网络空间安全系列丛书

密码算法应用实践

北京云班科技有限公司
文仲慧 何桂忠 周明波 等编著

非外借

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

网络空间安全系列丛书
工业和信息产业科技与教育专著出版资金资助出版

密码算法应用实践

北京云班科技有限公司
文仲慧 何桂忠 周明波 牛传涛 张海黎 编著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

信息安全对密码提出了越来越高的要求。作为密码学重要组成部分的密码体制，近十几年来取得了长足的发展。本书根据密码的加密原理，分为 4 章，即分组密码、Hash 函数、序列密码、P1363 公钥标准简介，通过图表的方式对每种加密算法的原理进行了展示，并给出了部分加密算法的伪代码，具有很强的实用性。

本书适合从事信息安全的人士阅读，也可作为相关专业的教材或教学参考书。

图书在版编目（CIP）数据

密码算法应用实践 / 文仲慧等编著. —北京：电子工业出版社，2019.9
（网络空间安全系列丛书）
ISBN 978-7-121-35390-1

I. ①密… II. ①文… III. ①密码算法 IV. ①TN918.1

中国版本图书馆 CIP 数据核字（2018）第 252765 号

责任编辑：田宏峰
印　　刷：三河市君旺印务有限公司
装　　订：三河市君旺印务有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　　邮编：100036
开　　本：787×980　1/16　　印张：15.75　　字数：352 千字
版　　次：2019 年 9 月第 1 版
印　　次：2019 年 9 月第 1 次印刷
定　　价：68.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：tianhf@phei.com.cn。

丛书编委会

前　言

目前，越来越多的重要信息是以电子的形式存储和交换的，信息安全对密码提出了越来越高的要求。作为密码学重要组成部分的密码体制，近十几年来取得了长足的发展，特别是进入20世纪90年代以来，单钥密码新体制大量涌现，公钥密码标准化得到了迅速的发展。由于目前难以找到一本系统、全面介绍各种密码算法的图书，为使有关研究人员能够及时掌握各种密码算法，了解密码算法的发展新动向，开阔视野、活跃思路，我们编著了《密码算法应用实践》。

本书共分4章。

第1章为分组密码。

分组密码是现代密码学中的一个重要研究分支，其诞生和发展有着广泛的实用背景和重要的理论价值。目前这一领域还有许多理论和实际问题有待继续研究和完善。这些问题包括如何设计可证明安全的密码算法、如何加强现有算法及其工作模式的安全性、如何测试密码算法的安全性、如何设计安全的密码组件等。

对于分组密码，早期的研究基本上是围绕DES进行的，推出了一些类似的加密算法，如LOKI、FEAL、GOST等加密算法。进入20世纪90年代，人们对DES的研究更加深入，特别是差分密码分析（Differential Cryptanalysis）和线性密码分析（Linear Cryptanalysis）的提出，使人们不得不研究新的密码结构。IDEA加密算法打破了DES的垄断局面，随后出现了SQUARE、SHARK、SAFER-64等加密算法，从理论上给出了最大差分特征概率和最佳线性逼近优势的界，证明了密码对差分密码分析和线性密码分析的安全性。

AES的征集掀起了分组密码研究的新高潮，15个AES候选加密算法反映了当前分组密码设计的水平，也可以说是近几年研究成果的一个汇总。

本章介绍的分组算法可以分为3类：第一类是采用菲斯特（Feistel）结构的分组密码，包括采用非平衡Feistel结构的分组密码和可看成Feistel结构扩展的SKIPJACK加密算法，以及采用Hash函数和其他密码函数构建的基于Feistel结构的分组密码；第二类是采用非Feistel结构的分组密码；第三类是AES征集的15个候选的加密算法。

由本章给出的分组密码的加密算法可以看出，分组密码的基本编码思想是由一些简单的基本变换构成层变换，通过多层迭代来达到混乱和扩散的目的的，从而构成强的密码变换。虽然加密的理论和技术又有了许多新发展，提出了一系列新思想、新技巧，但各种算法的研究和改进都是万变不离其宗，都是根据使用需求和实现环境，以期使单位时间的计算取得更

