МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ
ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ
УНИВЕРСИТЕТ РАДИОЭЛЕКТРОНИКИ

РАДИОТЕХНИКА

Всеукраинский межведомственный научно-технический сборник

ТЕМАТИЧЕСКИЙ ВЫПУСК
«ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ»

Основан в 1965 г.

В Ы П У С К 1 9 3

Харків
Харківський національний
університет радіоелектроніки
2018

УДК 621.3

Сборник включен в список специальных изданий ВАК Украины по физико-математическим и техническим наукам.
Регистрационное свидетельство КВ № 12098-969 ПР от 14. 12. 2006.
Ответственность за содержание статей несут авторы.

Ответственный за выпуск: И.Д. Горбенко, д-р техн. наук, проф.
Технический секретарь Е.С. Полякова

Рекомендовано Ученым советом Харьковского национального университета радиоэлектроники, протокол № 72 от 15.05.2018.

Адрес редакционной коллегии: Харьковский национальный университет радиоэлектроники (ХНУРЭ), просп. Науки, 14, Харьков, 61166, тел. (0572) 7021-397.

Сборник «*Радиотехника*» *включен в Каталог подписных изданий Украины, подписной индекс* ***08391***

СОДЕРЖАНИЕ ЗМІСТ

МЕТОДЫ, МЕХАНИЗМЫ И АЛГОРИТМЫ КРИПТОГРАФИЧЕСКИХ ПЕРСПЕКТИВНЫХ ПРЕОБРАЗОВАНИЙ

МЕТОДИ, МЕХАНІЗМИ ТА АЛГОРИТМИ КРИПТОГРАФІЧНИХ ПЕРСПЕКТИВНИХ ПЕРЕТВОРЕНЬ

МЕТОДЫ ПОСТРОЕНИЯ ЗАЩИЩЕННЫХ ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

МЕТОДИ БУДУВАННЯ ЗАХИЩЕНИХ ТЕЛЕКОМУНІКАЦІЙ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

CONTENT

METHODS, MECHANISMS AND ALGORITHMSC OF CRYPTOGRAPHIC PERSPECTIVE TRANSFORMATIONS

METHODS FOR CONSTRUCTION OF PROTECTED TELECOMMUNICATIONS AND INFORMATION TECHNOLOGIES

МЕТОДЫ, МЕХАНИЗМЫ И АЛГОРИТМЫ КРИПТОГРАФИЧЕСКИХ ПЕРСПЕКТИВНЫХ ПРЕОБРАЗОВАНИЙ

UDC 004.056.55

I.D. GORBENKO, Doctor of technical sciences, O.G. KACHKO, Ph.D. of technical sciences, M.V. YESINA, Ph.D. of technical sciences

GENERAL STATEMENTS AND ANALYSIS OF THE END-TO-END ENCRYPTION ALGORITHM NTRU PRIME IIT UKRAINE

Introduction

Today the questions concerning the stability of existing cryptographic algorithms to quantum cryptanalysis become topical. This is due, first of all, to the rapid development in the field of quantum computers. Therefore, it is necessary to evaluate the possibilities of quantum cryptanalysis and, on this basis, to modify existing cryptographic algorithms (for example, to increase the size of key parameters) or to create new cryptographic algorithms that will be resistant to attacks on quantum computers.

So, in light of the foregoing, NIST USA has announced a competition for the post-quantum algorithms search, including end-to-end encryption algorithms (E2EE) [7]. It is definitely known that for practical application algorithms must meet the requirements of stability, performance and should be low-resource. Submissions were received by NIST until November 30, 2017. They relate to: E2EE asymmetric algorithms and electronic signature (ES). Subsequently, their detailed analysis and comparison is expected, with a period of up to 3 years. This indicates the significant complexity of the problem to be solved.

With the participation of the authors of this article for the NIST USA competition, a cryptographic algorithm for NTRU Prime IIT Ukraine [8, 9], developed using NTRU [1] and NTRU Prime [2], was presented.

This article describes the differences between the proposed cryptographic algorithm and ANSI standard [1] and the NTRUPrime algorithm [2]. In each paragraph, attention is paid to the difference, the possibility of optimization, and the results of the research.

All experiments to determine the computational complexity were performed on the Intel(R) Core(TM) processor i5-4440 CPU @ 3.10 GHz. The spent time is determined in the processor tacts.

The objective of this paper is a general overview and description of the proposed cryptographic transformation end-to-end encryption «NTRU Prime IIT Ukraine», implementation specificity, comparison of the main characteristics and indicators, as well as the definition of differences from existing NTRU-like cryptographic algorithms.

1. Denotations and abbreviations

Most of the terms and denotations are the same as those adopted in the ANSI [1]. For convenience, we give them in this document.

