

# DES加密算法

---

基于DES加密的TCP聊天程序

## 所需参数

---

- key:8个字节的密钥
- data:所需加密或解密的数据
- mode:加密或者解密

## 步骤

---

### 一、初始置换和逆置换

- 初始置换：明文以64位分组，每组将64位明文按照置换表进行置换，后分成左右两部分L0和R0。
- 逆置换：位于加解密的最后一步，目的是打乱原有64位的顺序。

### 二、f函数

1. 第一步，将32位数据根据扩展表扩展为48位。
2. 第二步，两者异或作为S盒的输入。
3. 第三步，将第二步得到的48位结果分为8组，每组6bit，查找S盒置换。

**PS:**6位中高低2位作为行数，中间4位作为列数，查找对应值即为替换后的值，行列数是从0开始的。

4. 最后一步，将S盒压缩后的32位数进行P置换，得到f函数结果。

### 三、子密钥生成

1. 第一步去掉奇偶校验位，64位密钥根据PC1表置换为56位。
2. 第二步 上一步得到的56位结果分为左右各28位，根据循环移位表分别进行循环移位，并拼接。
3. 第三步，结果再按PC2表进行压缩置换，去掉某些位，得到48位子密钥。
4. 循环16次，得到16轮的密钥。

### 四、加密

1. 第一步，初始置换IP，位数是从左往右的，最左为第一位。
2. 第二步，将64位明文分成左右各32位。
3. 第三步，进行16轮迭代。
4. 第四步，合并left和right。
5. 最后一步，逆初始置换 $y=IP^{-1}(L16R16)$ 。

### 五、解密

- 逆序使用子密钥，与加密相同的操作。

### 六、目录结构

```
| -tcp-des-chat
|   -des-src          //DES算法实现
|   -Const.h         //一些DES用到的常量，如置换表
|   -Des.h           //封装的DES类，供外部调用加密和解密操作
|   -Des.cpp          //DES类成员的实现
|   -main.cpp         //DES加解密功能测试
|   -Makefile         //使用make编译Des模块，执行main
| -config.h //客户端和服务端共用的函数和常量
|   -tcp-client.cpp //客户端实现
|   -tcp-server.cpp //服务端实现
|   -Makefile        //make编译链接生成client和server可执行程序
```

## 七、参考文章

- [https://blog.csdn.net/qq\\_27570955/article/details/52442092](https://blog.csdn.net/qq_27570955/article/details/52442092)
- <https://blog.csdn.net/lisonglisonglisong/article/details/41777413>