**开发手册**

**一. 概述**

S-AES 加解密系统是一个实现了简化版高级加密标准（Simplified Advanced Encryption Standard，简称S-AES）的加密和解密工具。该系统提供了多种加密模式，包括基本的S-AES加密、ASCII字符加密、双重加密、三重加密、中间相遇攻击破解以及密码块链接（CBC）模式加密。本开发手册旨在为开发者提供详细的类和方法说明，以及如何扩展和维护系统。

**二. 系统架构**

S-AES加解密系统主要由两个Java类文件组成：

SAES.java：包含S-AES算法的核心实现。

SAESGUI.java：包含图形用户界面（GUI）的实现。

**三. 类和方法说明**

3.1 SAES类

3.1.1 变量

public static int[][] S\_box：正向S盒。

public static int[][] IS\_box：逆向S盒。

public static String[] RCON`：轮常量。

public static int[] “box4”、”box2”、”box9”：列混淆和逆列混淆相关的盒子。

3.1.2 方法

public static String xor(String a, String b)：异或操作。

public static String toBinary(int num, int n)：将整数转换为n位的二进制字符串。

public static String[] NS(String[] m)：半字节代替。

public static String[] INS(String[] m)：逆半字节代替。

public static String[] SR(String[] m)：行移位（逆行移位）。

public static String[] MC(String[] m)：列混淆。

public static String[] IMC(String[] m)：逆列混淆。

public static String g(String w, int n)：g函数。

public static String encrypt(String plaintext, String key)：加密函数。

public static String decrypt(String ciphertext, String key)：解密函数。

public static String ASCII\_encrypt(String plaintext, String key)：ASCII字符加密。

public static String ASCII\_decrypt(String ciphertext, String key)：ASCII字符解密。

public static String Double\_encrypt(String plaintext, String key)：双重加密。

public static String Double\_decrypt(String ciphertext, String key)：双重解密。

public static String Attack(String plaintext, String ciphertext)：中间相遇攻击。

public static String Triple\_encrypt(String plaintext, String key)：三重加密。

public static String Triple\_decrypt(String ciphertext, String key)：三重解密。

public static String CBC\_encrypt(String plaintext, String key)：CBC模式加密。

public static String CBC\_decrypt(String ciphertext, String key, String ov)：CBC模式解密。

3.2 SAESGUI类

3.2.1 变量

private CardLayout cardLayout：卡片布局管理器。

private JPanel cardPanel：卡片面板。

3.2.2 方法

public SAESGUI()：构造函数，初始化GUI组件。

private JPanel createSAESPanel()：创建S-AES加解密面板。

private JPanel createMultiEncryptPanel()：创建多重加密面板。

private JPanel createAttackPanel()：创建中间攻击破解面板。

private JPanel createCBCPanel()：创建CBC加解密面板。

3.2.3 事件监听器

各个按钮的事件监听器，用于响应用户的加密、解密等操作。

**四. 扩展和维护**

4.1 扩展

添加新功能：可以通过添加新的类或在现有类中添加新的方法来扩展系统功能。

优化算法：可以优化S-AES算法的实现，提高加密和解密的效率。

4.2 维护

更新文档：在修改代码或添加新功能时，更新开发手册以反映这些变化。

修复bug：定期检查系统，修复发现的任何bug。