This paper analyzes ways to improve the cryptographic strength of the symmetric block cipher RC5. The task to enhance the stability of the classic RC5 cipher is explained by the fact that it is part of various open cryptographic libraries and is frequently used in practice. Several methods have been considered, applying which theoretically contributes to improving the stability of cryptographic transformations. It is found that unlike other alternatives (increasing the number of rounds, the length of the key, and the encryption block), the use of nonlinear shift functions does not increase the computational complexity of the RC5 algorithm. The study result has helped build an analytical model that was implemented in the form of the MATLAB (USA) software application. The software interface provides the ability to manually change the encryption parameters of the RC5 crypto algorithm. The resulting upgrade of the RC5 crypto algorithm has been tested on different sets of input data during encryption and decryption. The resulting modification also does not lead to an increase in the calculation time but makes it possible to improve the resistance to hacking the encrypted data by several orders of magnitude (210), provided that differential analysis methods are used and the number of rounds is 14. For one of the nonlinear functions used, resistance to the differential cryptoanalysis used increased by 212 times already in the eleventh round of encryption. The reliability of the improved cryptosystem has been confirmed by the absence of statistical correlation between the blocks of incoming messages and output blocks, the absence of collisions at which it is possible to obtain the same sequences of bits at the output with different messages at the input. The resulting algorithm could be applied in computer systems with low computing performance

Keywords: nonlinear function, symmetric cryptosystem, shift function, RC5, block cipher, cryptoanalysis

UDC 003.26.09
DOI: 10.15587/1729-4061.2021.240344

DEVISING A METHOD FOR IMPROVING CRYPTO RESISTANCE OF THE SYMMETRIC BLOCK CRYPTOSYSTEM RC5 USING NONLINEAR SHIFT FUNCTIONS

Andrii Sahun
Corresponding author
PhD, Associated Professor*
E-mail: avd29@ukr.net
Vladyslav Khaidurov
PhD, Senior Researcher
Department of Monitoring and Optimization of Thermophysical Processes
Institute of Engineering Thermophysics of the Institute of Engineering Thermophysics of NAS of Ukraine
Mariyi Kapnist (Zheliabova) str., 2a, Kyiv, Ukraine, 03057
Valeriy Lakhno
Doctor of Technical Sciences, Professor*
Ivan Opirskyy
Doctor of Technical Sciences, Professor
Department of Information Security
Lviv Polytechnic National University
S. Bandery str., 12, Lviv, Ukraine, 79013
Vitalii Chubaievskyi
PhD, Associate Professor**
Deputy Chief of Department, Police Colonel
Department of Information and Analytical Support of the National Police of Ukraine
Akademika Bohomoltsia str., 10, Kyiv, Ukraine, 01601
Olena Kryvoruchko
Doctor of Technical Sciences, Professor**
Alona Desiatko
PhD, Associate Professor**
*Department of Computer Systems and Networks
National University of Life and Environmental Sciences of Ukraine
Heroiv Oborony str., 15, Kyiv, Ukraine, 03041
**Department of Software Engineering and Cyber Security
Kyiv National University of Trade and Economics
Kyoto str., 19, Kyiv, Ukraine, 02156

Received date 08.07.2021
Accepted date 01.09.2021
Published date 29.10.2021

***How to Cite**: Sahun, A., Khaidurov, V., Lakhno, V., Opirskyy, I., Chubaievskyi, V., Kryvoruchko, O., Desiatko, A. (2021). Devising a method for improving crypto resistance of the symmetric block cryptosystem rc5 using nonlinear shift functions. Eastern-European Journal of Enterprise Technologies, 5 (9 (113)), 17–29. doi: https://doi.org/10.15587/1729-4061.2021.240344*

1. Introduction

It is possible to encrypt data when they are transmitted through computer networks at any of the seven existing levels of the OSI open systems interaction model [1]. Current trends in the field of information protection provide for the widespread introduction of data encryption to ensure, among other things, network security. To this end, a variety

of symmetric crypto algorithms are used in practice. The most reliable way to protect the confidentiality of information in computer networks is channel encryption. The feature of channel encryption is that all data that pass through a specific communication channel (even the open text of the message) is encrypted. Information about the routing of this message and information about the routing protocol are also encrypted. Often, the symmetric block algorithm RC5 is used for encryption. The application of any of the symmetric ciphers requires that the switch must decrypt the data stream in order to process it, and then encrypt it again for transmission to another network switch. However, it is difficult to abandon channel encryption, especially symmetric and block type: channel encryption is a very effective means of protecting information in computer networks [2].

An important advantage of the block algorithm RC5 is the fact that there is no definite dependence of the cipher on the signs of the open message, and there is also no positional dependence of the cipher (there is no problem of error multiplication). This property is important for ensuring the integrity of information, which is especially in demand within the framework of the Internet of Things and smart home technologies. An important property of a given algorithm (as well as any block) is that each bit of the block of the encrypted message is a function of all (almost all) bits of the corresponding block of clear text. Thus, there are no two blocks of plaintext that can be represented by an identical block of ciphertext.

The algorithms of the RC5 family provide for the partitioning of a message into a certain number of parts (or characters) of fixed size, each of which is encrypted separately. This greatly simplifies the task of encryption.

If we analyze these advantages of RC5, especially taking into consideration its practical possibilities, it can be argued that the block symmetric encryption algorithm RC5 has prospects in various applications. It contains the possibility of enhancing the quality of encryption and reducing the computing load on the computing mechanisms of computer systems [3].

Software suites employing RC5 include the OpenSSL package [4], built into many of RSA Data Security Inc.'s core products, such as BSAFE, JSAFE, S/MAIL, and OpenVPN [5]. Therefore, it is a relevant task to improve the cryptographic strength of RC5 in order to increase its cryptographic stability and performance speed on "weak" computing platforms. This is necessary for its use in the promising systems of the "smart home", the Internet of Things, etc. [6].

2. Literature review and problem statement

In work [3], the authors of the RC5 crypto algorithm note that the proposed algorithm could easily be implemented in hardware. At the same time, studies [3, 7] also proposed the possibility of strengthening its cryptographic power without increasing computational complexity, which is very important in the applied application of cryptographic transformation. Modern trends in cryptographic protection (for example, the hardware implementation of the AES encryption algorithm in the computing cores of the microprocessor [8]) indicate the prospects of such implementations. Many cryptographic computer protection tools are currently implemented in the form of specialized hardware units or devices. The creators of the RC5 crypto algorithm implied that it could be easily implemented by both hardware and software methods [3].

It is because of the need to encrypt the entire volume of traffic on the Internet of Thing networks to protect network traffic that other methods of protection are used, for example, firewalls [6]. Such networks are usually built on the basis of low-power intelligent switches and controllers such as Arduino or Raspberry Pi, or their analogs.

The issue of increasing the cryptographic resistance of symmetric block systems is considered in many publications. Some of them address the classical crypto algorithm RC5. Moreover, it is a convenient basis for modification.

In terms of the number of users, RC5 is not inferior to such well-known algorithms as IDEA and Blowfish. For example, the popular PGP encryption system uses the IDEA code [9]. However, the use of the IDEA cipher is significantly limited by patent rules. The Blowfish algorithm described in [10] is analogous to the RC5 in question in the use of the Feistel network but cannot be considered as an alternative due to higher hardware requirements. The RC5 cipher is characterized by a variable number of rounds, block length, key length [3]. This expands the possibilities of use and simplifies the transition to a stronger version of the algorithm. One way to make this algorithm "stronger" is to improve the underlying RC5 algorithm by choosing a nonlinear shift in rounds using different functions. This approach to improving cryptographic resistance is proposed in several works [3, 7, 8]. Although the classical (basic) algorithm is built on a single shift function, it can be modified by using several successively nonlinear shift functions [11].

To assess the cryptographic strength of the resulting modernization, various methods of attacks (cryptoanalysis) are used. It is known that the classical RC5 algorithm requires approximately 2^{45} open texts for a successful attack [12]. It is also known that when using more than 14 rounds of encryption, instead of classic 12, it is almost impossible to crack an encrypted RC5 text using differential cryptoanalysis [11]. Thus, the most realistic method for cracking the RC5 algorithm with a large number of rounds (not counting options with a small number of rounds and with a short key) is a complete sorting of possible options for the encryption key. This is the basis for the assertion that the RC5 algorithm is resistant to differential cryptoanalysis.

