测试结果

—— written by 荔枝

小组成员：张芷芮、刘俐莹
联系方式：1635487611@qq.com

第 1 关：基本测试

【测试要求】根据 S-AES 算法编写和调试程序，提供 GUI 解密支持用户交互。输入可以是 16bit 的数据和 16bit 的密钥，输出是 16bit 的密文。

【测试用例】

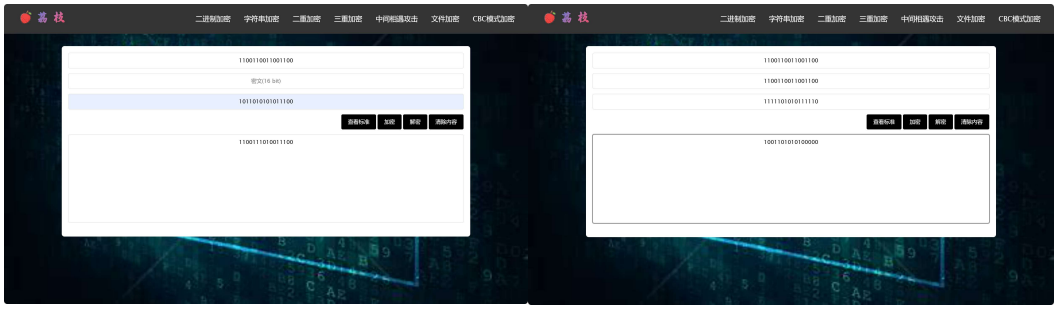
加密：明文：1100110011001100　　密钥：1011010101011100

解密：密文：1100110011001100　　密钥：1111101010111110

【测试结果】

加密结果：1100111010011100

解密结果：1001101010100000



可以看出，基础加解密功能无误，可以正常进行运算。

第 2 关：交叉测试

【测试要求】考虑到是"算法标准"，所有人在编写程序的时候需要使用相同算法流程和转换单元(替换盒、列混淆矩阵等)，以保证算法和程序在异构的系统或平台上都可以正常运行。

设有 A 和 B 两组位同学(选择相同的密钥 K)；则 A、B 组同学编写的程序对明文 P 进行加密得到相同的密文 C；或者 B 组同学接收到 A 组程序加密的密文 C，使用 B 组程序进行解密可得到与 A 相同的 P。

