基于 C 语言的 DES 加密算法密文安全性的研究

◆姚若禹　陈坤　顾子阳　马培森　吴佩　赵骋宇

（南京工程学院 电力工程学院　江苏　211100）

摘要：当今信息安全日趋重要，信息加密技术被广泛应用。从技术层面上说，密码学始终是信息安全的一个核心技术，而对称密码体制中的 DES 算法一直以来都作为数据加密的标准，其加密的安全性则是检测算法的主要指标。本文主要使用 C 语言对 DES 加密算法密文安全性进行研究。

关键词：DES 加密算法；密钥生成；C 语言；信息加密

基金项目：南京工程学院电力工程学院基金（编号 TB202304014）；南京工程学院本科生科技创新基金项目（项目号 TB202304014）

随着信息技术的迅速发展，大量信息通过数字网络进行传输，而数字信息的安全问题就愈发值得重视。数字信息安全不仅仅关系到个人与企业的隐私信息，更是关系到社会及国家的重要机密，其加密算法的安全性不可懈怠。DES 算法作为对称加密的经典代表，在信息安全领域有着重要的地位和应用。到目前为止，除了使用穷举法搜索其密钥空间寻找破译密码外，没有更有效的办法[1]。该算法在创立之初就凭借其简单性、高度安全性和广泛适应性备受很多个人及企业的青睐，成为众多领域的信息安全保障。

1　密码学原理

密码学是数学的一个分支，是研究密码技术的重要学科，具有保障信息安全的核心作用。密码编码学又分为对称密码学和非对称密码学。其中非对称密码学运算复杂，需要相当多的资源，并不适合在存储卡中使用。而对称密码的加密和解密双方使用相同的密钥，适合广泛使用。对于密码学的研究主要涉及两个方面：加密和解密技术。加密技术属于设计层面，是将明文转化为密文，以便在传输时保护信息不被窃取或篡改；解密技术属于分析层面，是将密文转化为明文，以便信息接收方能正常理解。

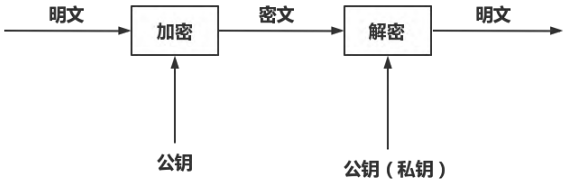

图 1 密码系统模型

在信息技术的支撑下，现代密码学理论蓬勃发展，密码算法设计与分析相互促进，出现了以 DES 为代表的对称密码体制和 RSA 为代表的非对称密码体制，制定了许多通用的加密标准，促进网络和技术的发展。密码学已经成为当今信息时代网络安全的重要支撑之一，它的应用范围广泛，从普通的数据传输到电子商务、金融、军事等领域的安全性问题都具有深远的影响[2]。

2　DES 加密算法原理

2.1 置换过程

DES 加密过程中涉及大量的盒置换过程。以 IP 置换为例，IP 置换表的前三位数据为 58、50、42，其意为将输入数据块的第 58、50、42 位的数据分别换到第 1、2、3 位上。其他置换也是同理。这些置换能对数据进行重新排列，起到打乱的作用。

2.2 子密钥生成

子密钥的生成包含置换和移位过程。用户提供 64 位的主密钥，将第 8，16，24，32，40，48，56，64 位作为校验位，不参与运算。PC-1 置换后剔除这 8 位校验位，剩余参与运算的 56 位密钥平均分为 C_0，D_0 两组，每组 28 位，通过移位次数表分别进行左移位后得到 C_1，D_1 两组，合并为 56 位密钥，经 PC-2 置换后变为 48 位的子密钥 K_1，即为 S 盒加密第一轮循环所要使用的密钥。重复以上步骤可以获得各轮循环需要使用的密钥[3]。