Therefore, the question of the nature of the round shift in RC5 to increase its cryptographic resistance is open. Devising an effective and fast algorithm for symmetric block encryption is possible only if it is reliable. For RC5, there are studies that show a special class of keys that allow RC5 to be opened during linear cryptoanalysis. However, this issue is typical for linear cryptoanalysis of any block cipher [13]. In addition, there are methods that accelerate the search for vulnerabilities in the encryption of this algorithm by the differential cryptoanalysis methods [11]. The authors of work [11] show a radical improvement in the results of cryptoanalysis due to a new approach with partial derivatives. At the same time, 2^{44} selected open texts are required for successful cryptoanalysis. Although the RC5 version for 64-bit words in that work also proved not to be resistant to cryptoanalysis, the attack did not use round encryption with nonlinear functions. Paper [13] describes a technique for the linear cryptoanalysis of the symmetric DES cipher with 8 rounds of encryption on 221 open messages and with 16 rounds and 2^{47} open messages. However, the DES cipher

has differences from RC5, which makes it impossible to directly use the attack proposed in work [13]. In [15], some possible extensions of attacks and some modifications of RC5 are considered. In particular, when it works with words in 32 bits with 12 rounds and a 128-bit user key. The cited paper also notes that RC5 has many weak keys to differential attacks. This weakness depends on the structure of the cipher, not on the distribution of the keys. However, the cryptographic attack on the RC5 cipher of the nonlinear function is not described in [15]. All this gives grounds to assert that it is expedient to conduct a study aimed at devising a method for increasing the cryptographic resistance of the symmetric block cryptosystem RC5 using nonlinear shift functions.

3. The aim and objectives of the study

The purpose of this study is to devise a method for improving the cryptographic strength of a symmetric block cryptosystem to work on "weak" platforms based on the RC5 algorithm using nonlinear shift functions.

To accomplish the aim, the following tasks have been set:

– to consider the implementation of the most practical encryption algorithm of the RC5 family, and select nonlinear functions for implementing a bit round shift;

– to define test datasets for encryption and decryption, determining the quantitative and qualitative characteristics of the obtained options for upgrading the RC5 algorithm;

– to carry out software implementation and modeling of the modified encryption algorithm of the RC5 family with different sets of standard parameters for determining the quantitative and qualitative parameters of the modified crypto algorithm;

– to choose the method of cryptoanalysis and determine the cryptographic resistance of the obtained variants of the RC5 cryptosystem;

– to analyze and generalize, to devise recommendations, and define problematic issues that arose during the construction of the method concerning its application, as well as promising areas for further research.

4. The study materials and methods

The following research methods have been used: methods of linear and differential cryptoanalysis to assess the cryptographic strength of the obtained modification of the RC5 cipher. Elements of number theory and modular arithmetic for the study and modification of the operation of shift functions in the encryption rounds of the classical RC5 algorithm. To implement and study the classical and modified RC5 algorithm, discrete data structures and elements of discrete mathematics methods were applied. The practical implementation of the proposed analytical model of the modified RC5 algorithm is based on the paradigm of functional programming in the built-in MATLAB language. To carry out differential cryptoanalysis, a mathematical apparatus based on statistical methods for estimating the probability of meeting the same sub-block round sequences for each round (the sequence size is 8 bits) was employed. In addition, to visualize the results of quantitative and qualitative parameters in the operation of the modified algorithm, we applied MATLAB charting tools with a logarithmic scale.

5. Results of devising a method for increasing the cryptographic resistance of the symmetric block cryptosystem RC5 using nonlinear shift functions

5. 1. Analysis of the RC5 encryption algorithm and selection of nonlinear functions for the implementation of a bit round shift

The operational principle of the crypto algorithm RC5 has been described in detail. Some of the main parameters of the RC5 algorithm are variable parameters [3, 11]. It is known that in the algorithm, in addition to the secret key, there are some others, namely:

– the word size w (in bits). The RC5 algorithm encrypts blocks of two words (hereinafter referred to as A and B, respectively). Valid values of w are the natural numbers 16, 32, or 64. However, the recommended word size is 32 bits;

– the number of rounds of the R-algorithm. As this parameter, one can use any integer in the range from 0 to 255 inclusive;

– the size of the secret key b (in bytes) may vary. This is any integer between 0 and 255 inclusive.

When encrypting two blocks A and B in binary representation, the classic RC5 algorithm is executed in such a way that before the first round the extended key S is superimposed

$$A=\left(A+S_0\right) \bmod 2^w,\ \ B=\left(B+S_1\right) \bmod 2^w. \quad (1)$$

The following actions are performed in each round:

$$A=\left(\left(A\oplus B\right)<<B\right)+S_{2i},$$

$$B=\left(\left(B\oplus A\right)<<A\right)+S_{2i+1}, \quad (2)$$

where A, B are the message blocks in binary encoding (the classic size of each block is 32, 64, or 128 bits);

S_i – S-key extension table;

w – half the length of the text block (half the number of bits in the block – 32, 64, or 128 bits).

A diagram of one round of operation of the block cipher of the RC5 family with the highlighted shift function is shown in Fig. 1.

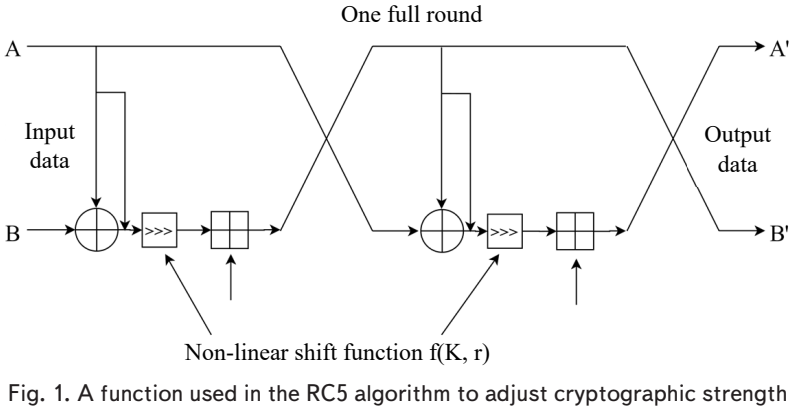

Fig. 1. A function used in the RC5 algorithm to adjust cryptographic strength in the encryption round structure

Fig. 1 shows the shift function by icons «>>» and «<<». In the RC5 algorithm, they can be used to increase its cryptographic resistance.

Recurrent dependences (1) and (2) underlie the classical RC5 cryptosystem. These expressions involve the use of a Feistel network, in which, after modulo addition operations and XOR operations, the positions of blocks A and B are reversed. This block exchange principle underlies the RC5 algorithm operation. These formulas involve the use of a Feistel network, in which, after modulo addition operations and XOR operations, the positions of blocks A and B change places. This principle of block exchange is the basis of RC5 operation.

Decryption is performed in reverse order, similar to other symmetric algorithms.

It is believed that the necessary cryptographic strength of the RC5 algorithm is sufficient for now [16]. However, with the rapid development of the capabilities of cloud data centers, the possibilities of cryptoanalysis are also growing.

Increasing the cryptographic stability of the RC5 crypto algorithm can be achieved by different methods. If we consider the modification of the cryptographic algorithm RC5, the main way for such an increase is the choice of a certain function of bit bias when encrypting the binary representation of blocks of information. The designation of a function in the context of "certain" is due to the fact that it is difficult to predict in advance which of the nonlinear shift functions would lead to a particularly "strong" result.

Theoretically, the correct choice of a nonlinear shift function to improve RC5 can reduce the number of rounds to 10, respectively, while the encrypted message cannot be opened using differential cryptoanalysis. At the same time, the computational complexity of the algorithm implementation remains comparable to the classical version of RC5.

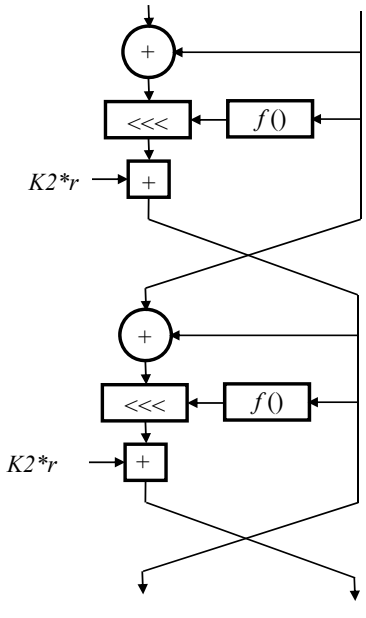

Fig. 2. Cryptographic transformation scheme in the RC5RA algorithm rounds

The maximum key size for RC5 that can be generated for the improved RC5 algorithm (also for the classic RC5) is 2,040 bits.

