP。

目的是对明文进行换位，以打乱排列次序。一个 64 位明文分组 x，通过初始置换 IP（如表 1.1.1 所示）获得 x_0，记 $x_0=\mathrm{IP}(x)=L_0R_0$，这里的 L_0 是 x_0 的前 32 比特，R_0 是 x_0 的后 32 比特。

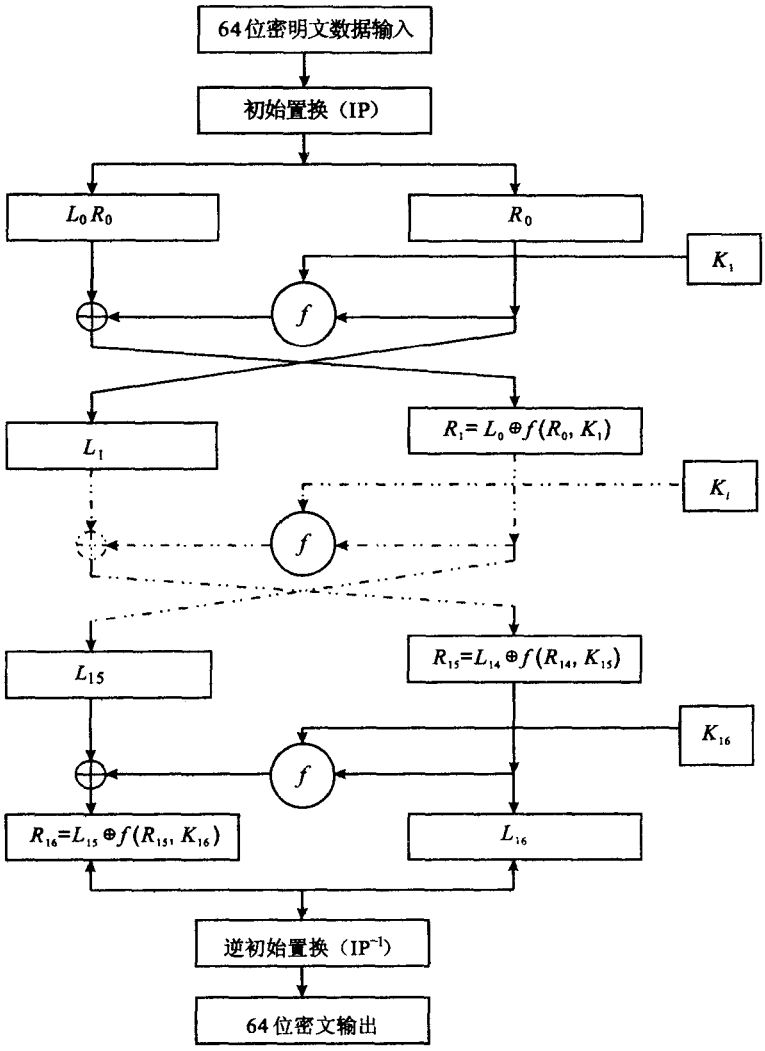


图 1.1.1 DES 加密流程

表 1.1.1 初始置换 IP

58	50	42	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

第二步：16 轮迭代过程。

16 轮的迭代运算完全相同（单轮迭代运算过程如图 1.1.2 所示），每一轮中依据下列规则计算 L_iR_i：

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

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i),\ 1 \leqslant i \leqslant 16,$$

其中 $\oplus$ 表示两个比特的异或，f 是一个函数（其计算过程将在后面描述），K_i（$1 \leqslant i \leqslant 16$）是密钥 K 的函数，它们通常被称为子密钥，长度均为 48 比特，关于 K_i 的生成方法也将在后面叙述。

