DES算法程序设计报告

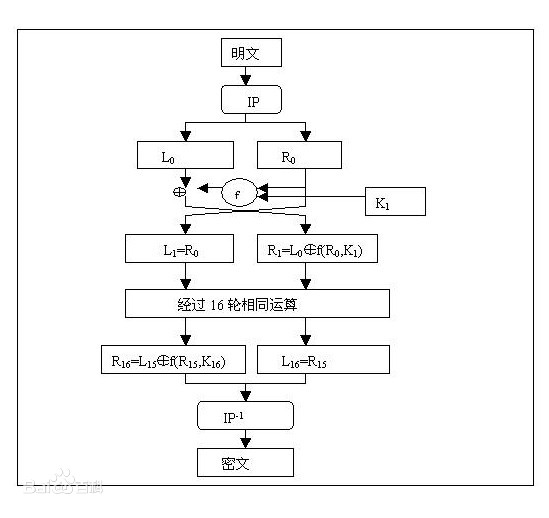
1. 算法原理概述

DES(Data Encryption Standard)算法为密码体制中的对称密码体制，又被称为美国数据加密标准，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。

其入口参数有三个：key、data、mode。

* key：加密解密使用的密钥，56位。
* data：为加密解密的数据，64位。
* mode：DES工作模式，加密/解密。

1. 总体结构



**DES算法的主要流程**

**加密**

1. 初始置换IP

给定64位明文块M,通过一个固定初始置换IP重排M中的二进制位，得到二进制串M0。将IP置换后的结果分为两部分，前32位为L0，后32位为R0。

1. 迭代T（16轮）

将L0，R0按以下规则进行16轮迭代

Li=Ri-1

Ri=Li-1⊕f(Ri-1,Ki)

i=1…16

其中，f(Ri-1,Ki)轮函数原理如下：

1. 长度为32位的串Ri-1作E-扩展得到48位的串E（Ri-1）
2. E（Ri-1）与长度为48位的子密钥Ki作二进制异或运算
3. 将2）得到的结果均分成8个分组，每个分组长度6位。分别经过8个不同的S-盒进行6-4转换，得到8个长度为4位的分组，顺序连接得到长度为32位的串。
4. 经过P置换得到32的结果输出。

子密钥Ki的生成：

给定64位密钥K，生成16个48位子密钥K1-K16

1. 对K的56个非校验位实行置换PC-1，将结果分为前28位和后28位C0D0
2. 分别对C0D0 进行循环左移[Ci=LSi（Ci-1），Di=LSi（Di-1）]后，对56位的CiDi压缩（去掉第9，18，22，25，35，38，43，54位）后进行PC-2置换得到48位的Ki



1. 交换置换W

将L16R16左右交换输出R16L16

1. 逆置换IP-1

对R16L16 使用初始置换IP的逆置换IP-1得到密文C

**解密**

与加密过程一样，只是迭代过程中解密的子密钥按K16，K15，…，K1次序调度

1. 模块分解
2. **int** Char8ToBit64(**char** ch[8],**char** bit[64]);
3. **int** Bit64ToChar8(**char** bit[64],**char** ch[8]);
4. **int** GenerateSubKeys(**char** key[64],**char** subKeys[16][48]);
5. **int** PC1\_Transform(**char** key[64], **char** tempbts[56]);
6. **int** PC2\_Transform(**char** key[56], **char** tempbts[48]);
7. **int** RSL(**char** data[56], **int** time);
8. **int** IP\_Transform(**char** data[64]);
9. **int** IP\_1\_Transform(**char** data[64]);
10. **int** E\_Transform(**char** data[48]);
11. **int** P\_Transform(**char** data[32]);
12. **int** S\_BOX(**char** data[48]);
13. **int** XOR(**char** R[48], **char** L[48],**int** count);
14. **int** Swap(**char** left[32],**char** right[32]);
15. **int** EncryptBlock(**char** plainBlock[8], **char** subKeys[16][48], **char** cipherBlock[8]);
16. **int** DecryptBlock(**char** cipherBlock[8], **char** subKeys[16][48], **char** plainBlock[8]);
17. **int** EncryptFile(**char** \*plainFile, **char** \*keyStr,**char** \*cipherFile);
18. **int** DecryptFile(**char** \*cipherFile, **char** \*keyStr,**char** \*plainFile);

* 子密钥生成（GenerateSubKeys）：

1. PC1\_Transform 64位密钥的56个非校验位进行PC-1置换
2. 16轮迭代产生16个子密钥
3. RSL 前后28位分别循环左移
4. PC2\_Transform 压缩成48位后进行PC2置换

* 主流程

1. 对64位明文进行初始置换IP（IP\_Transform）
2. 迭代T（重复16次）
3. 对R进行E-扩展（E\_Transform）
4. 结果与子密钥异或（XOR）
5. 结果分成8个分组，经过不同的8个S盒进行6-4转换，顺序连接为32的串（S\_BOX）
6. 结果经过P-置换（P\_Transform）
7. 上述32位数据与L异或结果赋给L
8. 将L,R左右交换（Swap）
9. 将L,R左右交换（Swap）
10. 对交换后的二进制串使用逆置换IP-1(IP\_1\_Transform)
11. 数据结构设计

* 将所有已知数据用int数组存储

1. 初始置换表IP
2. 逆置换表IP-1
3. E-扩展置换表
4. P-置换表
5. S-盒
6. 置换表PC-1
7. 置换表PC-2
8. 循环左移每轮移动的次数表

* 输入数据与输出数据

ascii码明文/密文,存储在char中

转化为64位比特进行DES加密解密过程

1. C 语言源代码

见附件

1. 编译运行结果