There is a variant of the classic symmetric cipher RC5 – this is the RC5RA cipher [17]. The latter cipher implements a cyclic shift in the encryption round by a variable number of bits, which depends on the round number of the algorithm. The value of the shift is determined not by the value of the lower $\log 2^w$ bits of another subblock but by some function $f()$ (Fig. 2). This shift function $f()$ can be any but nonlinear shift functions are used to increase cryptographic resistance.

The RC5RA algorithm produces a cryptographic transformation of the following form:

$$A_{i+1} = \left(\left(A_i \oplus B_i\right) << f(\cdot)\right) + S_{2i},$$

$$B_{i+1} = \left(\left(B_i \oplus A_i\right) << f(\cdot)\right) + S_{2i+1}.$$

Depending on the choice of the nonlinear function $f()$, a modification of the cryptographic algorithm can demonstrate a fairly serious improvement in its cryptographic resistance in general, compared to the basic algorithm of this family.

The proposed approach to improvement involves not so much modifying the algorithm but increasing its cryptographic resistance by selecting a nonlinear shift function. It, in turn, is the basis of this algorithm. Thus, the main point for improving RC5RA is the choice of nonlinear functions of the form $f(K, r)$, where the K is the round key, r is the number of the round. The essence of the choice of these functions for use as shift functions in RC5RA is to exclude the dependence of the nonlinearity of the output bit sequences of encryption blocks on the input bit sequences in the blocks during the formation of cyclic bit shifts.

One can suggest a way to improve the cryptographic strength of the RC5RA algorithm for encrypting data in the form using the following functions:

1. The first function takes the form:

$$f(K,r) = \left| r + \left[w \sin\left(wr \sum_{s=1}^{w} L_{bit_s} \right) \right] \right| \bmod w, \qquad (3)$$

where $m=r^2 \bmod w$;

r is the round number;

w is the length of half of the encoded block;

[x] is the integer part of the number x;

L_m is the block length from the initial message (clear text) in the encryption round;

L_{bit} is the binary representation of the L_m character encoded by the symbol of the word K, the length of which is w in the bit representation.

Fig. 3 shows the result of the obtained value of the cyclic bit shift for the length of the encoded word w=32 of the first round function.

Fig. 4 shows the dependence of the value of the cyclic bit shift on the round number and the length of the codeword for the first round function.

2. The second function takes the form:

$$f(K,r) = \left[w \exp\left(-\left| \frac{w}{10} \sin\left(wr \sum_{s=1}^{w} L_{bits_{m_s}} \right) \right| \right) \right] \bmod w. \qquad (4)$$

Fig. 5 shows the result of the cyclic bit shift value for the length of the encoded word w=32 for the second round function.

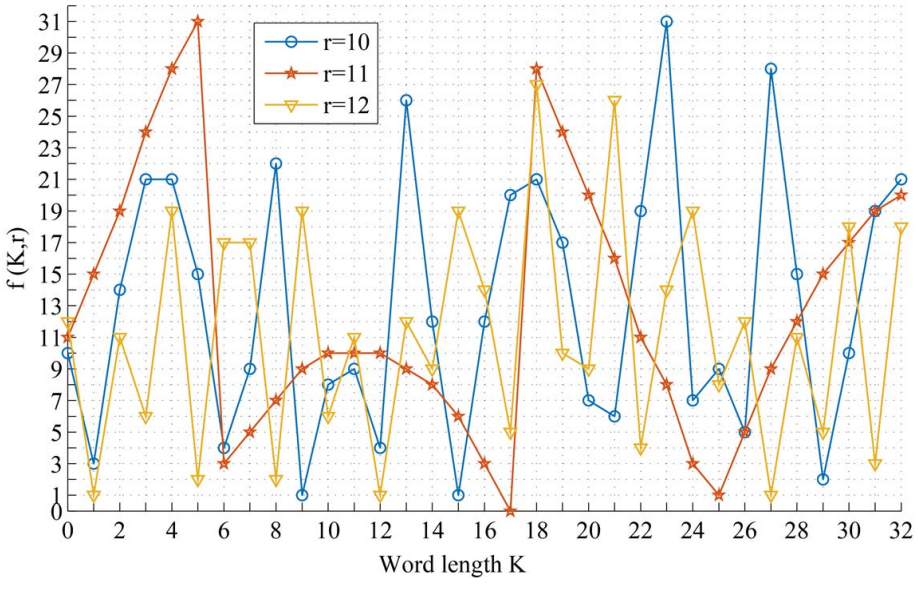

Fig. 3. Obtained values of the cyclic bit shift of the first function for the length of the encoded word *w*=32

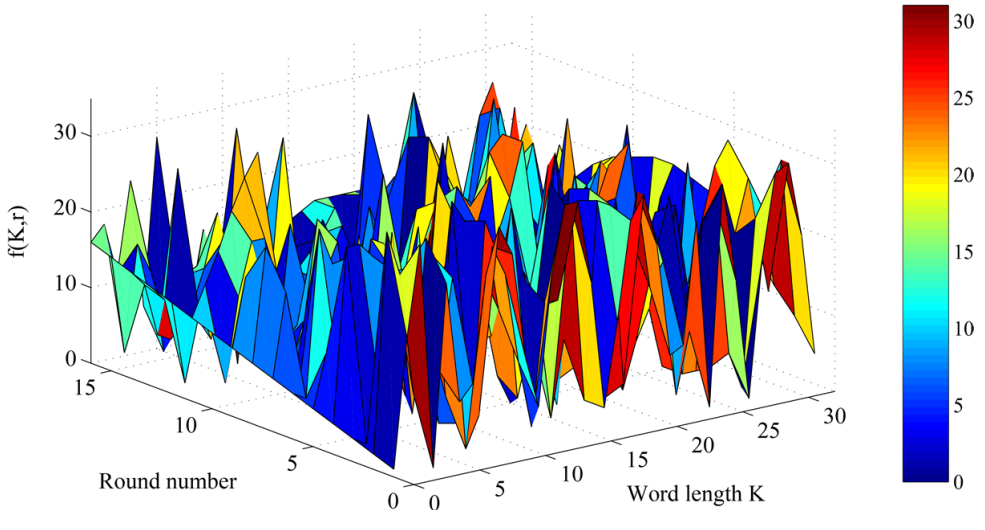

Fig. 4. Dependence of the value of the cyclic bit shift on the round number and the length of the codeword for the first round function

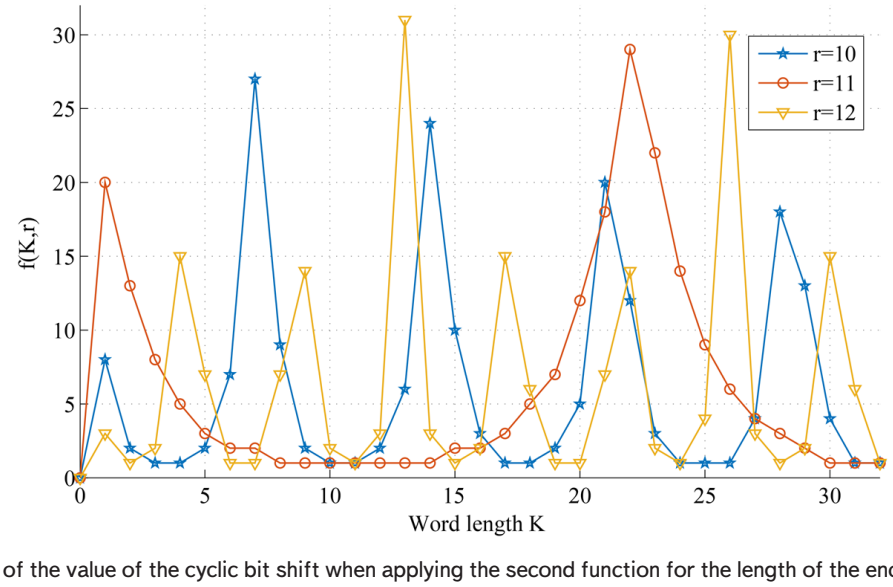

Fig. 5. The result of the value of the cyclic bit shift when applying the second function for the length of the encoded word *w*=32

Fig. 6 shows the dependence of the value of the cyclic bit shift on the round number and the length of the codeword of the second round function.