我们与叶咯吩组进行了交叉测试，分别使用了如下三种测试方式，测试用例和测试结果如下（A 组为荔枝，B 组为叶咯吩）：

【测试用例】

密钥 K：1011010101011100

1、同时对明文加密

明文：1010101010101010

2、A 组加密，B 组解密

明文：1111111111111111　密文：1100001000000011

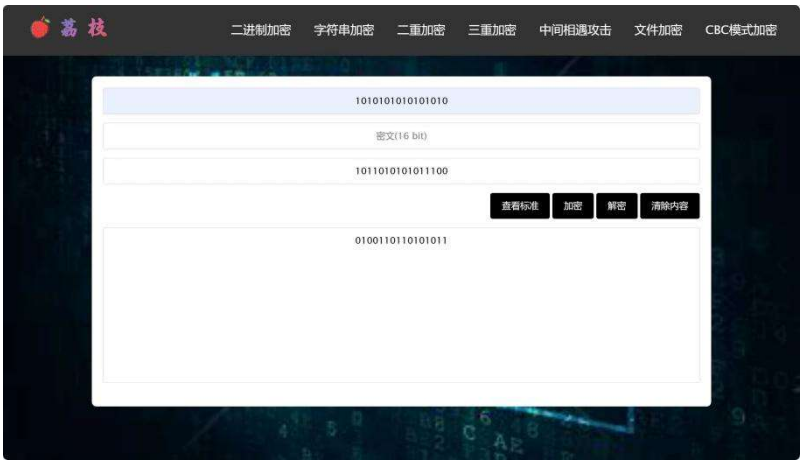
3、B 组加密，A 组解密

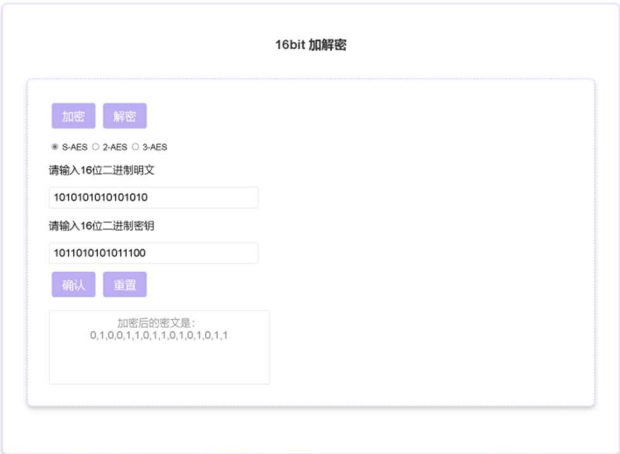
明文：0000000000000000　密文：0010010101101100

【测试结果】

1、A、B 两组选择相同的密钥，同时对明文 P 进行加密：

可以发现，两组的加密结果均为 0100110110101011。说明测试成功，两组加密算法无误



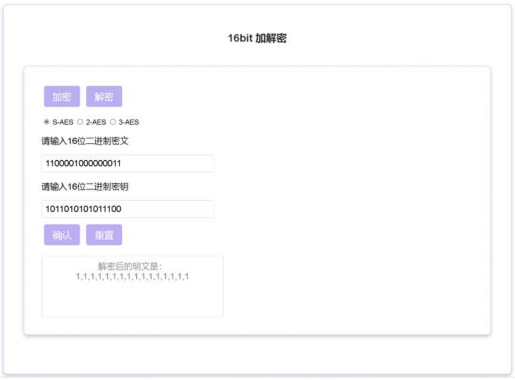


2、A 组对明文进行加密，B 组用 A 组的加密结果进行解密：

可以发现，B 组的解密结果与明文一致，说明测试成功，B 组解密算法无误

加密过程（荔枝组）：

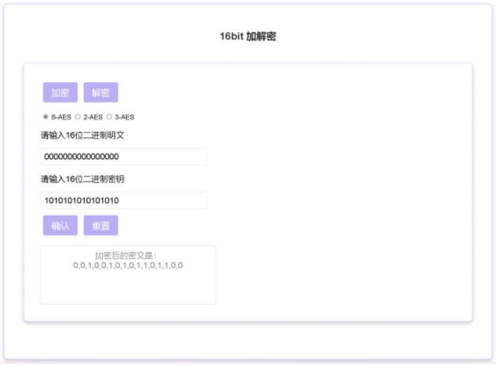
解密过程（卟咯吩组）：



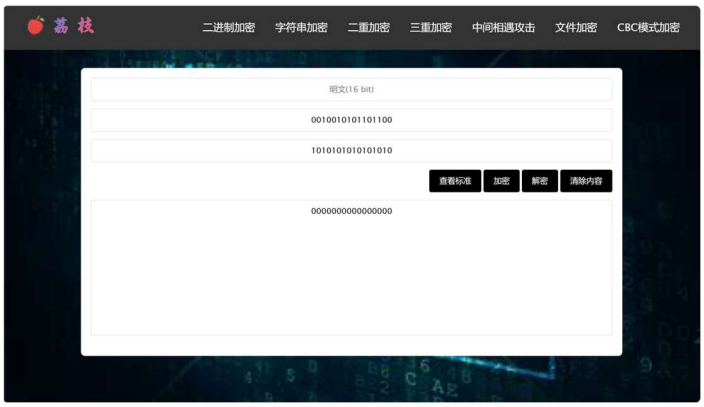
3、B 组对明文进行加密，A 组用 B 组的加密结果进行解密：

可以发现，A 组的解密结果与明文一致，说明测试成功，两组加解密算法均无误

加密过程（卟咯吩组）：