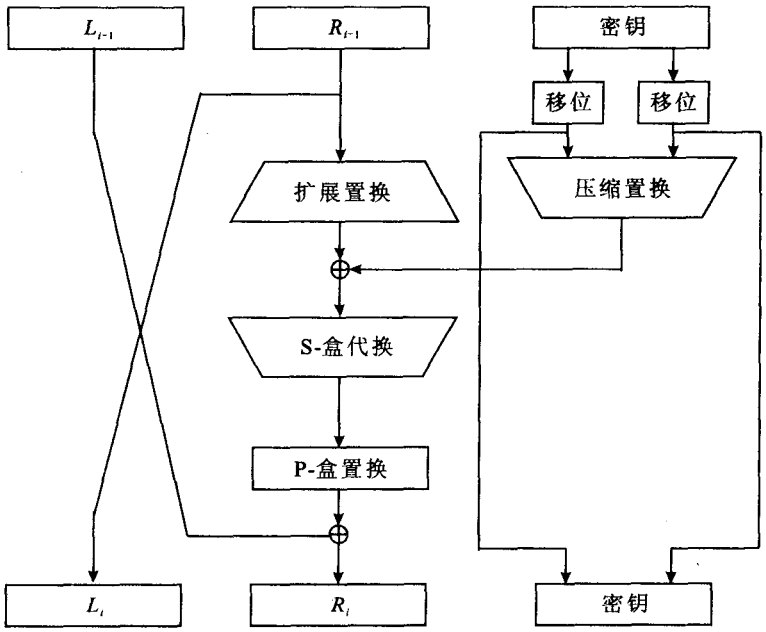


图 1.1.2　DES 单轮细节

第三步：逆初始置换 IP^{-1}。

这是 DES 算法的最后一步，对比特串 $R_{16}L_{16}$ 应用初始置换 IP 的逆置换 IP^{-1}（初始置换 IP 的逆置换 IP^{-1} 如表 1.1.2 所示），获得密文 y，即 $y = IP^{-1}(R_{16}L_{16})$。但最后一次迭代后，左边和右边不交换，而将 $R_{16}L_{16}$ 作为 IP^{-1} 的输入，目的是为了使算法可同时用于加密和解密。

表 1.1.2　初始置换 IP 的逆置换 IP^{-1}

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）变换中函数 $f(R_i, K_{i+1})$ $(0 \leqslant i \leqslant 15)$的计算

