

SMS4 密码算法

本算法是一个分组算法。该算法的分组长度为 128 比特，密钥长度为 128 比特。加密算法与密钥扩展算法都采用 32 轮非线性迭代结构。解密算法与加密算法的结构相同，只是轮密钥的使用顺序相反，解密轮密钥是加密轮密钥的逆序。

1. 术语说明

1.1 字与字节

用 Z_2^e 表示 e -比特的向量集， Z_2^{32} 中的元素称为字， Z_2^8 中的元素称为字节。

1.2 S 盒

S 盒为固定的 8 比特输入 8 比特输出的置换，记为 $Sbox(.)$ 。

1.3 基本运算

在本算法中采用了以下基本运算：

\oplus 32 比特异或

$\lll i$ 32 比特循环左移 i 位

1.4 密钥及密钥参量

加密密钥长度为 128 比特，表示为 $MK=(MK_0, MK_1, MK_2, MK_3)$ ，其中 $MK_i(i=0,1,2,3)$ 为字。

轮密钥表示为 $(rk_0, rk_1, \dots, rk_{31})$ ，其中 $rk_i(i=0, \dots, 31)$ 为字。轮密钥由加密密钥生成。

$FK=(FK_0, FK_1, FK_2, FK_3)$ 为系统参数， $CK=(CK_0, CK_1, \dots, CK_{31})$ 为固定参数，用于密钥扩展算法，其中 $FK_i(i=0, \dots, 3)$ 、 $CK_i(i=0, \dots, 31)$ 为字。

2. 轮函数 F

本算法采用非线性迭代结构，以字为单位进行加密运算，称一次迭代运算为一轮变换。

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

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

2.1 合成置换 T

$T: Z_2^{32} \rightarrow Z_2^{32}$, 是一个可逆变换, 由非线性变换 τ 和线性变换 L 复合而成, 即 $T(.)=L(\tau(.))$ 。

2.1.1 非线性变换 τ

τ 由 4 个并行的 S 盒构成。

设输入为 $A = (a_0, a_1, a_2, a_3) \in (Z_2^8)^4$, 输出为 $B = (b_0, b_1, b_2, b_3) \in (Z_2^8)^4$, 则

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

2.2.2 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 $C \in Z_2^{32}$, 则

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

2.2 S 盒

S 盒中数据均采用 16 进制表示。

	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

例：输入‘ef’，则经 S 盒后的值为表中第 e 行和第 f 列的值， $Sbox('ef') = '84'$ 。

3. 加/解密算法

定义反序变换 R 为：

$$R(A_0, A_1, A_2, A_3) = (A_3, A_2, A_1, A_0), \quad A_i \in Z_2^{32}, \quad i = 0, 1, 2, 3。$$

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 $rk_i \in Z_2^{32}, i = 0, 1, 2, \dots, 31$ 。则本算法的加密变换为：

$$X_{i+4} = F(X_i, X_{i+1}, X_{i+2}, X_{i+3}, rk_i) = X_i \oplus T(X_{i+1} \oplus X_{i+2} \oplus X_{i+3} \oplus rk_i), \quad i = 0, 1, \dots, 31。$$

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

本算法的解密变换与加密变换结构相同，不同的仅是轮密钥的使用顺序。

加密时轮密钥的使用顺序为：($rk_0, rk_1, \dots, rk_{31}$)

解密时轮密钥的使用顺序为：($rk_{31}, rk_{30}, \dots, rk_0$)

4. 密钥扩展算法

本算法中加密算法的轮密钥由加密密钥通过密钥扩展算法生成。

加密密钥 $MK = (MK_0, MK_1, MK_2, MK_3)$ ， $MK_i \in Z_2^{32}, i = 0, 1, 2, 3$ ；

令 $K_i \in Z_2^{32}, i = 0, 1, \dots, 35$ ，轮密钥为 $rk_i \in Z_2^{32}, i = 0, 1, \dots, 31$ ，则轮密钥生成方法为：

首先， $(K_0, K_1, K_2, K_3) = (MK_0 \oplus FK_0, MK_1 \oplus FK_1, MK_2 \oplus FK_2, MK_3 \oplus FK_3)$

然后，对 $i = 0, 1, 2, \dots, 31$ ：

$$rk_i = K_{i+4} = K_i \oplus T'(K_{i+1} \oplus K_{i+2} \oplus K_{i+3} \oplus CK_i)$$