Table 1

Denotations and abbreviations, that are used

db	The number of bits for the random component that is used during encryption. It is determined by cryptostability. Coincides with the length of the used hash.
n	Polynomial order. It determines the number of its coefficients. Prime for which the polynomial $x^n–x–1$ is irreducible.
q	The module, prime, by which the polynomial coefficients are given in $(Z/qz)[X](x^n–x–1)$; $q \geq 48t+3$.
F	Polynomial in $(Z/3z)[X](x^n–x–1)$. Specifies the private key.
G	Polynomial in $(Z/3z)[X](x^n–x–1)$. It is used to calculate the public key.
R	Blinding polynomial in $(Z/3z)[X](x^n–x–1)$.
f	Polynomial, which is calculated using the formula: $f=3F+1$.
h	Polynomial in $(Z/qz)[X](x^n–x–1)$. Calculated by the formula $h=3G/f$.
t	It determines the number of non-zero elements in the private key and message. For F, the number of 1 and -1 is $2t$, for G the number of 1 is $n/3+1$, and the number of -1 is $n/3$, for r – the number of 1 and -1 is $n/3$, for message the number of 0, 1 and -1 is not less than t.

Continuation of Table 1

Q	The module, prime, by which the polynomial coefficients are given in $(Z/qz)[X](x^n–x–1)$; $q≥48t+3$.
E_p	Encrypted message, the polynomial in $(Z/qz)[X](x^n–x–1)$.
E_b	Encrypted message, bytes string.
Hlen	The length of the hash (bit) coincides with *db*.
k	Security Level.
m	Message to encrypt, bytes string.
M	Message after addition the random string and other information. It is used to encryption and decryption.
MGF	Mask generation function
qBits	Number of bits to specify the number q.
maxMsgLenBytes	Maximum length of the message.

2. Basic and additional parameters

2.1 Basic parameters

k – Security level. It is defines the remaining parameters. In [2], this parameter is indicated by λ.

n –polynomial order. It determines the number of its coefficients. The prime for which the polynomial $x^n–x–1$ is irreducible. In [2], this parameter is indicated by p.

t – determines the number of non-zero elements in a private key and message. For F, the number of 1 and –1 is $2t$, for G the number of 1 and –1 are $2n/3+1$, and the number of –1 is $n/3$, for r – the number of 1 and –1 is $n/3$, for message the number of 0, 1 and –1 is not less than t.

q – the module in calculating the coefficients of the polynomial in $(Z/qz)[X](x^n–x–1)$. A prime number that satisfies the condition $q≥48t+3$.

Further, as parameters (n, q, t) are selected parameters from [2].

Exceptions:

1. According to the requirement of no decoding errors, parameters q, t must satisfy the requirement $q≥48t+3$. According to [2] these parameters satisfied the requirement $q≥48t+3$. Due to the change in requirement, some values of the q parameter have been changed. The parameters that have been changed are shown in the Table 2. The first column defines the parameter numbers in the Table B Parameters [2]. Numbering from 0.

Table 2

Values of changed parameters

№	It was			It has become		
	n	Q	T	n	q	t
5	463	6481	135	463	6529	135
29	587	5233	109	587	5237	109
87	823	4513	94	823	4519	94
97	881	3217	67	881	3221	67

2. For some parameters from [2], the values of $3t$ are approximately equal to n. The use of such parameters during encryption leads to the need for repeated execution of the encryption operation with various random data that is supplemented the message. This is due to the fact that the probability of obtaining a polynomial in which the number of 1, –1, 0 does not satisfy the requirement to remain at least t is quite large. Repeated execution of the encryption operation greatly increases the time of its execution.

In Table 3 are the values of parameters that are *not recommended* for use on this basis. The first column defines the parameter numbers in the Table B Parameters [2]. Numbering from 0.

Table 3

Values of parameters that are not recommended for use

№	*N*	*Q*	*t*	3*t*	*n*–3*t*
2	461	7607	153	459	2
3	461	8779	153	459	2
7	463	9371	154	462	1
11	491	8627	163	489	2
12	491	9277	163	489	2
13	499	8243	166	498	2
14	499	9029	166	498	2
16	503	8663	167	501	2
24	557	9323	185	555	2
32	599	9551	198	594	4

2.2 Additional parameters

qBits – number of bits for *q*, $qBits = \lceil \log_2 q \rceil$. It is used by functions of transforming a polynomial in $(Z/qz)[X](x^n–x–1)$ to the byte string, if the same number of bytes is used for each element. If you use a method of minimizing the key length, the parameter is not used.

db – the number of bits for the random component. It is calculated by the formula:

$$db = \begin{cases} 128 & k < 128 \\ 192 & k < 192 \\ 256 & k < 256 \end{cases}$$

For $k>200$ $db=256$.

maxMsgLenBytes – determines the message maximum length. It is determined by the polynomial with small coefficients length. In encoding 2 polynomial coefficients are replaced by the bit string of 3 bits long. Thus, the total length of the bit string is $(n–1)/2*3/8$ bytes. This string should contain a random component in the length of *db*/8 bytes, the message length is 1 byte and the message itself. That is, the message maximum length is calculated by the formula

$$maxMsgLenBytes=(n–1)/2*3/8–db/8–1;$$

bc, *bk* are constants for calculating the module by the Barret method;

c – the number of bits used to determine the polynomial non-zero element index (see IGF2 [1]). It is determined by *n*:

$$c = \begin{cases} 9 & n < 512 \\ 10 & n < 1024 \\ 11 & n \ge 1024 \end{cases}$$

dm0 – determines the number of non-zero elements in the encoded polynomial *dm0=t*;