好的安全效益，是权衡加密算法密度和计算时间开销的折中。

分组密码加密的新思想和新发展表现为：一是在体系结构上更灵活多样；二是在基本运算选择上更多变快速；三是非线性变换的理论基础更趋完善；四是层密钥生成方法变化纷呈；五是分组密码与其他密码函数的内在联系越来越清晰；六是在密码中设置陷门的研究已深入到分组密码。

第 2 章为 Hash 函数。

20 世纪 70 年代以来，随着网络的逐步普及，人类社会步入信息化时代。自此，电子文件大量出现，其安全性一开始就令人担忧。电子文件的安全问题之一是电子文件的完整性。所谓完整性，主要是指电子文件是否有部分改动、删除或插入。在社会客观需求推动下，经过包括密码学家在内的大批学者的共同努力，创造性地设计出了可满足消息完整性验证的密码算法——Hash 函数。

本章主要介绍了典型的基于 Hash 函数的消息压缩值算法。

第 3 章为序列密码。

序列密码也称为流密码（Stream Cipher），是对称密码算法的一种。序列密码具有实现简单、便于硬件实施、加解密处理速度快、没有或只有有限的错误传播等特点，因此在实际应用中，特别是在专用或机密机构中保持着优势，典型的应用领域包括无线通信、外交通信。

分组密码以一定大小作为每次处理的基本单元，而序列密码则是以一个元素（一个字母或一个比特）作为基本的处理单元。序列密码是一个随时间变化的加密变换，具有转换速度快、低错误传播的优点，硬件实现电路更简单；缺点是低扩散（意味着混乱不够）、插入及修改的不敏感性。分组密码使用的是一个不随时间变化的固定变换，具有扩散性好、插入敏感等优点；缺点是加解密处理速度慢、存在错误传播。

序列密码涉及大量的理论知识，提出了众多的设计原理，也得到了广泛的分析，但许多研究成果并没有完全公开。这也许是因为序列密码目前主要应用于军事和外交等机密部门的缘故。目前，公开的序列密码加密算法主要有 RC4、SEAL 等。

本章主要介绍序列密码体制以及由此反映出来的加密新思想，主要包括：基于 Fibonacci 序列和 CA 构建的乱源发生器，FISH 加密算法和 PIKE 加密算法以模 232 的 Fibonacci 序列为乱源，是乘同余序列的推广，CA 是移位寄存器（或级间模二加移位寄存器）的推广，可看成是一种特殊的自动机；采用择多走/停的新步进方式的密码体制，PIKE 加密算法和 A5 加密算法的控制信息可来自乱源序列的某些位，也可来自信息的奇、偶校验位；各种新颖的加密技术，如面向软件实现的 RC4 加密算法，与常见的基于线性反馈移存器的序列密码不同，以一个相对比较大的代替表为基础，在自身的控制下，通过表中元素的对换进行缓慢的变化，同时生成随机数序列；序列密码许多比较成熟的密码分析方法难以直接用于 RC4 的密码分析；与 RC4 加密算法类似的 ISAAC 加密算法，但它不用对换，而是采用移位和算术加；

TwoPrime 加密算法吸取了分组密码的加密方法，以线性和非线性变换交替复合，以块为单位进行加乱，每一周期生成 64 比特的乱数序列；WAKE 加密算法和 SEAL 加密算法是速度极快的序列密码，用 5 条指令即可生成 1 字节的乱数。

第 4 章为 P1363 公钥标准简介。

公开密钥加密（Public-Key Cryptography）也称为非对称加密（Asymmetric Cryptography），是密码学发展史上的一次革命。公钥密码体制的编码系统是基于数学中的单向陷门函数。更重要的是，公钥密码体制采用了两个不同的密钥，对在公开的网络上进行保密通信、密钥分配、数字签名和认证有着深远的影响。