解密过程（荔枝组）：



第 3 关：扩展功能

【测试要求】考虑到向实用性扩展，加密算法的数据输入可以是 ASII 编码字符串(分组为 2 Bytes)，对应地输出也可以是 ACII 字符串(很可能是乱码)。

【测试用例】

加密：明文：ab　　密钥：0100010111000011

解密：密文：1011111111111011　　密钥：0100010111000011

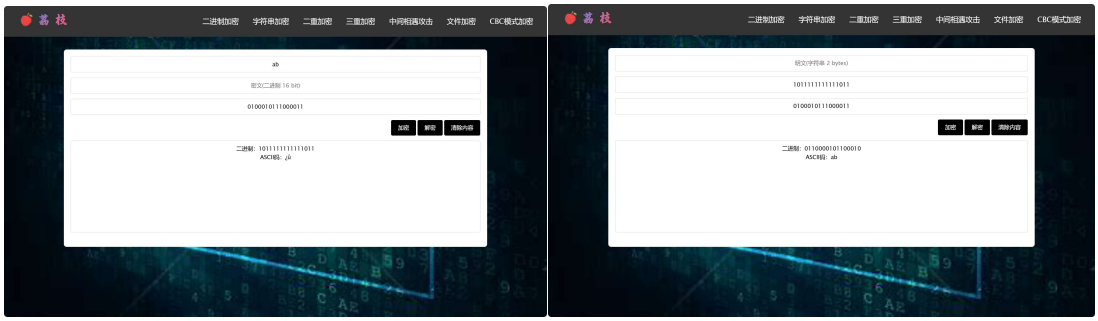
【测试结果】

加密结果：二进制：1011111111111011

ASCII 码：¿û（乱码无法完全显示）

解密结果：二进制：0110000101100010

ASCII 码：ab



可以看出，扩展 ASCII 码加解密功能无误，可以正常进行运算。

第 4 关：多重加密

4.1 双重加密

【测试要求】将 S-AES 算法通过双重加密进行扩展，分组长度仍然是 16 bits，但密钥长度为 32 bits。

【测试用例】

双重加密算法 1：

明文：1111000011110000

密文：0110101100010001

密文：11110000111100001111000011110000111100001111000011110000111

双重加密算法 2：

明文：1111110100110111

密文：1101010110000110

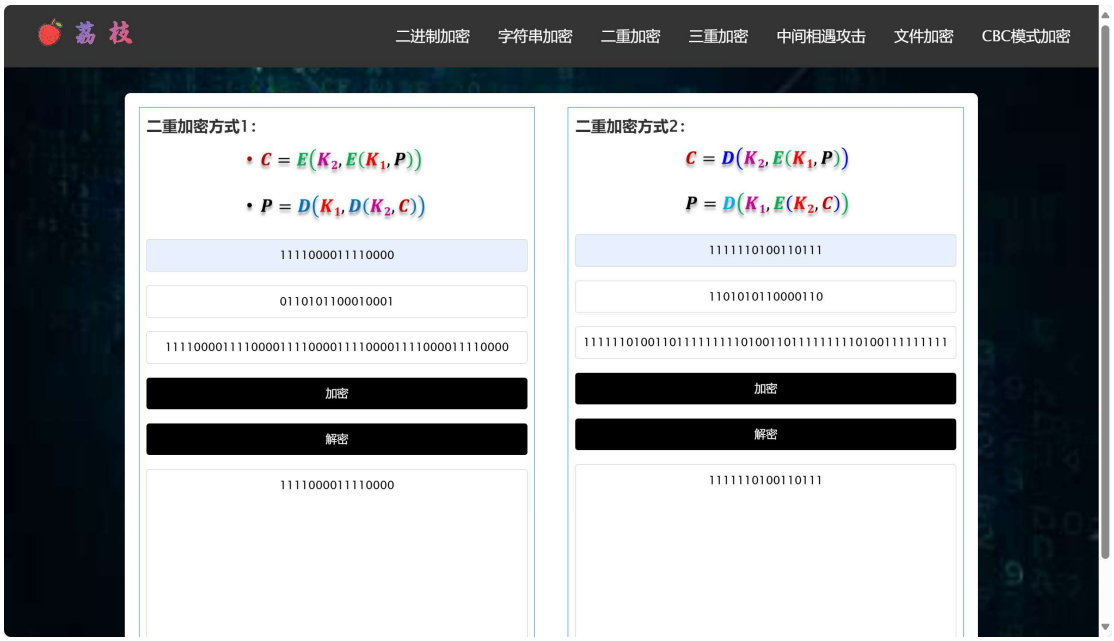
密钥：11111101001101111111111010011011111111101001111111111

