UDC 004.056.55

DEVELOPMENT OF ENCRYPTION ALGORITHM BASED ON THE ALGORITHMS OF RUBIK'S CUBE

EUGENE IVANENKO, YURI PASTUKHOV
Polotsk State University, Belarus

The article deals with the design of encryption algorithm based on Rubik's cube algorithms. A generalized algorithm for solving the problem is presented.

Data encryption and decryption algorithms are widely used in computer technology to hide confidential and commercial information from malicious use by third parties. The main principle in them is the condition that the transmitter and the receiver know in advance the encryption algorithm, as well as the key to the message, without which the information is just a set of characters that do not make sense.

The Rubik's cube-based encryption algorithm is a permutation cipher. The method of permutation is that the characters of the encrypted text are rearranged according to certain rules within the encrypted block of characters, that is, transformations lead to a change only in the order of the characters of the original message.

In 1991, V. M. Kuzmich proposed a permutation scheme based on the Rubik's cube. According to this scheme, the open text is written to the cells of the cube faces by rows. After the implementation of specified number of specified turns of the layers of the cube to read the ciphertext is carried out by columns. The complexity of decryption in this case is determined by the number of cells on the cube faces and the complexity of the layer rotations. A permutation based on the Rubik's cube is called a volume (multidimensional) permutation. [1]

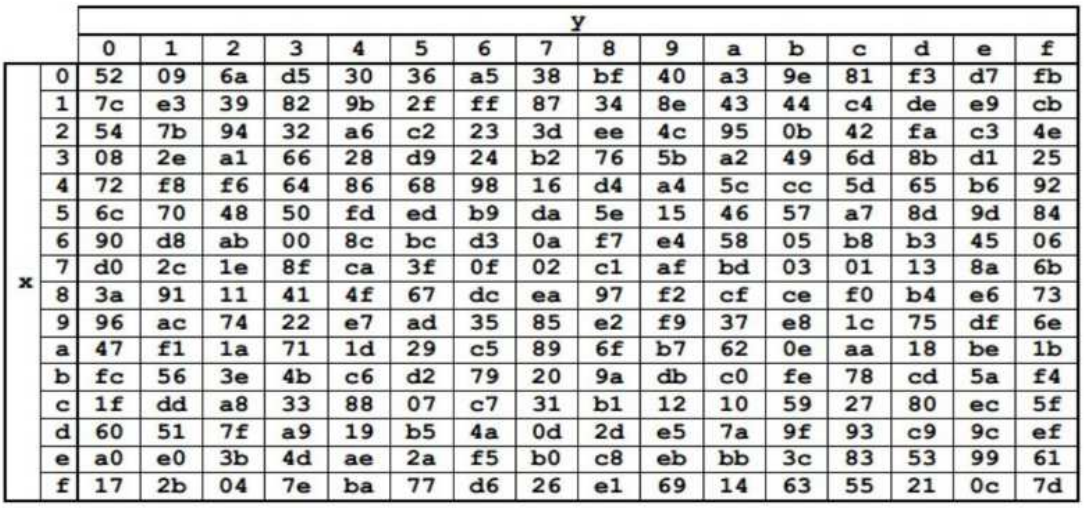
This encryption algorithm based on the Rubik's cube algorithm has been changed to work not with the characters of the encrypted text, but with an array of bytes, which is obtained by converting the encrypted text. This algorithm is a symmetric encryption algorithm.

The encryption algorithm based on the Rubik's cube algorithm works with encryption keys of different lengths. The algorithm can use the keys 16 bytes (128 bits), 32 bytes (256 bits), 64 bytes (512 bits), 128 bytes (1024 bits) and 256 bytes (2048 bits). The key length determines the number of rounds that will be used to encrypt or decrypt the characters of the original message.

A generalized algorithm for solving the problem.

The encryption algorithm includes the following steps:

1. An encryption key of the required length is generated or set for the algorithm.
2. The characters of the original message are converted to an array of bytes.
3. From the obtained key is generated by an array of keys.
4. The data array is divided into blocks of 6 bytes (48 bits). If there is not enough data to form the whole block, the block is supplemented with "0".
5. Each byte in the data block is replaced with the corresponding byte in the constant table shown in figure 1.