本章简要地介绍了 P1363 公钥标准，主要包括 RSA 方案、共同密钥生成方案、数字签名方案和校验方法，以及密钥交换算法 KEA。

本书在编写过程中，参考了大量关于密码算法的论文和标准，在此表示衷心的感谢。

鉴于笔者水平有限，加之时间仓促，本书难免会有不足和疏漏之处，敬请广大读者批评指正。

作　者

2019 年 7 月

目　录

第1章

分组密码

1.1 数据加密标准和分组密码的四种工作方式

1.1.1 DES加密算法的研制经过

DES（Data Encryption Standard）加密算法的研究历经6年的时间（1968年—1974年）。20世纪60年代末，IBM公司的一个研究组在菲斯特（Feistel）的带领下研制出了Lucifer密码。1971年，IBM公司请塔其曼（Tuchman）和迈耶（Meyer）在Lucifer密码的基础上进一步做研究，以增加密码强度。他们经过3年多的研究，试图破解Lucifer密码，通过搜索强有力的代替置换网络和密钥编制函数，完成了DES加密算法的设计和编制工作。

1976年年底，DES加密算法被采纳为联邦标准，自1977年7月15日起正式生效。1986年，NSA宣布停止执行DES计划。从1988年1月1日起，除了电子资金过户（Electronic Fund Transfer），不再批准政府部门使用采用DES加密算法的产品，但已批准的产品可继续销售和使用。美国安全局（National Security Agency，NSA）感到DES加密算法的广泛使用已使之成为众矢之的。

1.1.2 DES加密算法

1. DES加密算法的框架

DES加密算法如图1.1.1所示。输入明文块M_0（64比特），$M_0=m_0m_1\cdots m_{63}$，经初始置换IP（见表1.1.1）换位成$M_0=m_{57}m_{49}\cdots m_6$。

将M_0分成左右两个半块（32比特），$M_0=(L_0,R_0)$，(L_0,R_0)经过第1层变换后记为(L_1,R_1)，其中，$L_1=R_0$，$R_1=L_0\oplus f(R_0,K_1)$。

经过第i（$i=1,2,\cdots,16$）层变换后记为(L_i,R_i)。其中，$L_i=R_{i-1}$，$R_i=L_{i-1}\oplus f(R_{i-1},K_i)$，此处的$f$函数将在下面详述，$K_i$为密钥，经过16层变换后，预输出为$(R_{16},L_{16})$，再经$IP^{-1}$置换成密文块$C$。

2. f函数

f函数是DES加密算法的核心部分（见图1.1.2），进入第i层时，f函数的输入R_{i-1}（32

比特）先经扩张函数 E 变成 48 比特，设 $R_{i-1}=r_0r_1r_2\cdots r_{31}$ 经 E 扩张后记为 $T_{i-1}=E(R_{i-1})$。

$$T_{i-1}=t_0t_1t_2\cdots t_{47}$$

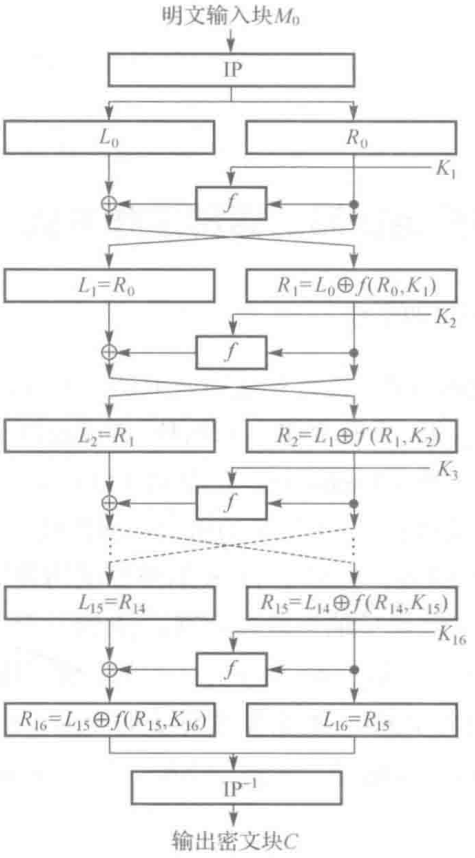

图 1.1.1　DES 加密算法

表 1.1.1　置换表 IP 和 IP^{-1}

IP							
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
56	48	40	32	24	16	8	0
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6

IP^{-1}							
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
32	0	40	8	48	16	56	24