【测试结果】

【左右两边使用两种加解密逻辑，加解密原理在公式中展示】

加密过程：

解密过程：



可以发现，加解密结果能够自洽（加密结果为密文，解密结果为明文），双重加密算法测试成功。

4.2 中间相遇攻击

【测试要求】假设你找到了使用相同密钥的明、密文对(一个或多个)，请尝试使用中间相遇攻击的方法找到正确的密钥 Key(K1+K2)。

【测试用例】

输入明密文对如下

明文：1001100110011011　　密文：1011011010010110

【测试结果】

中将相遇攻击结果：

找到的密钥组合为: 00011011101000111100101101101101

接下来在二重加密中验证攻击结果：

使用得到的密钥进行加密得到密文与原密文相符；

使用得到的密钥进行解密得到明文同样与原明文相符。

4.3 三重加密

【测试要求】将 S-AES 算法通过三重加密进行扩展，下面两种模式选择一种完成：

(1)按照 32 bits 密钥 Key(K1+K2)的模式进行三重加密解密

(2)使用 48bits(K1+K2+K3)的模式进行三重加解密。

【测试用例】

(1)按照 32 bits 密钥 Key(K1+K2)的模式进行三重加密解密

明文：1100110011001100

密文：1011101011110001

密钥：01000101110000110100010111000011

(2)使用 48bits(K1+K2+K3)的模式进行三重加解密。

三重加密算法 1：

明文：1111000011110000

密文：1111110100110111

密钥：111100001111000011110000111100001111000011110000

三重加密算法 2：

明文：1111110100110111

密文：1011011101001110

密钥：111111010011011111111101001101111111110100111111

【测试结果】

(1)按照 32 bits 密钥 Key(K1+K2)的模式进行三重加密解密

加密过程：

解密过程：

(2)使用 48bits(K1+K2+K3)的模式进行三重加解密。

【左右两边使用两种加解密逻辑，加解密原理在公式中展示】

加密结果：

解密结果：

可以发现，加解密结果能够自洽（加密结果为密文，解密结果为明文），三重加密算法测试成功。

第 5 关：工作模式

【测试要求】基于 S-AES 算法，使用密码分组链(CBC)模式对较长的明文文本消息进行加密。注意初始向量(16 bits) 的生成，并需要加解密双方共享。

在 CBC 模式下进行加密，并尝试对密文分组进行替换或修改，然后进行解密，请对比篡改密文前后的解密结果。

【测试用例】

明文：Hello S-AES and CBC!