$f(R_i, K_{i+1})$是每一轮变换的核心，其计算过程如图 1.1.3 所示，包含三个子过程。

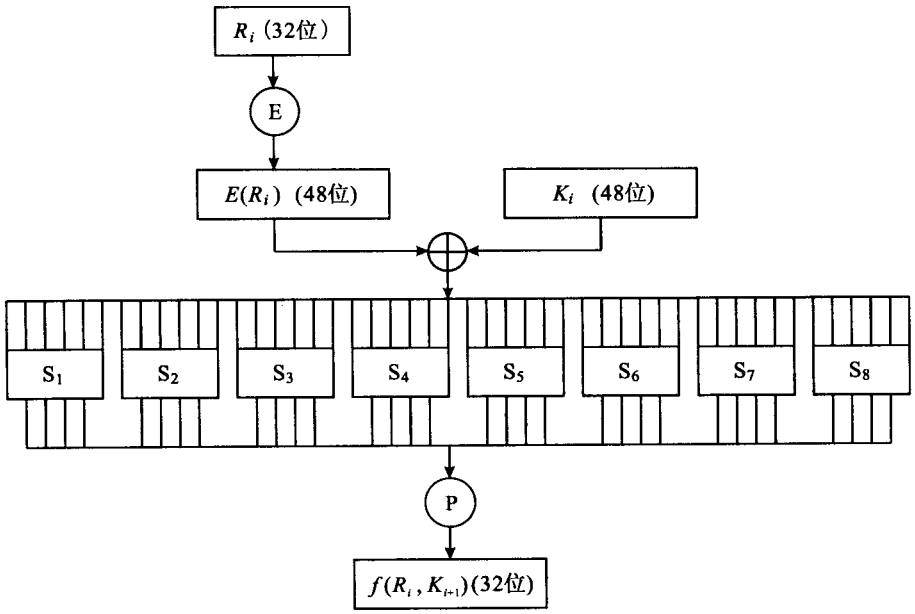


图 1.1.3 加密函数 $f(R_i, K_{i+1})$ 的计算

① 利用一个固定的扩展置换 E 将 R_i 扩展为一个长度为 48 的比特串 $E(R_i)$，这个过程与密钥无关，扩展置换 E 如表 1.1.3 所示。

表 1.1.3 扩展置换 E

扩展位	固定位				扩展位
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

② 计算 $E(R_i) \oplus K_{i+1}$，并将结果分成 8 个长度为 6 的比特串，记为

$$E(R_i) \oplus K_{i+1}=T_1T_2T_3T_4T_5T_6T_7T_8。$$

使用 8 个 S 盒 S_1、S_2、S_3、S_4、S_5、S_6、S_7、S_8（8 个 S 盒如表 1.1.4 所示）将 48 位输入变换为 32 位的输出，每一个 S_i 是一个固定的 4×16 阶矩阵，它们的元素来自于 0~15 这 16 个整数。给定一个长度为 6 的比特串，比方说 $T_j = t_1\, t_2\, t_3\, t_4\, t_5\, t_6$，我们按下列办法计算 S_j 的值：用 2 比特 $t_1\, t_6$ 对应的整数 r（0~3）来确定 S_i 的行（这里 $t_1\, t_6$ 就是 r 的二进制表示），用 4 比

特 $t_2t_3t_4t_5$，对应的整数 p（0~15）来确定 S_i 的列（这里 $t_2t_3t_4t_5$ 就是 p 的二进制表示）。记 $Y_j=S_j(T_j)$，$1\leqslant j\leqslant 8$。

表 1.1.4　S 盒

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
S_1	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7	**0**
	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8	**1**
	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0	**2**
	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13	**3**
S_2	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10	**0**
	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5	**1**
	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15	**2**
	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9	**3**
S_3	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8	**0**
	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1	**1**
	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7	**2**
	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12	**3**
S_4	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15	**0**
	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9	**1**
	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4	**2**
	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14	**3**
S_5	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9	**0**
	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6	**1**
	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14	**2**
	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3	**3**
S_6	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11	**0**
	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8	**1**
	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6	**2**
	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13	**3**
S_7	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1	**0**
	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6	**1**
	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2	**2**
	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12	**3**
S_8	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7	**0**
	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2	**1**
	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8	**2**
	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11	**3**

③ 依据一个固定的置换 P（称为 P 盒替换，如表 1.1.5 所示），将 S 盒压缩替换得到的长度为 32 的比特串 $Y = Y_1Y_2Y_3Y_4Y_5Y_6Y_7Y_8$ 进行重新排列，将所得结果 P(Y)记为 $f(R_i, K_{i+1})$ $(0 \leqslant i \leqslant 15)$。

表 1.1.5 置换 P

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

（3）子密钥生成

密钥的生成如图 1.1.4 所示。下面是它的计算过程。

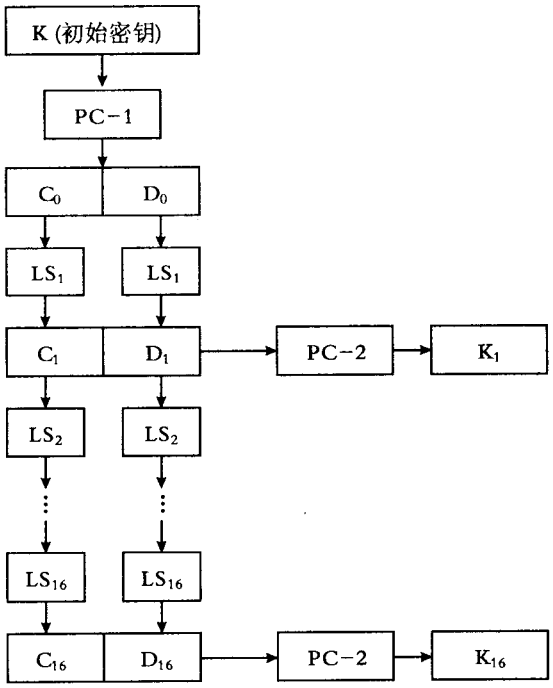


图 1.1.4 子密钥生成

① 给定一个 64 比特的密钥 K，删掉 8 个校验比特并利用一个固定的置换 PC-1（如表 1.1.6 所示）置换 K 的剩下的 56 比特，记 PC-1（K）= C_0D_0，这里 C_0 是 PC-1（K）的前 28 比特，D_0 是 PC-1（K）的后 28 比特。

表 1.1.6　置换 PC-1

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	50	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

