DOI:10.18686/ahe.v7i24.10078

Research on the Implementation of SM4 National Security Algorithm

Lingyun Liu

Physics and Electronic Information Engineering, Jining Normal University , Ulanqab, Inner Mongolia 012000

Abstract: SM4 packet cipher algorithm is a symmetric packet cipher designed by China, which provides secure and complete data encryption for many information systems. There is not much work on the software optimization of SM4 algorithm implementation. In this paper, we propose a general software optimization method for implementing symmetric grouping cryptographic algorithms, which can be used for fast implementation of all symmetric encryption algorithms. The method is generalized to the fast software implementation of all symmetric encryption algorithms.

Keywords: SM4; Algorithm; Encryption; Key

Fund Project:

JiNing Normal University Master Lecturer Project Source Jining Normal University Natural Science Project Project number: jsky2021098

1. Introduction

SM4 packet cipher algorithm is a symmetric packet cipher designed by China, which provides secure and complete data encryption for many information systems.

It provides a secure and complete data encryption solution for many information systems. The efficient software implementation of the SM4 algorithm has provided strong support for the replacement of the cryptographic algorithm from international standards to national standards for security products in China.

It provides a strong support for the replacement of international standards with national standards, and provides a strong support for the wide use of SM4 algorithm in government offices, public security, banks, taxation, electric power and other information systems with high requirements for autonomous control.

It provides a reliable guarantee that SM4 algorithms are widely used in government offices, public security, banking, taxation, electric power and other information systems with high requirements for autonomous control. There is not much work on the software optimization of SM4 algorithm.

However, due to the relatively large size of the substitution table, when the CPU does the table lookup operation, the data in the table is frequently swapped between memory and cache, resulting in the table lookup delay.

In addition, the table lookup method is not resistant to cache-timing side channel attacks, thus limiting the performance and security of SM4 software implementation to some extent.

2. Introduction to the SM4 Algorithm

The SM4 algorithm is designed by Shu-Wang Lu, Da-Wei Li, Chao Zhang and others and drafted by the related standard, its group length and key length are 128 bits, and the algorithm adopts the non-equilibrium Feistel structure. The SM4 is decrypted by 32 rounds of nonlinear iteration with an inverse order transformation, which is convenient for decryption, and the decryption round key only needs to be the inverse order of the encryption round key, while the decryption algorithm remains the same as the encryption algorithm.

3. Algorithm Structure

SM4 is a block cipher algorithm. Its block length and cipher key length are both of 128 bits. SM4 adopts an unbalanced Feistel structure and iterates its round functions for 32 times in both encryption and key expansion algorithm. The structure of decryption is the same as the encryption. But the decryption round keys are in the reverse order of the encryption round keys.

4. Key and Key Parameters

The 128-bit cipher key is represented as $MK = (MK_0, MK_1, MK_2, MK_3)$, where $MK_i = (i = 0,1,2,3)$ are 32-bit words.

The round keys are represented as $(rk_0, rk_1 \ldots, rk_{31})$, where rk_i $(i = 0, \ldots, 31)$ are 32-bit words. The round keys are generated from the cipher key via key expansion algorithm.

The system parameter is $FK = (FK_0, FK_1, FK_2, FK_3, FK_4)$, and the fixed parameter is $CK = (CK_0, CK_1 \ldots, CK_{31})$, where the FK_i $(i = 0,1,2,3)$ and CK_i $(i = 0, \ldots, 31)$ are 32-bit words used in the key expansion algorithm.

5. Round Function *F*

5.1 Round Function Structure

Suppose the input to round function is $(X_0, X_1, X_2, X_3) \in (Z_2^{32})^4$ and the round key is $rk \in Z_2^{32}$, then F can be represented as:

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

5.2 Permutation $\boldsymbol{T}$

T: $Z_2^{32} \rightarrow Z_2^{32}$ is an invertible transformation, composed of a nonlinear transformation τ and a linear transformation L. That is, $T(\cdot) = L(\tau(\cdot))$.

(a)Nonlinear transformation

τ is composed of 4 S-boxes in parallel. Suppose $A = (a_0, a_1, a_2, a_3) \in (Z_2^8)^4$ is input to τ, and $B = (b_0, b_1, b_2, b_3) \in (Z_2^8)^4$ is the corresponding output, then

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

		y															
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
x	**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

The S-box is as follows:

Note: substitution values for the byte xy (in hexadecimal format), e.g. when the input is 'EF', then the output is the value in row

E and column F, i.e. $Sbox(EF) = 84$.

(b) Linear transformation $\boldsymbol{L}$

The output from the nonlinear transformation τ is the input to the linear transformation L. Suppose the input to L is $B \in Z43$, and the corresponding output is

$C \in Z_2^{32}$, then

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

6. Algorithm Description

6.1 Decryption

The structure of the decryption transformation is the same as the encryption transformation. The only difference is the order of the round keys. In decryption, the round keys are used in the order of ($rk_{31}, rk_{30}, \ldots, rk_0$).

6.2 Key Expansion

The round keys in this algorithm are generated from the cipher key via the key expansion algorithm.

Suppose the cipher key is $MK = (MK_0, MK_1, MK_2, MK_3 \in (Z_2^{32})^4$ then the round keys are generated as follows:

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

$rk_i = K_{i+4} = K_i \oplus T'(K_{i+1} \oplus K_{i+2} \oplus K_{i+3} \oplus CK_i)$, $i = 0,1, \ldots, 31$,

where

(a) T' replaces the linear transformation L in permutation T by L': $L'B = B \oplus (B <<< 13) \oplus (B <<< 23)$.

(b) The system parameter FK is:

$$FK_0 = (A3B1BAC6), \quad FK_1 = (56AA3350),$$

$$FK_2 = (677D9197), \quad FK_3 = (B27022DC).$$

(c) The fixed parameter CK is used in the key expansion algorithm. Suppose $CK_{i,j}$ is the j -th byte of CK_i ($i = 0,1, \ldots, 31$, $j = 0,1,2,3$), i.e. $CK_i = (ck_{i,0}, ck_{i,1}, ck_{i,2}, ck_{i,3}) \in (Z_2^8)^4$, then $CK_{i,j} = (4i + j) \times 7 (mod\ 256)$. To be specific, the values of the fixed parameters CK_i ($i = 0,1, \ldots, 31$) are:

00070E15,	1C232A31,	383F464D,	545B6269,
70777E85,	8C939AA,	A8AFB6D	C4CBD2D9,
E0E7EEF5,	FC030A11,	181F262D,	343B4249,
50575E65,	6C737A81,	888F969D,	A4ABB2B9,
C0C7CED5,	DCE3EA1,	F8FF060D,	141B2229,
30373E45,	4C535A61,	686F767D,	848B9299,
A0A7AEB5,	BCC3CA1,	D8DFE6D,	F4FB0209,
10171E25,	2C333A41,	484F565D,	646B7279.

7. Summary

In this paper, we propose a software optimization method for the SM4 algorithm, which is resistant to external attacks and thus improves the security of the algorithm implementation. In addition, the optimization method is scalable and can be extended from SM4 algorithm to all current symmetric encryption algorithms.

References:

[1] Chao, P.E.I. A Method of masking SM4 and analysis against DPA attacks. J. Cryptol. Res. 2016, 3, 79–90.

[2] Di, W.; Wu, L.; Zhang, X. Key-leakage hardware Trojan with super concealment based on the fault injection for block cipher of SM4. Electron. Lett. 2018, 54, 810–812.