TANGRAM测试报告

本次算法实现与测试针对TANGRAM算法，根据分组长度和密钥长度分为128-128,128-256,256-256三个模块。

在初步实现算法并根据测试向量进行测试后，从两个角度对其进行优化：

1. 将原有循环移位的操作，从“c[1] = (b[1] << 1) | ((b[1] & 0x8000000000000000) >> 63)”优化为使用命令：\_rotl,\_rotr,\_rotl64,\_rotr64.
2. 将原有通过移位操作将unsigned char 8比特数拼接成32比特或64比特数，优化为直接利用地址来寻址赋值。

经过测试，上述两种优化均对算法效率有提升。

在Debug模式下，对TANGRAM128-128进行优化提升测试。

测试环境：

|  |  |  |  |
| --- | --- | --- | --- |
| 测试项目 | 未优化 | 使用\_rotl后 | 宏定义地址寻址赋值 |
| 加密1byte数据所需的cycle数 | 270.421915 | 250.248817 | 237.272311 |
| 1秒加密数据量 | 123.582005 Mbit | 112.052634 Mbit | 127.432194 Mbit |

在Release环境下对128-128,128-256,256-256分别进行效率测试：

|  |  |  |  |
| --- | --- | --- | --- |
| 测试项目 | 128-128 | 128-256 | 256-256 |
| 加密1byte数据所需的cycle数 | 24.580097 | 26.716248 | 21.284009 |
| 1秒加密数据量 | 654.216534 Mbit | 628.163438 Mbit | 785.224287 Mbit |
| 解密1byte数据所需的cycle数 | 25.676590 | 28.594542 | 22.024623 |
| 1秒解密数据量 | 678.356390 Mbit | 626.525013 Mbit | 782.154055 Mbit |