3) The third function takes the form:

$$f(K,r)=\left[3r\left|th\left(\frac{w}{10}\sin\left(wr\sum_{s=1}^{w}L_{bits_{m_s}}\right)\right)\right|\right]\bmod w. \quad (5)$$

Fig. 7 shows the result of the cyclic bit shift value for the length of the encoded word w=32 for the third round function.

Fig. 8 shows the dependence of the value of the cyclic bit shift on the round number and the length of the codeword when applying the third round function.

4) The fourth function takes the form:

$$f(K,r)=\left(\left|\begin{array}{l}rw\sum_{s=1}^{w}L_{bits_{m_s}}+\\+\left[rw\cos\left(3\sum_{s=1}^{w}L_{bits_{m_s}}+2r\right)\right]\end{array}\right|\right)\bmod w. \quad (6)$$

Fig. 9 shows the result of the cyclic bit shift value for the length of the encoded word w=32 for the fourth round function.

Fig. 10 shows the dependence of the value of the cyclic bit shift on the round number and the length of the codeword when applying the fourth round function.

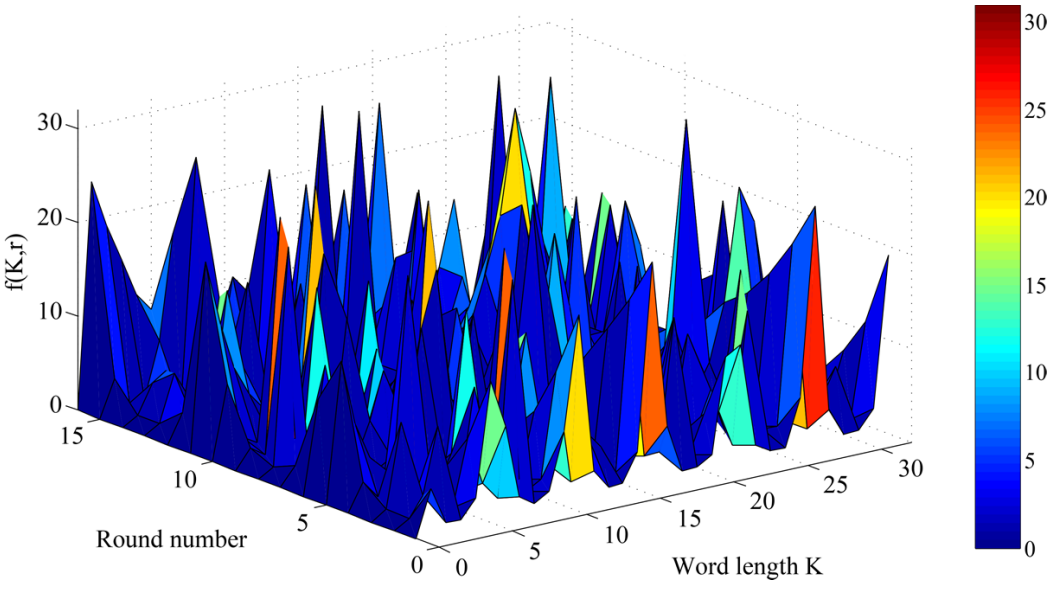

Fig. 6. Dependence of the value of the cyclic bit shift on the round number and length of the codeword for the second round function

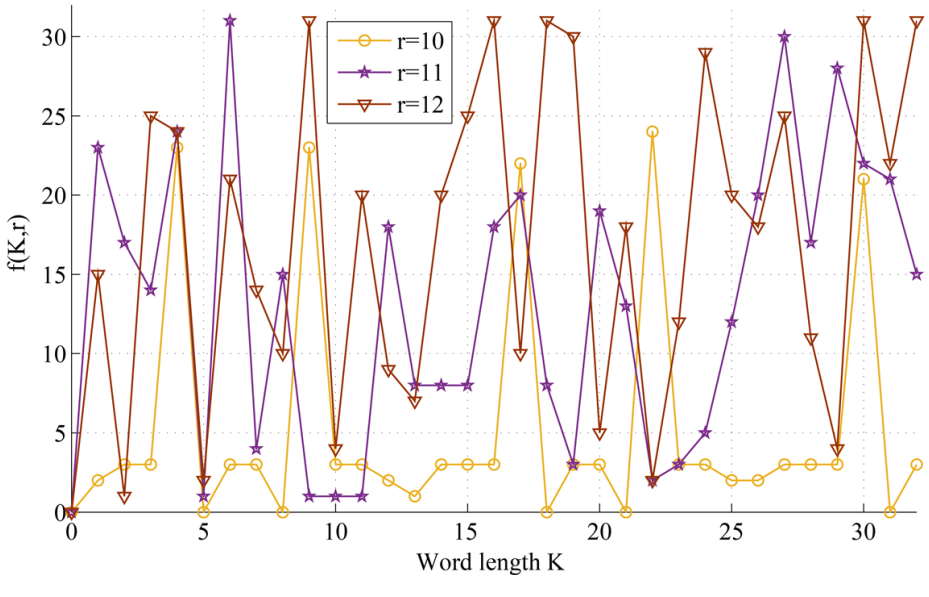

Fig. 7. The result of the value of the cyclic bit shift at the length of the encoded word w=32 and the application of the third function

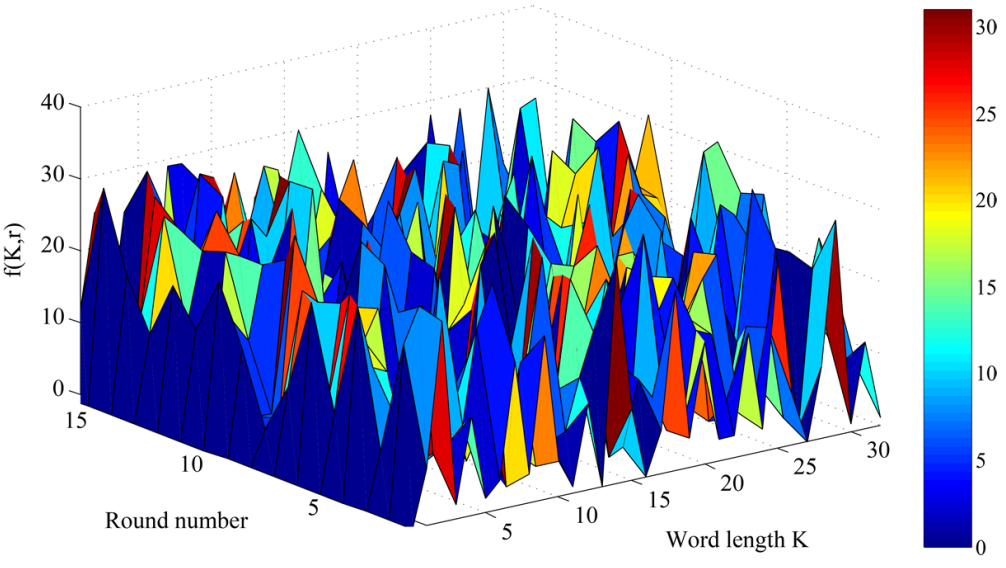

Fig. 8. Dependence of the value of the cyclic bit shift on the round number and length of the codeword when applying the third round function

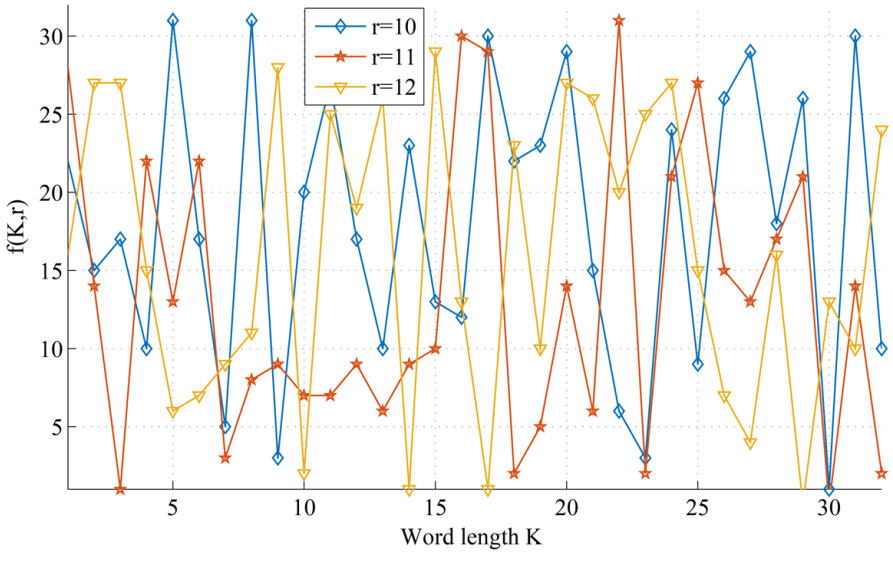

Fig. 9. The result of the value of the cyclic bit shift when applying the fourth function for the length of the encoded word *w*=32