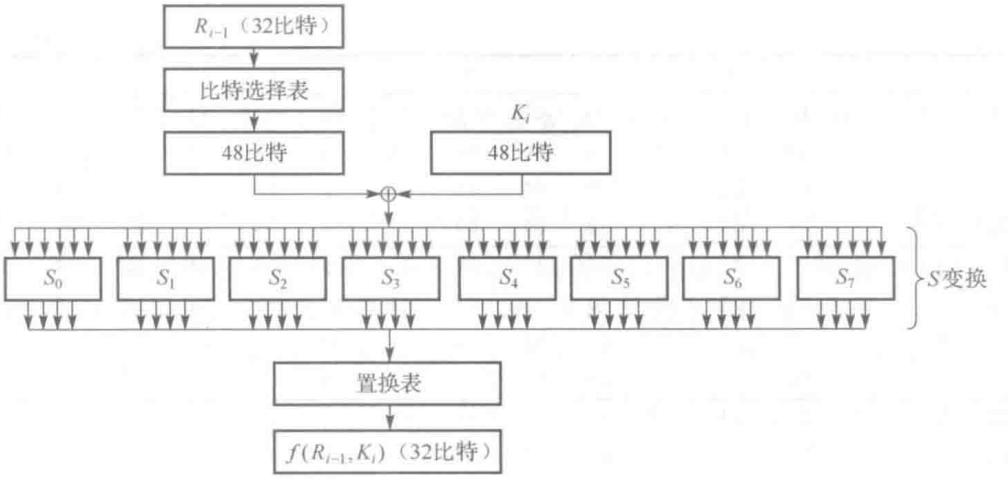

图 1.1.2　f 函数框图

参考表 1.1.2 的比特选择表和置换表，其中 $t_0=r_{31},t_1=r_0,t_2=r_1,\cdots,t_{46}=r_{31},t_{47}=r_0$。$T_{i-1}$ 与 K_i 的 48 比特按位进行模 2 加可得 $B=b_0b_1\cdots b_{47}$。B 顺序地按 6 比特分组，分别进入代替表（S 盒）$S_0,S_1,\cdots,S_7$，见表 1.1.3。

表 1.1.2　比特选择表和置换表

比特选择表					
31	0	1	2	3	4
3	4	5	6	7	8
7	8	9	10	11	12
11	12	13	14	15	16
15	16	17	18	19	20
19	20	21	22	23	24
23	24	25	26	27	28
27	28	29	30	31	0

置换表			
15	6	19	20
28	11	27	16
0	14	22	25
4	17	30	9
1	7	23	13
31	26	2	8
18	12	29	5
21	10	3	24

表 1.1.3　S 盒

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07	S_0
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08	
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00	
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13	
0	15	01	08	14	06	11	03	04	09	07	02	13	12	00	05	10	S_1
1	03	13	04	07	15	02	08	14	12	00	01	10	06	09	11	05	
2	00	14	07	11	10	04	13	01	05	08	12	06	09	03	12	15	
3	13	08	10	01	03	15	04	02	11	06	07	12	00	05	14	09	