对每一个 i, $1 \leqslant i \leqslant 16$，计算

$$C_i = \mathrm{LS}_i(C_{i-1}),$$
$$D_i = \mathrm{LS}_i(D_{i-1}),$$

其中 LS_i 表示一个或两个位置的左循环移位，当 i=1, 2, 9, 16 时，移动一个位置，当 i=3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 14, 15 时，移动两个位置。

② PC-2 是另一个固定置换，也称为选择置换，如表 1.1.7 所示。其作用是从 56 位密钥比特串中依据 PC-2 置换后输出的 48 位比特串，作为第 i 次迭代的子密钥 K_i 使用，表示为

$$K_i = \text{PC-2}\ (C_i D_i)。$$

表 1.1.7　置换 PC-2

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

（4）解密过程

解密采用同一算法实现，把密文 y 作为输入，且反过来使用密钥方案即以逆序 $K_{16}, K_{15}, \cdots, K_1$ 密钥方案，输出明文 x。解密过程可以简单描述为：

$$R_{i-1} = L_i\ ,$$
$$L_{i-1} = R_i \oplus f(L_{i-1}, K_i),\ \ 16 \geqslant \mathrm{i} \geqslant 1。$$

1.1.3 实验内容

1. 算法分析

分析 DES 加密 / 解密算法的每一个步骤，熟悉 DES 加密算法具体的每一个加密步骤和变换过程。

2. DES 算法实现

（1）利用现有的算法验证算法加密和解密的过程和效果；

（2）试着实现 DES 算法。

1.1.4 实验环境

运行 Windows 或 Linux 操作系统的 PC 机，具有 gcc（Linux）、VC（Windows）等 C 语言的编译环境。

1.1.5 实验报告

作为大实验，分组协同完成 DES 算法的实现，并利用编写的代码实现对数据的加密和解密，验证加密和解密的结果。要求明文分别为 8 的整数倍个字符和任意字符。提交报告和实验所编程序代码。

思考题

1. 查阅 IDEA（International Data Encryption Algorithm）和 AES（Advanced Encryption Standard）的相关资料，分析它们与 DES 的区别和联系，熟悉它们的基本原理和实现机制，以及安全性分析等知识。

2. 查阅相关的资料，了解一些 DES 等对称密码体制的攻击策略和方法。

第二节 公钥密码 RSA 算法实验

1.2.1 实验目的

通过实际编程了解公钥密码算法 RSA 的加密和解密过程，了解公钥密码算法的特点，加深对公钥密码体制的认识。

1.2.2 实验原理

1．公钥密码体制简介

对称密码体制要求任何密文传输之前，通信双方必须使用一个安全渠道协商加密密钥。此外，如何为数字化的信息或文件提供一种类似于为书面文件手写签字的方法，也是对称密码体制难于解决的问题。1976 年，Diffie 和 Hellman 发表了“密码学中的新方向”（New Directions in Cryptography）一文，提出了公钥密码体制的观点，使密码学发生了一场变革。公钥密码体制很好地解答了前面两个问题。

在公钥密码体制中，公钥密码算法采用了两个相关密钥，将加密与解密分开，其中一个密钥是公开的，称为公钥，用于加密；另一个是用户专有的，因而是保密的，称为私钥，用于解密。公钥密码体制具有如下重要特性：已知密码算法和公钥，求解私钥，计算上是不可行的。公钥密码算法按下述步骤对信息实施保护，为了叙述方便，把信息发送方设为 A，信息接收方设为 B：

① 要求信息接收方 B 产生一对密钥 PK_B 和 SK_B，其中 PK_B 为公钥，SK_B 为私钥；接收方 B 将 PK_B 公开，而将 SK_B 秘密保存。

② A 要想向 B 发送信息 m，则使用 B 的公钥 PK_B 加密 m，表示为 $c = E_{PK_B}(m)$，其中 c 是密文。

③ B 收到密文 c 后，则用自己的私钥 SK_B 解密，表示为 $m = D_{SK_B}(c)$。

2．RSA 公钥密码算法

1977 年由 Rivest、Shamir 和 Adleman 提出了第一个比较完善的公钥密码体制 RSA。RSA 公钥算法的数学基础是初等数论中的 Euler（欧拉）定理，并建立在大整数因子分解的困难性基础之上。