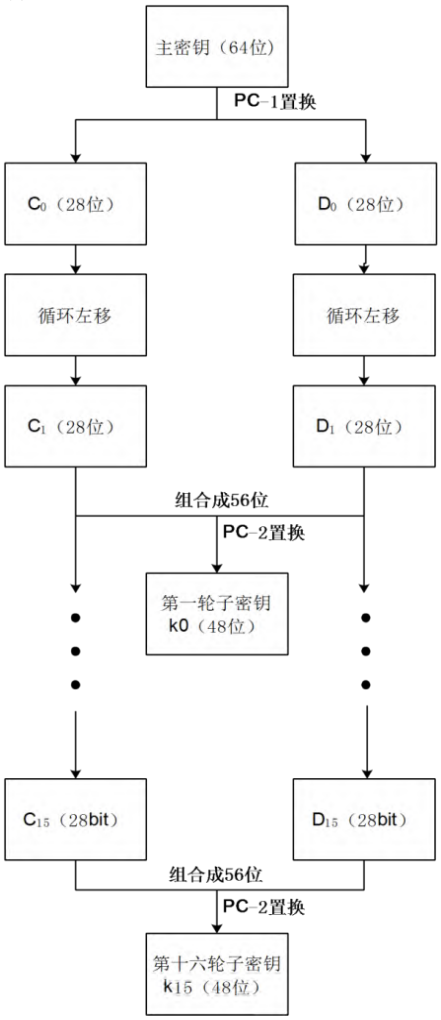

图 2 子密钥生成示意图

2.3 初始置换（IP 置换）

加密或解密前，需要对数据进行重新排列。将加密的数据以 64 位为一组分割成若干组数据，按照 8 行 8 列进行排列，称为明文。然后使用固定的 IP 置换表对 64 位的二进制明文块进行重新排列。

表 1　IP 置换表

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

2.4 轮函数运算

初始置换后得到打乱顺序的明文，平均分为两组：L_0（32 位），R_0（32 位），对 R_0 进行 E 盒扩展置换得到 $R_0^{'}$（48 位），与第一轮的轮密钥 K_0（48 位）的每一位进行异或运算得到 48 位输出，将该 48 位输出平均分为 8 组，每组 6 位，输入到 S 盒中进行代换。这 8 组数据的最终代换结果是 32 位数据，将结果与 L_0 的每一位异或得到的结果作为 R_0 的最终结果。此时将 L_0 与 R_0 数据互换得到 L_1 与 R_1 然后开始下一轮循环。循环 16 次后得到 L_{15} 和 R_{15}。

