1 SM4算法

1.1 轮函数F

1.1.1 轮函数的结构

输入为$(X_0, X_1, X_2, X_3) \in (Z_2^{32})^4$，轮密钥$rk \in Z_2^{32}$，轮函数$F$为

$$F(X_0, X_1, X_2, X_3, rk) = X_0 \oplus T(X_1 \oplus X_2 \oplus X_3 \oplus rk)$$

1.1.2 合成置换T

$T : Z_2^{32} \rightarrow Z_2^{32}$是个可逆变换，由非线性变换$\tau$和线性变换$L$复合而成，即$T(\cdot) = L(\tau(\cdot))$

非线性变换τ

由4个并行的S盒构成，设输入为$A = (a_0, a_1, a_2, a_3)$，输出为$B = (b_0, b_1, b_2, b_3)$。

$$(b_0, b_1, b_2, b_3) = \tau(A) = (Sbox(a_0), Sbox(a_1), Sbox(a_2), Sbox(a_3))$$

Sbox的数据见附录。

线性变换L

非线性变换τ的输出是线性变换L的输入。设输入为$B \in Z_2^{32}$，输出为$C \in Z_2^{32}$。

$$C = L(B) = B \oplus (B <<< 2) \oplus (B <<< 10) \oplus (B <<< 18) \oplus (B <<< 24)$$

1.2 算法描述

1.2.1 加密算法

加密算法由32次迭代运算和1次反序变换R组成。

设明文输入为$(X_0, X_1, X_2, X_3) \in (Z_2^{32})^4$，密文输出为$(Y_0, Y_1, Y_2, Y_3) \in (Z_2^{32})^4$，轮密钥为$rk_i \in Z_{2^{32}}, i = 0, 1, 2, \cdots, 31$。

32次迭代运算

$$X_{i+4} = F(X_i, X_{i+1}, X_{i+2}, X_{i+3}), i = 0, 1, \cdots, 31$$

反序变换

$$(Y_0, Y_1, Y_2, Y_3) = R(X_{32}, X_{33}, X_{34}, X_{35}) = (X_{35}, X_{34}, X_{33}, Y_{32})$$

1.2.2 解密算法

解密算法与加密算法结构相同，不同仅是轮密钥的使用顺序。解密时，使用轮密钥顺序$(rk_{31}, rk_{30}, \cdots, rk_0)$。

1.2.3 密钥扩展算法

加密过程使用的轮密钥由加密密钥生成，其中加密密钥$MK = (MK_0, MK_1, MK_2, MK_3) \in (Z_2^{32})^4$。

$$(K_0, K_1, K_2, K_3) = (MK_0 \oplus FK_0, MK_1 \oplus FK_1, MK_2 \oplus FK_2, MK_3 \oplus FK_3)$$
$$rk_i = K_{i+1} = K_i \oplus T'(K_{i+1} \oplus K_{i+2} \oplus K_{i+3} \oplus CK_i), i = 0, 1, \cdots, 31$$

置换T'

T'是将合成置换T的线性变换L替换成L'

$$L'(B) = B \oplus (B <<< 13) \oplus (B <<< 23)$$

参数FK

系统参数FK的取值为:

$$FK_0 = A3B1BAC6, FK_1 = 56AA3350, FK_2 = 677D9197, FK_3 = B27022DC$$

参数CK

固定参数CK，设ck_{i+j}为CK_i的第j字节$(i = 0, 1, \cdots, 31; j = 0, 1, 2, 3)$，即$CK_i = (ck_{i,0}, ck_{i,1}, ck_{i,2}, ck_{i,3}) \in (Z_2^8)^4$，则$ck_{i+j} = (4i + j) \times 7(mod256)$

固定参数CK具体值见附录。

2 SM4查表优化

为了提升效率，可将S盒与后续的循环移位L进行合并

$$L(Sbox(a_0), Sbox(a_1), Sbox(a_2), Sbox(a_3)) =$$

$$L(Sbox(a_0) << 24) \oplus L(Sbox(a_1) << 16) \oplus L(Sbox(a_2) << 8) \oplus L(Sbox(a_3))$$

根据上式，可定义4个$8bits \to 32bits$查找表

$$T_0(a) = L(Sbox(a) << 24)$$

$$T_1(a) = L(Sbox(a) << 16)$$

$$T_2(a) = L(Sbox(a) << 8)$$

$$T_3(a) = L(Sbox(a))$$

3 实现结果

如下图所示，加密之后再解密得到的明文和原来的明文一样，SM4算法实现成功。优化后的算法明显快于优化前，优化实现成功。

```
明文为123456789abcdeffedcba9876543210
密钥为123456789abcdeffedcba9876543210
密文为681edf34d206965e86b3e94f536e4246
解密后得到明文为123456789abcdeffedcba9876543210
优化前进行1000000次加密后结果: 595298c7c6fd271f402f804c33d3f66, 耗时:3973888 微秒
优化后进行1000000次加密后结果: 595298c7c6fd271f402f804c33d3f66, 耗时:2264122 微秒
```

图 1: 实现结果

附录

行/列	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	D6	90	E9	FE	CC	E1	3D	B7	16	B6	14	C2	28	FB	2C	05
1	2B	67	9A	76	2A	BE	04	C3	AA	44	13	26	49	86	06	99
2	9C	42	50	F4	91	EF	98	7A	33	54	0B	43	ED	CF	AC	62
3	E4	B3	1C	A9	C9	08	E8	95	80	DF	94	FA	75	8F	3F	A6
4	47	07	A7	FC	F3	73	17	BA	83	59	3C	19	E6	85	4F	A8
5	68	6B	81	B2	71	64	DA	8B	F8	EB	0F	4B	70	56	9D	35
6	1E	24	0E	5E	63	58	D1	A2	25	22	7C	3B	01	21	78	87
7	D4	00	46	57	9F	D3	27	52	4C	36	02	E7	A0	C4	C8	9E
8	EA	BF	8A	D2	40	C7	38	B5	A3	F7	F2	CE	F9	61	15	A1
9	E0	AE	5D	A4	9B	34	1A	55	AD	93	32	30	F5	8C	B1	E3
A	1D	F6	E2	2E	82	66	CA	60	C0	29	23	AB	0D	53	4E	6F
B	D5	DB	37	45	DE	FD	8E	2F	03	FF	6A	72	6D	6C	5B	51
C	8D	1B	AF	92	BB	DD	BC	7F	11	D9	5C	41	1F	10	5A	D8
D	0A	C1	31	88	A5	CD	7B	BD	2D	74	D0	12	B8	E5	B4	B0
E	89	69	97	4A	0C	96	77	7E	65	B9	F1	09	C5	6E	C6	84
F	18	F0	7D	EC	3A	DC	4D	20	79	EE	5F	3E	D7	CB	39	48

表 1: SM4算法的S盒

i	CK_i	i	CK_i	i	CK_i	i	CK_i
0	A3B1BAC6	1	56AA3350	2	677D9197	3	B27022DC
4	D014F9A8	5	C7C3DF06	6	B6A899B2	7	F6FA0FAD
8	B7A3DE73	9	D1310BA6	10	98DFB5AC	11	2FFD72DB
12	D01ADFB7	13	B8E1AFED	14	6A2FADF5	15	B9A92723
16	6D80E4A3	17	BDA9CB7A	18	63030CA6	19	15188B1B
20	2C0BBD09	21	D58A0B87	22	1830F50E	23	0EC6E0E8
24	79CCE593	25	5B537E34	26	6436970C	27	70988C82
28	71F241F0	29	158F9222	30	163FAD59	31	3405F97F

表 2: SM4算法系统参数CK_i