续表

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0	10	00	09	14	06	03	15	05	01	13	12	07	11	04	02	08	S_2
1	13	07	00	09	03	04	06	10	02	08	05	14	12	11	15	01	
2	13	06	04	09	08	15	03	00	11	01	02	12	05	10	14	07	
3	01	10	13	00	06	09	08	07	04	15	14	03	11	05	02	12	
0	07	13	14	03	00	06	09	10	01	02	08	05	11	12	04	15	S_3
1	13	08	11	05	06	15	00	03	04	07	02	12	01	10	14	09	
2	10	06	09	00	12	11	07	12	15	01	03	14	05	02	08	04	
3	03	15	00	06	10	01	13	08	09	04	05	11	12	07	02	14	
0	02	12	04	01	07	10	11	06	08	05	03	15	13	00	14	09	S_4
1	14	11	02	12	04	07	13	01	05	00	15	10	03	09	08	06	
2	04	02	01	11	10	13	07	08	15	09	12	05	06	03	00	14	
3	11	08	12	07	01	14	02	13	06	15	00	09	10	04	05	03	
0	12	01	10	15	09	02	06	08	00	13	03	04	14	07	05	11	S_5
1	10	15	04	02	07	12	09	05	06	01	13	14	00	11	03	08	
2	09	14	15	05	02	08	12	03	07	00	04	10	01	13	11	06	
3	04	03	02	12	09	05	15	10	11	14	01	07	06	00	08	13	
0	04	11	02	14	15	00	08	13	03	12	09	07	05	10	06	01	S_6
1	13	00	11	07	04	09	01	10	14	03	05	12	02	15	08	06	
2	01	04	11	13	12	03	07	14	10	15	06	08	00	05	09	02	
3	06	11	13	08	01	04	10	07	09	05	00	15	14	02	03	12	
0	13	02	08	04	06	15	11	01	10	09	03	14	05	00	12	07	S_7
1	01	15	13	08	10	03	07	04	12	05	06	11	00	14	09	02	
2	07	11	04	01	09	12	14	02	00	06	10	13	15	03	05	08	
3	02	01	14	07	04	10	08	13	15	12	09	00	03	05	06	11	

代替的规则如下：输入 S_i 的 6 比特的左端 1 比特和右端 1 比特组成二进制数 0～3 中的某个数，作为取 S_i 的行数；中间 4 比特组成二进制数 0～15 中的某个数，作为取 S_i 的列数；在 S_i 表行列交叉处取得一个数，用 4 比特表示作为 S_i 盒的输出。例如，若输入 S_0 盒的 6 比特为“011011”，则行数为“01”，列数为“1101”，即 13。S_0 的第 1 行第 13 列的元素为 5，于是输出为“0101”，如下所示。

S_0 盒

	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	1	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	7	5	11	3	14	10	0	6	13

经过 S 盒后的 32 比特再经过置换表可改变比特位置，结果作为 f 函数的输出。

3. 子密钥发生器

64 比特的初始密钥 K 通过子密钥发生器变成 $K_1,K_2,\cdots,K_{16}$，分别作为 1 到 16 层的 f 函数子密钥（48 比特）。

在 64 比特的初始密钥 K 中，除去 8 比特（第 7、15、23、31、39、47、55、63 比特作为校验位），其余 56 比特送入置换 PC I，经过坐标置换后分成两组，每组 28 比特，分别送入 C 寄存器和 D 寄存器中。在各次迭代中，C 和 D 寄存器分别将存储的数按移位次数表进行循环左移。每次移位后将 C 和 D 寄存器的第 6、9、14、25 比特除去，其余比特经置换后 PC II 送出 48 比特，作为第 i 次迭代时所用的子密钥 K_i。子密钥发生器如图 1.1.3 所示。