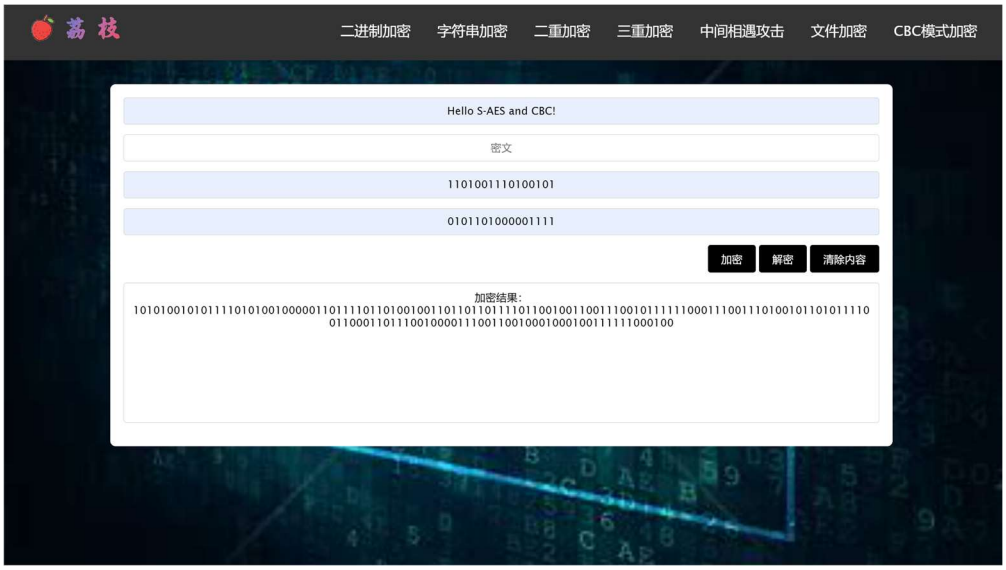
密钥：1101001110100101

初始向量：0101101000001111

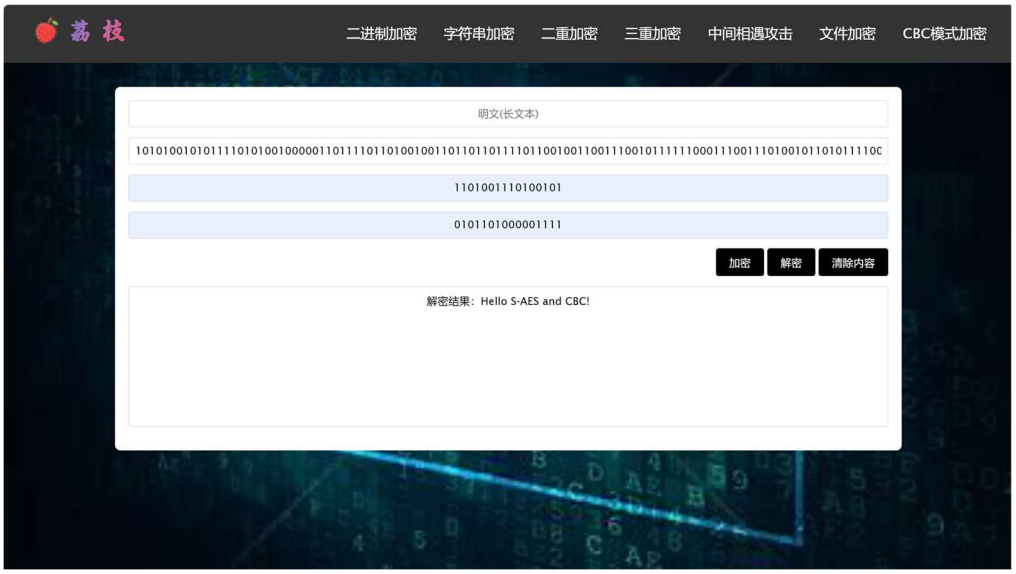
密　文　：　1010100101011110　1010010000011011　1101101001001101　1011011110110010　0110011100101111　1100011100111010　0101101011110011　0001101110010000 1110011001000100 0100111111000100

【测试结果】

1.加密过程：



2.解密过程：



可以发现，加解密结果能够自洽（加密结果为密文，解密结果为明文），CBC 模式加密算法测试成功。

3. 篡改密文发现解密获得的明文前面的 block 未改变，而最后的 block 出现错误

解密结果：Hello S-AES and CB£-