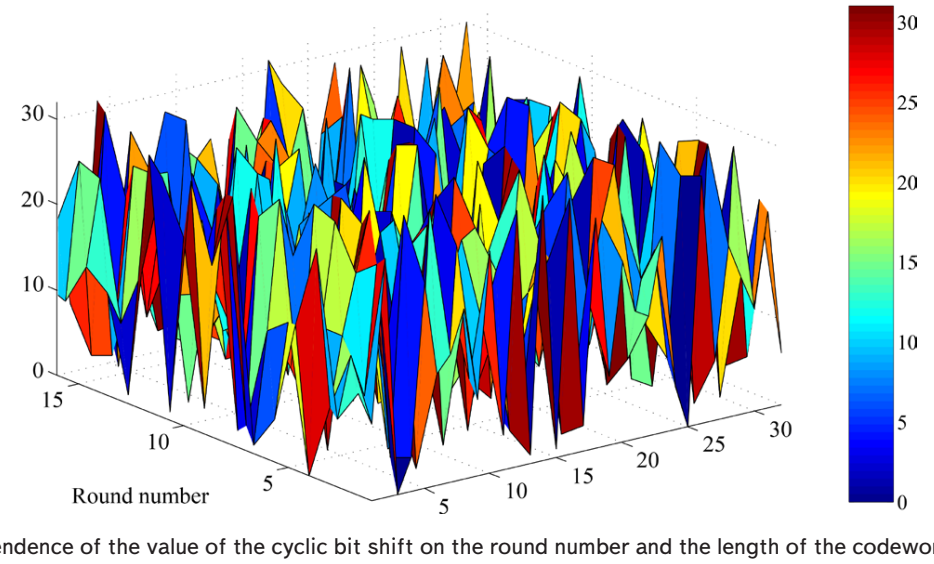

Fig. 10. The dependence of the value of the cyclic bit shift on the round number and the length of the codeword when applying the fourth round function

5) The fifth function takes the form:

$$f(K,r)=\left(\left|\sum_{s=1}^{w}L_{bits_{m_s}}+\left|+\left[\log_2\left(1+rw\left|\sum_{s=1}^{w}L_{bits_{m_s}}\right|\right)\right]\bmod r\right|\right)\bmod w. \quad (7)$$

Fig. 11 shows the result of the cyclic bit shift value for the length of the encoded word w=32 for the fifth round function.

Fig. 12 shows the dependence of the value of the cyclic bit shift on the round number and the length of the codeword when applying the fifth round function.

The above nonlinear functions have a range of valid values for the entire period of their existence. This satisfies the condition of forming coefficients in the round.

As the base for the modernization, a version of the RC5 family algorithm, RC5RA, was used. In this version of the classic RC5, the number of shift bits is determined using some function of another "sub-block". In practical implementation, a sequence of data of different bit depths was used as a key (16, 32, 64). The key sequence was generated using a standard built-in congruent pseudo-random number generator of the programming language, underlying the practical implementation of the modification of the RC5RA algorithm.

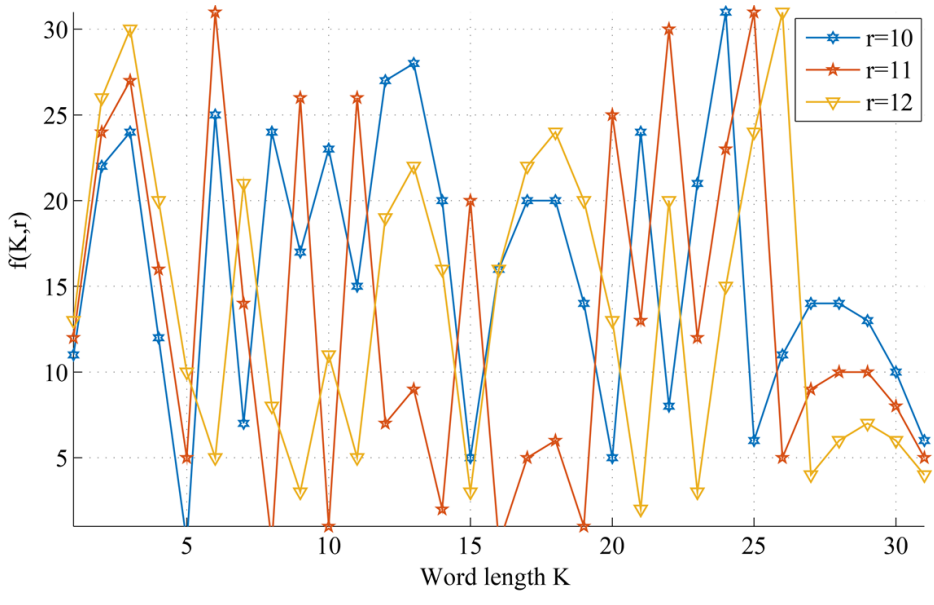

Fig. 11. The result of the value of the cyclic bit shift when applying the second function for the length of the encoded word w=32

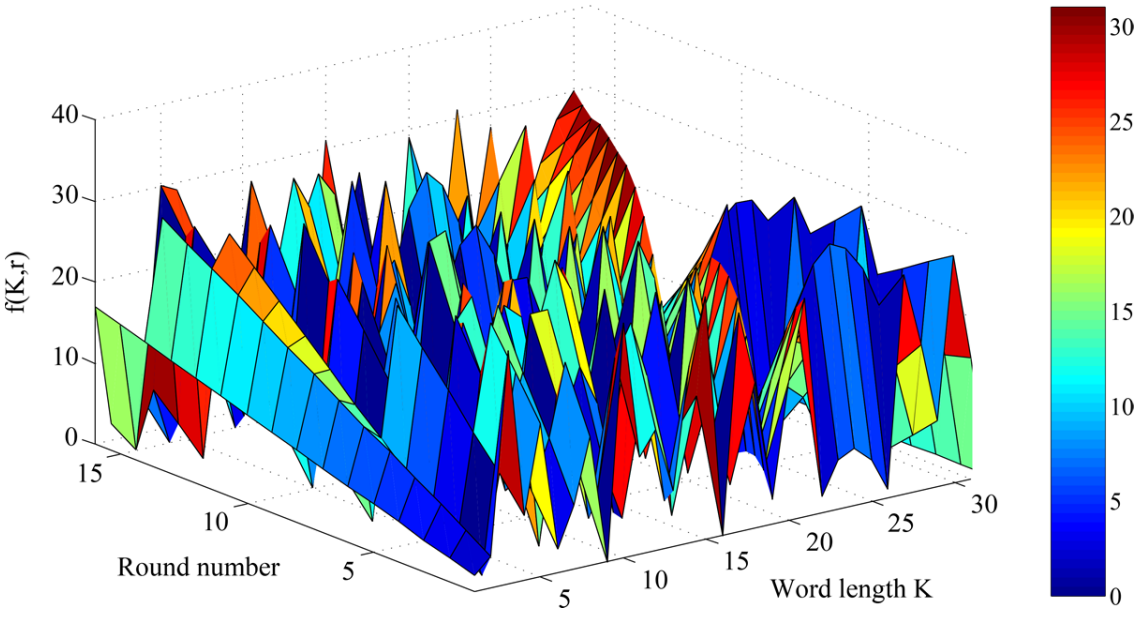

Fig. 12. Dependence of the value of the cyclic bit shift on the round number and length of the codeword when applying the fifth round function

5. 2. Test samples (datasets) for encryption and decryption

All the considered nonlinear functions (3) to (7) can be used to improve the cryptographic stability of the classical RC5 algorithm.

To test theoretical assumptions about increasing the cryptographic strength of encryption, it is necessary to simulate the operation of the algorithm and form a test sample of open data for encryption.