说明：

(1) T' 变换与加密算法轮函数中的 T 基本相同，只将其中的线性变换 L 修改为以下 L' ：

$$L'(B) = B \oplus (B \lll 13) \oplus (B \lll 23)；$$

(2) 系统参数 FK 的取值，采用 16 进制表示为：

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

(3) 固定参数 CK 的取值方法为：

设 $ck_{i,j}$ 为 CK_i 的第 j 字节 ($i = 0, 1, \dots, 31; j = 0, 1, 2, 3$)，即 $CK_i = (ck_{i,0}, ck_{i,1}, ck_{i,2}, ck_{i,3}) \in (Z_2^8)^4$ ，

则 $ck_{i,j} = (4i+j) \times 7 \pmod{256}$ 。32 个固定参数 CK_i ，其 16 进制表示为：

00070e15, 1c232a31, 383f464d, 545b6269,
70777e85, 8c939aa1, a8afb6bd, c4cbd2d9,
e0e7eef5, fc030a11, 181f262d, 343b4249,
50575e65, 6c737a81, 888f969d, a4abb2b9,
c0c7ced5, dce3eaf1, f8ff060d, 141b2229,
30373e45, 4c535a61, 686f767d, 848b9299,
a0a7aeb5, bcc3cad1, d8dfe6ed, f4fb0209,
10171e25, 2c333a41, 484f565d, 646b7279

5. 加密实例

以下为本算法 ECB 工作方式的运算实例，用以验证密码算法实现的正确性。其中，数据采用 16 进制表示。

实例一：对一组明文用密钥加密一次

明文：01 23 45 67 89 ab cd ef fe dc ba 98 76 54 32 10

加密密钥：01 23 45 67 89 ab cd ef fe dc ba 98 76 54 32 10

轮密钥与每轮输出状态：

rk[0] = f12186f9	X[0] = 27fad345
rk[1] = 41662b61	X[1] = a18b4cb2
rk[2] = 5a6ab19a	X[2] = 11c1e22a
rk[3] = 7ba92077	X[3] = cc13e2ee
rk[4] = 367360f4	X[4] = f87c5bd5
rk[5] = 776a0c61	X[5] = 33220757
rk[6] = b6bb89b3	X[6] = 77f4c297
rk[7] = 24763151	X[7] = 7a96f2eb
rk[8] = a520307c	X[8] = 27dac07f
rk[9] = b7584dbd	X[9] = 42dd0f19
rk[10] = c30753ed	X[10] = b8a5da02
rk[11] = 7ee55b57	X[11] = 907127fa
rk[12] = 6988608c	X[12] = 8b952b83
rk[13] = 30d895b7	X[13] = d42b7c59
rk[14] = 44ba14af	X[14] = 2ffc5831
rk[15] = 104495a1	X[15] = f69e6888
rk[16] = d120b428	X[16] = af2432c4
rk[17] = 73b55fa3	X[17] = ed1ec85e
rk[18] = cc874966	X[18] = 55a3ba22
rk[19] = 92244439	X[19] = 124b18aa
rk[20] = e89e641f	X[20] = 6ae7725f
rk[21] = 98ca015a	X[21] = f4cba1f9
rk[22] = c7159060	X[22] = 1dcdfa10
rk[23] = 99e1fd2e	X[23] = 2ff60603
rk[24] = b79bd80c	X[24] = eff24fdc
rk[25] = 1d2115b0	X[25] = 6fe46b75
rk[26] = 0e228aeb	X[26] = 893450ad
rk[27] = f1780c81	X[27] = 7b938f4c
rk[28] = 428d3654	X[28] = 536e4246
rk[29] = 62293496	X[29] = 86b3e94f
rk[30] = 01cf72e5	X[30] = d206965e
rk[31] = 9124a012	X[31] = 681edf34

密文：68 1e df 34 d2 06 96 5e 86 b3 e9 4f 53 6e 42 46

实例二：利用相同加密密钥对一组明文反复加密 1000000 次

明文：01 23 45 67 89 ab cd ef fe dc ba 98 76 54 32 10

加密密钥：01 23 45 67 89 ab cd ef fe dc ba 98 76 54 32 10

密文：59 52 98 c7 c6 fd 27 1f 04 02 f8 04 c3 3d 3f 66