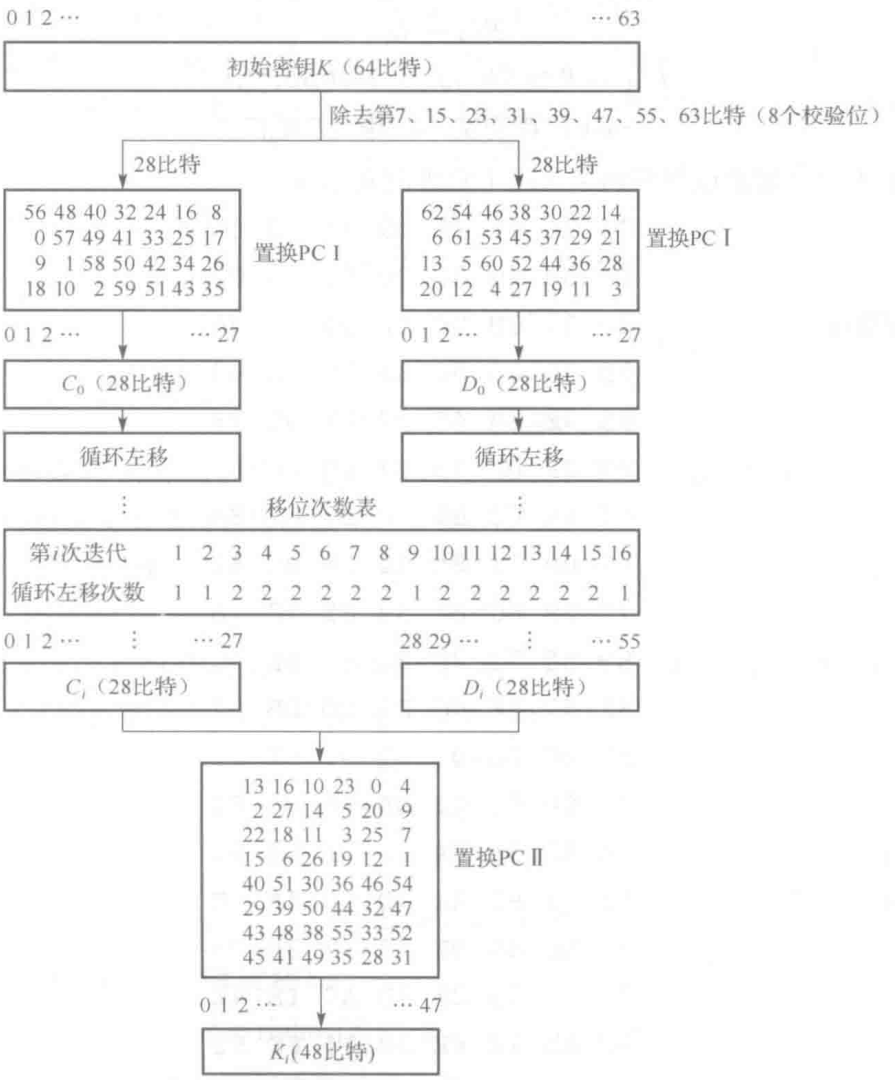

图 1.1.3　子密钥发生器

DES 加密算法可以简单归结如下。

加密过程：

$$L_0,R_0 \leftarrow \text{IP（64 比特输入）}$$

$$L_i \leftarrow R_{i-1}$$

$$R_i \leftarrow L_{i-1} \oplus f(L_i,\cdots,K_i),\ i=1,2,\cdots,16$$

$$\text{（64 比特密文）} \leftarrow \text{IP}^{-1}(R_{16}L_{16})$$

DES 的加密运算是可逆的，其解密过程与加密过程类似。

解密过程：

$$R_{16},L_{16} \leftarrow \text{IP（<64 比特密文>）}$$

$$R_{i-1} \leftarrow L_i$$

$$L_{i-1} \leftarrow R_i \oplus f(R_{i-1},K_i),\ i=16,15,\cdots,1$$

$$\text{<64 比特明文>} \leftarrow \text{IP}^{-1}(L_0R_0)$$

这里给出了一个加密过程的例子（用十六进制表示）：

密钥：	03 96 48 C5 39 31 39 65
明文：	00 00 00 00 00 00 00 00
经置换 IPRG：	00 00 00 00 00 00 00 00
第 1 层：	00 00 00 00 85 7E 2A 43
第 2 层：	85 7E 2A 43 D7 2F 0D 7B
第 3 层：	D7 2F 0D 7B C7 6E 6C B1
第 4 层：	C7 6E 6C B1 4C B0 77 8A
第 5 层：	4C B0 77 8A 72 2B BC 81
第 6 层：	72 2B BC 81 59 85 72 7B
第 7 层：	59 85 72 7B 82 67 AE 9C
第 8 层：	82 67 AE 9C E7 DD DB 94
第 9 层：	E7 DD DB 94 71 90 0F 11
第 10 层：	71 90 0F 11 0A AD 33 E4
第 11 层：	0A AD 33 E4 51 61 B2 81
第 12 层：	51 61 B2 81 7D DD 4A 9E
第 13 层：	7D DD 4A 9E 75 17 39 28
第 14 层：	75 17 39 28 9D A0 1E 4E
第 15 层：	9D A0 1E 4E BB 14 FC F2
第 16 层：	73 6A 7F 8A BB 14 FC F2
经过 IP^{-1} 后的密文：	C4 D7 2C 9D EE DE 5E 8B