It can be assumed that for the block symmetric cipher RC5 or modifications based on it, there are no confirmed direct dependences between the sizes or multiplicity of the sample files with clear text for encryption on the result of encryption. Or, especially, the influence of these parameters on the cryptographic resistance of the received ciphertext or the appearance of collisions.

The graphical and textual data presented in the test samples contain only numerical integer data. When encoding pixel graphics and texts using well-known and relevant coding systems, we receive a file in the form of a vector sequence of integers.

Input datasets for encryption include test sets of text data, 81 to 1,968 Kilobytes in length, and graphical sets, with lengths from 351 to 3,412 Kilobytes. Files that contain English texts are taken as text inputs. Graphic data are a set of simple images of standard formats, such as the jpg format. The formation of keys is made on the basis of the pseudo-random number generator, which is a standard component of the software development environment for encrypting and decrypting data based on the classical RC5RA algorithm using the nonlinear functions described above, in order to increase the cryptographic strength of the resulting algorithm modification. Such sets of algorithm parameters are used for encryption and decryption, determining the quantitative and qualitative characteristics obtained as a result of the development of variants of modernizations of the RC5 algorithm. The formation of the key, as noted above, is made using a pseudo-random number generator based on modular arithmetic, which forms the keys immediately in the bit representation. The number of generated keys for each set of input data is 10 keys.

5. 3. Software implementation and simulation of the modified RC5 encryption algorithm with different sets of standard parameters

This is necessary to define the quantitative and qualitative parameters of the resulting RC5 modification. The software application was in the MATLAB simulation environment. The built-in MATLAB language was used as a programming language. In the software interface, it was possible to manually change the encryption parameters of the RC5 crypto algorithm.

For encryption, one can load text (txt) and image files (jpg) into the program, enter a keyword (K), round parameters, and the length of the encoding block (Fig. 12).

To test the modified RC5 algorithm and obtain some results at run time, identical key parameter data, encryption rounds, and the function number for each series of test samples were used.

The resulting MATLAB software application consists of the following main blocks:

– user interface;

– subroutines for encrypting files of test samples (graphic and text);

– subprograms for generating encrypted data and writing them to the current folder;

– subprograms for extracting and decrypting information from graphic and text files.

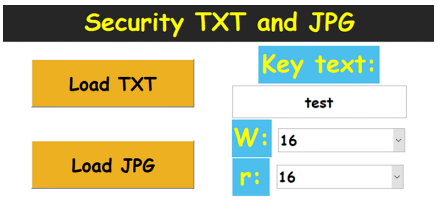

Fig. 13. General view of the graphical user interface of the software implementation of the RC5 algorithm

Next, the quantitative and qualitative parameters of the modified variants of the RC5 crypto algorithm were determined.

The obtained results regarding the modified RC5 algorithm operation using the shift functions (3) to (7) and various parameters of the crypto algorithm are shown in Fig. 14–17.

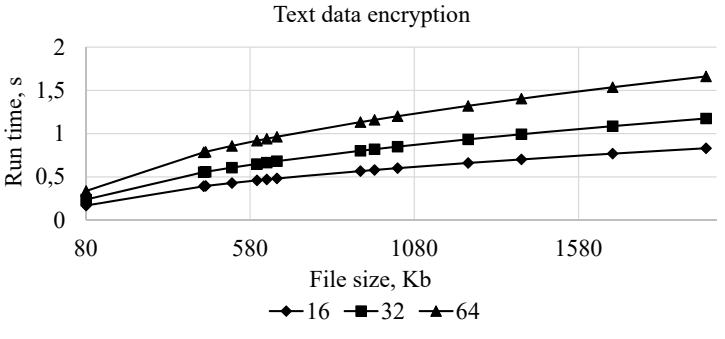

Fig. 14. Program run time when encrypting text information at parameters r=16 and values k=16, 32, 64 bits

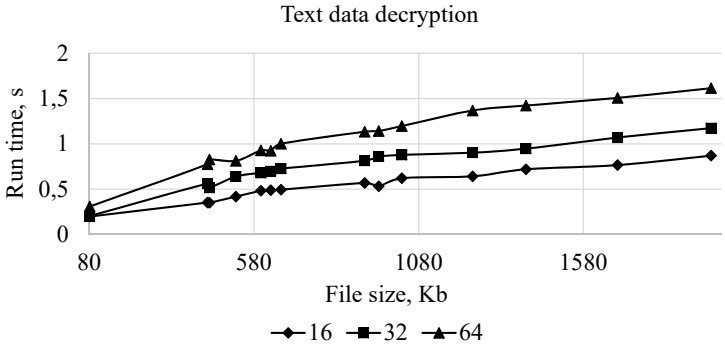

Fig. 15. Program run time when decrypting text information at parameter r=16 and values K=16, 32, 64 bits

Fig. 13–16 show the results of the program run when encrypting and decrypting text and graphic documents of a test sample of data obtained at the classical round shift and the use of the five nonlinear functions discussed above. The encryption and decryption time parameters for the classic RC5RA variant (at a constant shift value) coincide with the same time parameters for five shift modifications using nonlinear functions. The corresponding

numerical results are given in Tables 1, 2. Matches appear on all test samples used.

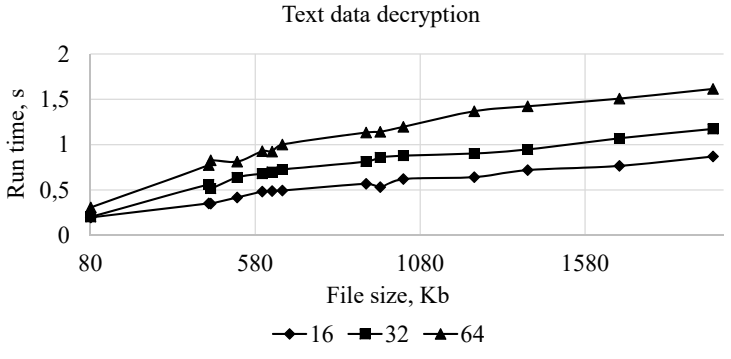

Fig. 16. Program run time when encrypting graphic data at r=16 and values K=16, 32, 64 bits

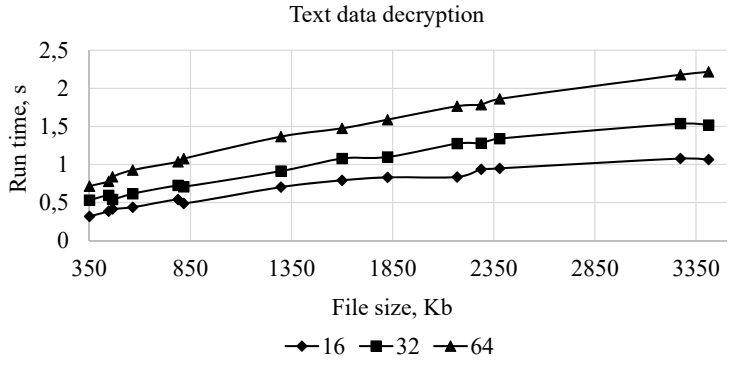

Fig. 17. Program run time when decrypting graphic data at r=16 and values K=16, 32, 64 bits

Tables 1, 2 give the run time of encryption and decrypa tion of graphic information with the number of rounds r=16 and different values of the W parameter.