2.5 逆初始置换（FP 置换）

DES 加密的最后一步是逆初始置换，它将最后一次轮函数的结果 L_{15} 和 R_{15} 组合成 64 位数据进行 FP 置换，最终得到密文输出。

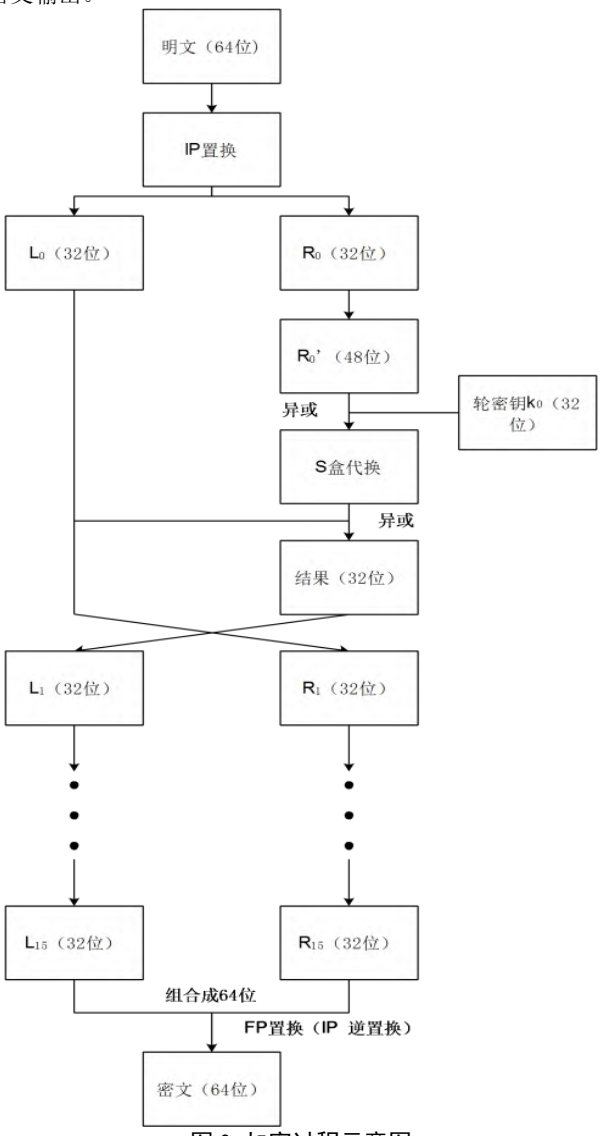

图 3 加密过程示意图

表 2　FP 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

3　C 语言实现

对 DES 加密算法的几大板块分别编写程序实现功能，其中包括置换板块、密钥生成板块、轮函数板块。[4]

3.1 置换

置换是按照特定的顺序打乱一个盒的过程，其代码如下：

```
void displace（int a[],int box[],int length_a,int length_box）
{
int copy[64]={0};
for（int i=0;i<length_a;i++）copy[i]=a[i];
for（int i=0;i<length_box;i++）a[i]=copy[box[i]-1];
}
```

3.2 密钥生成

3.2.1 移位次数表

密钥生成时，除了盒置换操作以外还需要使用移位次数表，其代码如下：

```
int move[16]={1,1,2,2,2,2,2,2,1,2,2,2,2,2,2,1};
    void key_move（int KEY[],int KEY_length,int movedistence）
{
for（int i=0;i<movedistence;i++）
{
int temp=KEY[0];
for（int j=1;j<KEY_length;j++）
{
KEY[j-1]=KEY[j];
}
KEY[KEY_length-1]=temp;
}
}
```

3.2.2 轮密钥生成

使用总密钥及移位次数表生成轮密钥，其代码如下：

```
void key_round_generate（int KEY[],int k[][48],int KEY_length）
{
int copy[64]={0};
for（int i=0;i<KEY_length;i++）copy[i]=KEY[i];
displace（copy,PC_1_box,64,56）;
for（int i=0;i<16;i++）
{
key_move（copy,28,move[i]）;
key_move（&copy[28],28,move[i]）;
int copy_copy[56]={0};
for（int j=0;j<56;j++）copy_copy[j]=copy[j];
displace（copy_copy,PC_2_box,56,48）;
for（int j=0;j<48;j++）k[i][j]=copy_copy[j];
}
}
```

其中主密钥为：int

```
Key[64]={0,0,0,1,0,0,1,1,0,0,1,1,0,1,0,0,0,0,1,0,1,0,1,1,1,0,1,1,1,1,
0,0,1,1,0,0,1,1,0,1,1,1,0,1,1,1,1,0,0,1,1,0,1,1,1,1,1,1,1,1,0,0,0,1
};
```

3.3 轮函数

轮函数为 DES 的核心模块，加密和解密都通过此模块来实现，其代码如下：

```
void roundfunction（int box[],int round_number）
{
int left[32],right[32],right_extend[48];
for（int i=0;i<32;i++）
{
left[i]=box[i];
right[i]=box[i+32];
right_extend[i]=box[i+32];
}
displace（right_extend,e_box,32,48）;
for （ int i=0;i<48;i++ ） right_extend[i]=xor_
（right_extend[i],key_round[round_number][i]）;
for（int i=0;i<8;i++）
{
int
temp=s_box[round_number][right_extend[i*6+0]*2+right_exten
d[i*6+5]*1][right_extend[i*6+1]*8+right_extend[i*6+2]*4+righ
t_extend[i*6+3]*2+right_extend[i*6+4]*1];
right[i*4+3]=temp%2;
temp=temp/2;
right[i*4+2]=temp%2;
temp/=2;
right[i*4+1]=temp%2;
temp/=2;
right[i*4+0]=temp%2;
}
displace（right,p_box,32,32）;
for（int i=0;i<32;i++）right[i]=xor_（right[i],left[i]）;
swapbox（left,right）;
for（int i=0;i<32;i++）
{
box[i]=left[i];
box[i+32]=right[i+32];
}
}
```