		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	52	09	6a	d5	30	36	a5	38	bf	40	a3	9e	81	f3	d7	fb
	1	7c	e3	39	82	9b	2f	ff	87	34	8e	43	44	c4	de	e9	cb
	2	54	7b	94	32	a6	c2	23	3d	ee	4c	95	0b	42	fa	c3	4e
	3	08	2e	a1	66	28	d9	24	b2	76	5b	a2	49	6d	8b	d1	25
	4	72	f8	f6	64	86	68	98	16	d4	a4	5c	cc	5d	65	b6	92
	5	6c	70	48	50	fd	ed	b9	da	5e	15	46	57	a7	8d	9d	84
	6	90	d8	ab	00	8c	bc	d3	0a	f7	e4	58	05	b8	b3	45	06
	7	d0	2c	1e	8f	ca	3f	0f	02	c1	af	bd	03	01	13	8a	6b
	8	3a	91	11	41	4f	67	dc	ea	97	f2	cf	ce	f0	b4	e6	73
	9	96	ac	74	22	e7	ad	35	85	e2	f9	37	e8	1c	75	df	6e
	a	47	f1	1a	71	1d	29	c5	89	6f	b7	62	0e	aa	18	be	1b
	b	fc	56	3e	4b	c6	d2	79	20	9a	db	c0	fe	78	cd	5a	f4
	c	1f	dd	a8	33	88	07	c7	31	b1	12	10	59	27	80	ec	5f
	d	60	51	7f	a9	19	b5	4a	0d	2d	e5	7a	9f	93	c9	9c	ef
	e	a0	e0	3b	4d	ae	2a	f5	b0	c8	eb	bb	3c	83	53	99	61
	f	17	2b	04	7e	ba	77	d6	26	e1	69	14	63	55	21	0c	7d

Figure 1. – Constant replacement table

6. The data block is converted to a 6 by 8 bit matrix.
7. The resulting matrix is mixed according to the Rubik's cube algorithm.
8. The matrix obtained after mixing is modulo two with a round key.
9. Each block is shifted 1 byte to the left.

Round keys are generated from the key of the cipher K by means of the key expansion procedure, as a result of which an array of round keys is formed, from which the necessary round key is then directly selected.

Each round key has a length of 128 bits (or 4 four-byte words w_i, w_{i+1}, wi_{+2}, wi_{+3}, and the length in bits of all round keys is 128 bits). The first four words w_i, w_{i+1}, wi_{+2}, wi_{+3} in the key array are filled with the cipher key,4 words for the round key are selected from the remaining 44 words. The choice of words is simple: the first four words (they match the cipher key) are the key with number 0, the next four words are w_4, w_5, w_6, w_7 – round key for the first full round, etc.

New words w_{i+4}, w_{i+5}, w_{i+6}, w_{i+7} next round key are determined from words w_i, w_{i+1}, wi_{+2}, wi_{+3} previous key based on equations:

– w_{i+5} $w_{i+4} \oplus w_{i+1}$;

– w_{i+6} $w_{i+5} \oplus w_{i+2}$;

– w_{i+7} $w_{i+6} \oplus w_{i+3}$.

The first word w_{i+4} in each round the key is modified differently:

– $w_{i+4} = w_i \oplus g(w_{i+3})$.

Here, the action of function g is reduced to the sequential execution of three steps, displaying word for word:

1 Cyclic shift of a four-byte word to the left by one byte.

2 Replace each byte of the word obtained in step 10 according to the constant substitution table used in encryption.

3 Mod summation of 2 bytes received in step 2, with a round constant $R_{con}[i]$ = (RC[i],0,0,0), unclassified and unique for each round key. The right-most three bytes of this constant is zero, and a non-zero left bytes varies according to the known law of recursion: RC[1] =1 , RC [i] = 2 * RC[i-1] , i1,2,...11 .

The purpose of summation with round constants is to break any symmetry that can occur at different stages of key expansion and lead to the appearance of weak keys, as in the DES algorithm.

The operation of the key expansion algorithm is shown in figure 2.

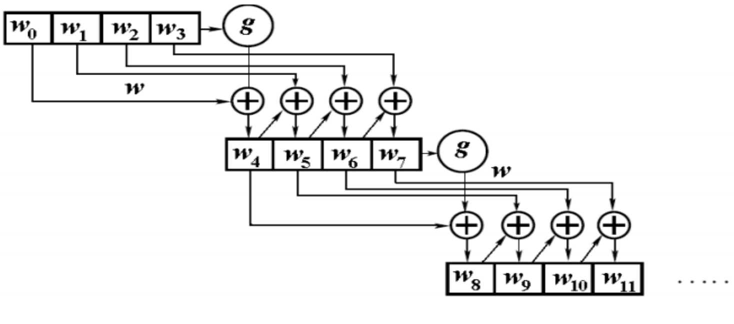


Figure 2. – Key expansion Algorithm

In the course of this study, an encryption algorithm based on the Rubik's cube algorithm was designed. At the same time, the developed algorithm leaves the possibility for further development and introduction of additional security features.

REFERENCES

1. Материал из StudFiles — файловый архив студентов. Шифры перестановки [Электронный ресурс]. – Режим доступа: https://studfiles.net/preview/5470123/page:8/. – Дата доступа: 20.09.2018.
2. Птицын, Н. Приложение теории детерминированного хаоса в криптографии / Н. Птицын. – М. : МГТУ им. Н.Э. Баумана, 2002. – 80 с.
3. Гатчин, Ю.А. Основы криптографических алгоритмов : учеб. пособие / Ю.А.Гатчин, А.Г. Коробейников. – СПб. : ГИТМО (ТУ), 2002. – 29 с.
4. Жданов, О.Н., Актуальные проблемы безопасности информационных технологий / О.Н. Жданов. – Красноярск : Сиб. гос. аэрокосмич. ун-т, 2009. – 144 с.