Table 1

Run-time of encryption and decryption of text files with the number of rounds r=16

File size, Kbytes	Program run-time, seconds: Encryption/decryption w=16/w=16	Program run-time, seconds: Encryption/decryption w=32/w=32	Program run-time, seconds: Encryption/decryption w=64/w=64
81	0.17/0.20	0.24/0.20	0.34/0.31
439	0.39/0.35	0.55/0.56	0.78/0.78
445	0.40/0.35	0.56/0.52	0.79/0.83
525	0.43/0.42	0.61/0.64	0.86/0.81
601	0.46/0.48	0.65/0.68	0.92/0.93
631	0.47/0.49	0.67/0.70	0.94/0.92
662	0.48/0.49	0.68/0.73	0.96/1.00
916	0.57/0.57	0.80/0.81	1.13/1.13
959	0.58/0.53	0.82/0.86	1.16/1.14
1,029	0.60/0.62	0.85/0.88	1.20/1.20
1,244	0.66/0.64	0.93/0.90	1.32/1.37
1,406	0.70/0.72	0.99/0.95	1.40/1.42
1,684	0.77/0.77	1.09/1.07	1.54/1.51
1,968	0.83/0.87	1.17/1.17	1.66/1.61

Table 2

Run-time of encryption and decryption of the modules of graphic files at number of rounds r=16

File size, Kbytes	Program run-time, seconds: Encryption/decryption w=16/w=16	Program run-time, seconds: Encryption/decryption w=32/w=32	Program run-time, seconds: Encryption/decryption w=64/w=64
351	0.35/0.32	0.50/0.53	0.70/0.72
446	0.40/0.39	0.56/0.60	0.79/0.78
465	0.40/0.41	0.57/0.54	0.81/0.84
565	0.45/0.44	0.63/0.62	0.89/0.93
789	0.53/0.54	0.74/0.73	1.05/1.03
817	0.54/0.49	0.76/0.71	1.07/1.08
1,298	0.67/0.70	0.95/0.91	1.35/1.37
1,600	0.75/0.79	1.06/1.08	1.50/1.47
1,825	0.80/0.83	1.13/1.10	1.60/1.59
2,169	0.87/0.84	1.23/1.27	1.74/1.77
2,288	0.90/0.94	1.27/1.28	1.79/1.79
2,379	0.91/0.95	1.29/1.34	1.83/1.86
3,273	1.07/1.08	1.52/1.54	2.14/2.18
3,412	1.09/1.06	1.55/1.52	2.19/2.22

5. 4. Choosing a cryptoanalysis method and determining the cryptographic resistance of the obtained variants of modifications of the RC5 cryptosystem

The structure of the RC5 algorithm is very simple and convenient not only for implementation but also for assessing its crypto-analytical stability. The same can be attributed to a modified version of the algorithm, RC5RA.

In the experimental process, it makes sense to use linear and differential cryptoanalysis. Simple texts were used for cryptoanalysis. In general, an attack based on known open texts, in which standard passages are present in the ciphertext, and their meaning is known to the analyst in advance, is effective in many cases [18].

Several procedures of cryptoanalysis relative to the classical RC5 cipher are described, which can be applied to its modernized version. In work [14], it was found that an attack by linear cryptoanalysis on the classical RC5 with 6 rounds, using 2^{57} known open texts, almost always fails. That is, attempts at linear cryptoanalysis of ciphertext after 6 rounds of encryption do not give results. Requirements for clear text are practically not completed by opening the cipher after 6 rounds of encryption. Differential attack on RC5-32/12/16 using 2^{63} selected plain texts is described in [15]. Improvement of this attack up to 512 times was given in [19, 20]. The idea of this type of cryptoanalysis is to find open texts so that there are no repetitions in the first few half-rounds. Once these open texts have been identified, the differential cryptoanalysis described in [14] can be performed successfully with greater probability. This causes weak avalanche properties and high key dependence of the properties of the cipher.

Given this, differential cryptoanalysis (attack) based on data statistics about the sequences of input and output bits modernized by nonlinear function shifts was chosen for crypto analysis in order to determine the degree of dependence between them.

The number of texts needed for crypto analysis depended on how soon a similar "differential pair" to the desired one could be met. The test set consisted of similar Latin texts, compiled on the principle of "A...Z". Accordingly, Table 3 gives the number of input (simple) texts required to crack the modified classical RC5RA algorithm with a different number of rounds (from 10 to 14).

Table 3

The number of input (simple) texts required to crack the modified RC5RA algorithm with different numbers of rounds

Number of rounds/Number of nonlinear function	r=10	r=11	r=12	r=13	r=14
Function one	2^{46}	2^{50}	2^{56}	2^{62}	2^{80}
Function two	2^{46}	2^{52}	2^{58}	2^{66}	2^{88}
Function three	2^{46}	2^{54}	2^{60}	2^{66}	2^{90}
Function four	2^{46}	2^{54}	2^{62}	2^{66}	2^{92}
Function five	2^{46}	2^{56}	2^{64}	2^{68}	2^{96}

Our differential analysis showed that the improved version of RC5 has a high level of protection against attacks for pairs of input and output bit sequences with different numbers of rounds.

5. 5. Analyzing the results obtained in devising a method to increase the cryptographic resistance of the RC5 algorithm

With the number of rounds r=11, the improved version of RC5, due to the use of nonlinear shift functions, shows 2^{10} greater cryptographic resistance to attacks of this type (when using the fifth function).

Table 3 illustrates that the number of open texts for cracking the modified RC5RA by the selected differential analysis method at 10 rounds of classical shift encryption is 2^{46} in the modified version (for any of the five nonlinear shift functions applied). For 14 rounds, the number of open texts for hacking the modified RC5RA by the selected cryptoanalysis method is 2^{80} to 2^{96} (depending on which nonlinear function is used).

6. Discussion of results of devising the method related to its application; promising areas for further research

As can be seen from the original study, the computational complexity of the resulting modifications almost does not depend on the algebraic complexity of the applied nonlinear function. However, when the modified RC5RA algorithm processes real numbers (floating-point numerical data), it is necessary to separately simulate the resulting modifications of the RC5RA algorithm. After that, it is necessary to investigate the time characteristics of the operation of these modifications to clarify their computational complexity and establish the impact on the computing unit and the amount of RAM of the computer system.

Along with the ability of the improved cipher, identified and confirmed in the authentic work, to significantly increase its resistance through the use of alternative round-shift mechanisms, a given algorithm is practically devoid of the ability to reduce its computational complexity. That does not apply to cases of changing the length of the key or the number of rounds of encryption.

The disadvantage of the obtained implementation, revealed in the course of the authorized study, is the fact that the results obtained on test samples cannot be unambiguously transferred to real data (to files too small or too large).

In further research, it is necessary to formulate approaches for the formation or defining the requirement for the nonlinearity of functions that are planned to be used as round-shift functions to enhance the cryptographic resistance of the RC5 algorithm.

It can be assumed that when using a genetic algorithm for the cryptoanalysis of a given block cipher, the issue of forming the initial population could arise. After all, with a variable key length in RC5, the population would also be variable, and this greatly complicates the implementation of the cryptoanalysis algorithm. Theoretically, the genetic algorithm, due to its inherent parallelism, has the ability to significantly accelerate crypto analysis but in practice, the cryptoanalysis time was 38 hours of continuous operation on a test hardware quad-core platform (AMD Ryzen 5 2600).

One should note another weakness of the original work, namely, the practical impossibility of obtaining the operational parameters of the modified versions of RC5 at all possible values of keys, rounds, blocks, etc. However, such a computational experiment (cryptoanalysis) can take too long (especially given the available hardware resources). And the representation of the results of its operation would go beyond reasonable limits of presentation.

For a deeper study into the properties of the proposed modification of the RC5RA algorithm, it is desirable to develop a specific method of cryptoanalysis, the ad hoc type. However, the goal of the development is to obtain a modification of the RC5RA suitable for practical work on "weak" computing platforms. Devising a new method of cryptoanalysis for all variants of implementations of the RC5RA algorithm modified by the shift functions is a promising task.

As a result of consideration of several described variants of implementation of the RC5 block encryption algorithm, nonlinear functions were selected to implement a bit round shift.

The software implementation of the modified RC5RA algorithm in the built-in programming language MATLAB with different sets of standard parameters and test samples has made it possible to determine the quantitative and qualitative parameters of the modified crypto algorithm. The results obtained when applying nonlinear functions are almost similar to the implementation of the RC5RA algorithm when using a standard shift function.

The selected test sets of input data enable determining the quantitative and qualitative characteristics of the obtained options for upgrading the RC5 algorithm for different types of data (Fig. 14–17). They show that when the key length is increased by a multiple of 2, starting from 16 bits, the file encryption time increases from 0.5 seconds for a 3,350-KB file to about 1.5 seconds for a 3,350-KB file (key size, k=32). This parameter is a good result for a symmetric algorithm. This indicator is confirmed by the values given in Tables 1, 2.

For the chosen method of cryptoanalysis (an attack based on known open texts in which standard passages are present in the ciphertext, and their meaning is known to the analyst in advance), the cryptographic strength of the RC5RA crypto algorithm modification obtained when using 5 nonlinear functions has been determined. Compared to the classical implementation, with round values from 10 to 14 (Table 3),

the stability improvement amounted to 2^{12} times (when using the first function) to 268,435,456 times (when using the fifth shift function).