3.4 调用函数，生成密文

在主函数中调用这些函数，输入明文后得到密文，即为 DES 加密的结果。

```
明文：00001001110001010110000100011110001010101100111111100011101100 11
密文：10011001010011010011011000010100101111110000011001011000000101001
明文：10000101010011110110011110011101101010010001000000011101110101100
密文：11111000010000100101101001001011100100100001110000000100111101100
```

4　结果分析

```
明文：0000101001001000010001010111100111110010110001100111010100101010
密文：0010011000101100110110010110010101010101001100110010001100011001
```

以这对明文和密文为例，使用软件绘制出其时序波形图。

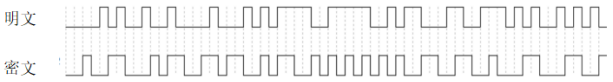

图 4　时序波形图

加密效果如上图密文所示。由此得出密文与明文的关联性不大。为了定量描述明文与密文的关联性，我们定义一个变量：

$$\text{相同率} = \frac{\text{明文与密文相同位数}}{\text{总位数}} \times 100\%$$

借助程序随机生成 1000 组明文与密文，计算其相同率。对于这 1000 组数据产生的 1000 个相同率，取 10 个为一组计算平均值，并绘制了如下的散点图：

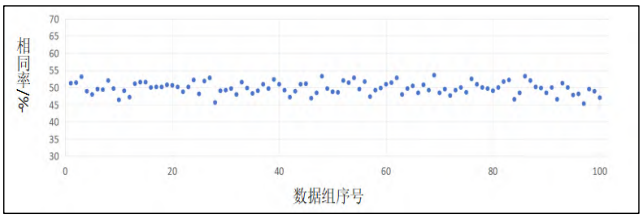

图 5　相同率分布散点图

由于明文与密文均为二进制数表示，故相同率越接近 50%，明文与密文的直接关联度越低，也就是密文对于明文的随机度越高。由此得出 DES 加密算法具有较强的安全性。

5　结束语

综上所述，DES 算法作为一种比较常用的加密算法，具有良好的安全性。其计算速度快，加密和解密使用同一种算法，便于学习和应用[5]。如今传统的 DES 加密算法虽然已被攻破，但其加密思路仍可为其他算法提供借鉴，对该算法的研究也必不可少。

参考文献：

[1]周明全 等.网络信息安全技术（第 2 版）[M].西安电子科技大学出版社，2010：55-57.

[2]徐强，宋依青.m 序列在信息安全中的应用[J].常州工学院学报，2005（01）：34-38.

[3]管莹，敬茂华.DES 算法原理及实现[J].电脑编程技巧与维护，2009（04）：5-7+13.

[4]贾伟，朱磊.DES 加密算法在网络通信中的实现[J].网络安全技术与应用，2020（03）：34-36.

[5]张乐.基于 DES 和 RSA 加密技术的大数据加密传输技术的算法研究[J].无线互联科技，2022，19（18）：125-127.

基于 AES 算法的医院数字化管理信息加密方法研究

◆郭妍

（北京市和平里医院　北京　100000）

摘要：随着医院数字化管理的普及和发展，加密方法在保护医院管理信息安全方面变得越来越重要。本研究以 AES（Advanced Encryption Standard）算法为基础，旨在提出一种有效的加密方法来保护医院数字化管理信息的机密性。首先，对医院数字化管理信息进行分类和分析，确定需求和加密目标。随后，通过对 AES 算法的原理和特性进行研究，设计了一套适用于医院数字化管理信息的加密方案。该方案包括密钥生成、数据加密和解密等步骤，采用 AES 算法的高强度加密和解密机制确保了信息的安全性。最后，通过对加密方法进行实验评估和性能分析，验证了该方法的有效性和可行性。本研究的结果对于提高医院数字