算法描述如下：

（1）密钥对的生成

- 选择两个大素数，p 和 q；
- 计算 $n = pq$，以及其欧拉函数值 $\phi(n) = (p-1)(q-1)$；
- 随机选择一整数 e，要求 e 和 $\varphi(n)$ 互质；
- 利用欧几里德算法计算解密密钥 d，满足

$$ed = 1 \pmod{\phi(n)},$$

则 e 和 n 是公钥，d 是私钥；p 和 q 不再需要，可以被舍弃，但绝不可泄漏。

（2）加密

加密信息 M（采用二进制表示）时，首先把 M 分组，使得每个分组对应的十进制数小于 n，即分组长度小于 $\log_2 n$。然后对每个明文分组 m 作加密运算如下：

$$c = m^e \pmod{n}。$$

（3）解密

对密文的解密运算如下：

$$m = c^d \pmod{n}。$$

RSA 的安全性是基于大整数的素分解问题的难解性，目前尽管尚没有从理论上证明大整数的素分解问题是难解问题，但迄今还没有找到一个有效的分解算法，这是 RSA 的基础。如果 RSA 的模数 n 被成功分解为 pq，则立即可算出 $\phi(n)=(p-1)(q-1)$ 和 d，因此攻击成功。而且分解 n 也是攻击 RSA 最显然的方法。随着计算能力的不断提高和分解算法的进一步改善，原来认为不可能被分解的大数可以被成功分解，因此为了抵抗现有的整数分解算法，保证算法的安全性，对 p 和 q 的选取提出了以下要求：

① $|p-q|$ 很大，通常 p 和 q 的长度相差不大；

② $p-1$ 和 $q-1$ 分别含有大的素因子，且最大公约数尽可能小。

3．RSA 的攻击*

（1）RSA 的选择密文攻击

由 RSA 算法的加密变换可知，对一切 x_1，$x_2 \in Z_n$，有 $E_K(x_1x_2)=E_K(x_1)(x_2) \bmod n$ 成立，这个性质称为 RSA 算法的同态性质。选择密文攻击就是利用了这一性质，如果敌手知道密文 c_1、c_2 的明文 m_1、m_2，就知道了 $c_1c_2 \bmod n$ 对应的明文为 $m_1m_2 \bmod n$。

设用户 A 拥有一个 RSA 算法，模数为 n，其公钥为 e，私钥为 d。如果一个敌手希望用户 A 为其解密一个特定的密文 $c=m^e \bmod n$，并且如果用户 A 还为敌手解密除 c 外的任意密文。这样敌手随机选择一个非零整数 $x \in Z_n$，并计算 $\bar{c}=cx^e \bmod n$，敌手一旦将 $\bar{c}$ 交于用户 A，则 A 将为其解密计算 $\bar{m}=(\bar{c})^d \bmod n$，并将计算结果交给敌手，敌手通过如下计算即可得到明文 m：

$$m=\bar{m}x^{-1} \bmod n \equiv (\bar{c})^d x^{-1} \bmod n \equiv (cx^e)^d x^{-1} \bmod n。$$

因此，该攻击提醒人们在用 RSA 算法时，应注意破坏 RSA 算法的同态性质。

（2）RSA 的公共模数攻击

假定用户 A 有一个 RSA 算法，模数为 n，公钥为 e_1；用户 B 也有一个 RSA 算法，模数亦为 n，公钥为 e_2，并且 e_1 和 e_2 互素。若用户 C 想加密同一明文 m 发给 A 和 B，那么 C 先计算 $y_1=m^{e_1} \bmod n$ 和 $y_2=m^{e_2} \bmod n$。然后将 y_1 发给 A，将 y_2 发给 B，如果敌手截获了 y_1 和 y_2，那么他可用如下方法得到 m。因为 e_1 和 e_2 互质，故用欧几里得算法能找到 r 和 s，使得

$$re_1+se_2=1。$$

假设 r 为负数（因为 r 或 s 必有一个是负数），需再用欧几里得算法计算 y_1^{-1}，则

$$(y_1^{-1})^{-r} y_2^S = m \bmod n。$$

这说明，敌手解密 C 发送的密文是可能的。公共模数攻击告诫人们，不要在不同用户之间共享模 n。

---

* 此部分是选读内容。