The study of the persistence of the RC5RA crypto algorithm modification using the above-mentioned shift functions was also carried out using classical differential analysis. The test dataset (texts) consisted of the first 5 characters of the Latin alphabet. These specially prepared texts contained 2 and 4 identical consecutive characters. For different pairs of texts from this sample, a "differential" was calculated. Based on the obtained "differential", the "differential" of other pairs of encrypted texts was evaluated. This estimate is a limit value of 25 % when the number of rounds in the crypto algorithm takes a value from 10 to 14, as shown in Table 3. This means that with a probability of 0.25 with data on such a structure, it is possible to open half of the key or the entire key completely.

Based on the results given in Table 3 (the number of texts), it is recommended to use the number of rounds in the crypto algorithm not less than 11. The obtained RC5RA modification's practical crypto resistance is achieved at the number of rounds exceeding 16.

As a result of the analysis and generalization, the question of the influence of the nature of the round shift in RC5 on increasing the cryptographic stability of this crypto algorithm was partially answered. Fig. 4, 6, 8, 10, 12 show that the best values of cryptographic resistance of the modified algorithm are demonstrated by those functions whose variance of output values is most homogeneous. However, this issue needs more research and verification.

The problematic points in the original study include the need to search for and formulate a specific method of cryptoanalysis applicable to each of the five modifications of the RC5RA algorithm obtained. This is necessary to elucidate the full spectrum of its cryptographic strength. As a recommendation, it is possible to highlight the desirability of implementing the obtained modifications on various hardware platforms, in different programming languages.

One of the promising areas of further research is the search for a shift function with an ideal variance at the output.

The study results confirm the possibility of enhancing the cryptographic power of the RC5 encryption algorithm (RC5RA variant) without increasing the computational complexity of the algorithm itself or increasing the number of encryption rounds. At the same time, the resulting solution makes it possible to move away from the use of cryptographic computer protection in the form of specialized hardware units. This approach is important in the application of RC5 cryptographic transformation to protect network traffic in networks based on the Internet of Thing technology. At present, such and topologically similar networks employ other methods of protection (for example, firewalls) to protect network traffic. Thus, it becomes possible to implement the additional protection of data privacy in networks and systems on the Internet of Things and similar ones.

7. Conclusions

1. As a result of our consideration of the existing variants of the RC5 cipher, the most practically suitable version of the RC5 symmetric encryption algorithm, RC5RA, was chosen. That has made it possible to fully realize the potential for increasing cryptographic strength (due to the possibility of using nonlinear round shift functions); it has increased resistance to differential cryptoanalysis.

Five nonlinear functions were also selected for use as round-shift functions in the RC5RA algorithm. The use of these functions has significantly improved the cryptographic resistance of modified versions of the RC5RA algorithm.

2. Test datasets have been defined to test the operation of the modified and classical version of the RC5 algorithm under encryption and decryption modes. Thus, numerical data of the integer type, with a length from 81 to 1,968 Kilobytes, were used as text data. As graphic ones – numerical data, with a length from 351 to 3,412 Kilobytes.

3. The software implementation of the obtained modifications of the RC5RA algorithm was carried out in the MATLAB environment. When testing the program using test samples, it was found that the program's operating time when encrypting and decrypting text and graphic information at the parameters r=16 and values K=16, K=32, K=64. For the classical and modified versions of the algorithm, the resulting speed of the cryptographic transformation is identical for both the classical and modified versions (it differs within the statistical error). This allows us to argue that the computational complexity of modified versions has not increased. It should be noted that the choice of the language for software implementation is not crucial in this case. This is due to the fact that the original work reported relative performance indicators of the algorithms, which could be used to judge the difference in the operation of the classical RC5 algorithm and its modifications obtained on the basis of the RC5RA version. It can be assumed that when using other programming languages, the resulting difference in the performance indicators of the algorithms of the classical and described modifications of RC5RA would be preserved.

4. As a method of cryptoanalysis, the differential method of cryptoanalysis was used, which is based on statistical methods for estimating the probabilities of the appearance of identical sub-block round sequences for each round. It has a sequence size of 8 bits. A given method was chosen due to its relatively low hardware requirements and high efficiency.

5. The authentic study confirms the data obtained in work [11] regarding the fact that the RC5 algorithm is highly dependent on the rotation of data in it. Therefore, it became clear to us that the probability that the differential pair would be able to avoid differences in the number of turns is much higher than the developers of the RC5 expected.

The result of the cryptoanalysis has established that the number of open texts for hacking for 10 rounds by the classic shift is 2^{56} in the modified version (2^{44} in the classic version), and, for 14 rounds, it is from 2^{80} to 2^{96} (2^{56} in the classic version). Moreover, already at r=11, the cryptographic resistance of this algorithm when applying the fifth nonlinear function is 2^{56}. The obtained indicators of increased cryptographic resistance make it possible to apply the resulting modifications of RC5RA in computer systems and network routers with low performance, which is especially important for the technology and concept of IoT and "smart home".

References

1. Recommendation X.200 (07/94). Available at: https://www.itu.int/rec/T-REC-X.200-199407-I
2. Understanding Layer 2 Encryption. Technical Whitepaper (2013). SafeNet. Available at: https://newberrygroup.com/wp-content/uploads/2017/10/understanding-layer-2-encryption-wp-en-v2-dec022013-web.pdf
3. Rivest, R. L. (1995). The RC5 encryption algorithm. Lecture Notes in Computer Science, 86–96. doi: https://doi.org/10.1007/3-540-60590-8_7
4. OpenSSL. Cryptography and SSL/TLS Toolkit. Available at: https://www.openssl.org/
5. RSA® BSAFE® Crypto-J JSAFE and JCE Software Module 6.2.4 Security Policy Level 1 (2020). Available at: http://csrc.nist.gov/CSRC/media/projects/cryptographic-module-validation-program/documents/security-policies/140sp3172.pdf
6. Blozva, A., Kydyralina, L. M., Matus, Y. V., Osypova, T. Y., Sauanova, K., Brzhanov, R. T., Shalabayeva, M. (2021). IoT Devices Integration and Protection in available Infrastructure of a University computer Network. Journal of Theoretical and Applied Information Technology, 99 (8), 1820–1833. Available at: http://www.jatit.org/volumes/Vol99No8/11Vol99No8.pdf
7. Luzhetskyi, V., Horbenko, I. (2015). Metody shyfruvannia na osnovi perestanovky blokiv zminnoi dovzhyny. Zakhyst informatsiyi, 17 (2), 169–175.
8. Biryukov, A., Khovratovich, D. (2009). Related-Key Cryptanalysis of the Full AES-192 and AES-256. Lecture Notes in Computer Science, 1–18. doi: https://doi.org/10.1007/978-3-642-10366-7_1
9. Garfinkel, S. (1994). PGP: Pretty Good Privacy: Pretty Good Privacy. O'Reilly Media, 432.
10. Schneier, B. (1994). Description of a new variable-length key, 64-bit block cipher (Blowfish). Lecture Notes in Computer Science, 191–204. doi: https://doi.org/10.1007/3-540-58108-1_24
11. Biryukov, A., Kushilevitz, E. (1998). Improved cryptanalysis of RC5. Advances in Cryptology – EUROCRYPT'98, 85–99. doi: https://doi.org/10.1007/bfb0054119
12. Furlong, M., Heys, H. (2005). A timing attack on the CIKS-1 block cipher. Canadian Conference on Electrical and Computer Engineering, 2005. doi: https://doi.org/10.1109/ccece.2005.1556916
13. Matsui, M. (1994). Linear Cryptanalysis Method for DES Cipher. Lecture Notes in Computer Science, 386–397. doi: https://doi.org/10.1007/3-540-48285-7_33
14. Kaliski, B. S., Yin, Y. L. (1995). On Differential and Linear Cryptanalysis of the RC5 Encryption Algorithm. Lecture Notes in Computer Science, 171–184. doi: https://doi.org/10.1007/3-540-44750-4_14
15. Knudsen, L. R., Meier, W. (1997). Differential cryptanalysis of RC5. European Transactions on Telecommunications, 8 (5), 445–454. doi: https://doi.org/10.1002/ett.4460080503
16. Aggregate Statistics (2021). RC5-72 / Overall Project Stats. Available at: https://stats.distributed.net/projects.php?project_id=8
17. Panasenko, S. P. (2009). Algoritmy shifrovaniya. Spetsial'niy spravochnik. Sankt-Peterburg: BHV, 576.
18. Welchman, G. (1982). The Hut Six Story: Breaking the Enigma Codes. Harmondsworth: Allen Lane.