Hlen – hash length

$$Hlen = \begin{cases} 160 & k \le 112 \\ 256 & k > 112 \end{cases}$$

minCallsMask – determines the number of the hash function calls for the algorithm (MGF_TP-1). It is determined by the formula:

$$minCallsMask=(16*n+HashLenBits*5–1)/(HashLenBits*5)+1;$$

minCallsR – determines the number of the hash function calls for the algorithm (IGF-2). It is determined by the formula: $minCallsR=(t*4*c+HashLenBits)/HashLenBits$;

pkLen – the number of public key bits that are used to form the string for encryption; *pkLen=db*

OID – 3 *байта*, *OID*[0]=0; *OID*[1]=1; *OID*[2]=2;

3. Key generation

3.1 Key data

Private key consists of:
Polynomial G in $(Z/3z)[X](x^n–x–1)$, the number of non-zero elements is equal to $2n/3+1$;
Polynomial F in $(Z/3z)[X](x^n–x–1)$, the number of non-zero elements is equal to $2t/3$;
Polynomial $f=3F+1$ in $(Z/qz)[X](x^n–x–1)$;
Polynomial f^{-1} in $(Z/qz)[X](x^n–x–1)$;
Public key – polynomial h in $(Z/qz)[X](x^n–x–1)$.

3.2 Polynomial with a given number of nonzero elements generation algorithm

Small polynomial Generation (*SmallPolynomialGeneration*)
Component The parameters *n*
Input Not zero items count *count*, random numbers generator *rand*
Output the polynomial *dest* of degree *n*–1
The Small polynomial Generation function shall be computed by the following or an equivalent sequence of steps;
Set dest := 0
Set i := 0
While i <count do
a. Set ind := rand() % n
b. If dest [ind] = 0
i. Set value:=rand()%2
ii. If value == 0
1. value = -1
iii. dest [ind] =value
c. Set i:=i + 1
Output dest

3.3 Key generation algorithm

Algorithm 1 Random key generation primitive
Component The parameters *n, t, q*
Input Small polynomial Generation Function (*Small polynomial Generation*)
Output Polynomial *F*, Polynomial *h*.
The Random key generation shall be computed by the following or an equivalent sequence of steps;
1 Set count := 2n/3+1
2 Call SmallPolynomialGeneration(count) for computer the polynomial G
3 Set count := 2t
4 Call SmallPolynomialGeneration(count) for computer the polynomial F
5 Compute the polynomial f=3F+1 in $(Z/qz)[X](x^n–x–1)$
6 Compute the polynomial f^{-1}= such as $f^{-1}*f=f*f^{-1}=1$ in $(Z/qz)[X](x^n–x–1)$. If f^{-1} not exist go to step 4
7 Compute the polynomial $h = 3gf^{-1}$ in $(Z/qz)[X](x^n–x–1)$
Output F, h

The inversion calculation is performed using the extended Euclidean algorithm. When calculating the inversion for the standard NTRU [1], the value q (module for polynomial coefficients) was 2048. This made it possible to first calculate the inverse by the module 2, and then use it to calculate the inversion by modules 4, 16, 256, 65536, and then go to module 2048. This made it possible to significantly reduce computing costs compared with the use of the extended Euclidean algorithm [6]. Unfortunately, in NTRUPrime mathematics, q is a large prime number, so the authors did not find a way to gradually calculate the inversion.

To reduce the computational complexity of the algorithm, the following techniques were used:

AVX operations were used to perform all operations on polynomials;

The Barrett method was used to calculate the module;

Only positive coefficients were used in the calculation, the transition to the value in the range $[-q/2, q/2]$ was performed after the final calculation of the inversion.

For further optimization, you can simultaneously form Small Polynomials F, G, but their formation time is no more than 5% relative to the inversion calculation function, so this optimization is not used.

Time characteristics of the key generating function for some NTRUPrime parameters are given in Table 4. The first and 5 columns define the parameter numbers in the Table B Parameters [2]. Numbering from 0. Parameter № 64 (n=739, q=9829) is given for comparison with the data given in [2]. Parameter №74 (n=761, q=4591) is chosen for comparison with the data given in [3].

Table 4

Time characteristics of the key generating function for some NTRUPrime parameters

№	*N*	*Q*	KeyGenerations (tacts)	№	*N*	*Q*	KeyGenerations (tacts)
0	439	6833	17542928	9	479	6089	17317560
1	457	6037	16663192	10	491	6287	17758164
4	467	3911	14472788	15	503	2879	14384316
5	463	6529	17583180	17	523	3331	15206436
6	463	6841	17779208	64	739	9829	29850996
8	479	5689	16777688	74	761	4591	24910804

The maximum time for parameter generation is 38976232, the minimum – 14472788 processor tacts for the last parameter and parameter 4 from the parameter table [2].

4. Algorithms for converting polynomials into an array of bytes and vice versa

4.1 Converting a polynomial $(Z/3z)[X](x^n-x-1)$ into an array of bytes (package)

The algorithm is used to encode a private key.

The coding table for the polynomial coefficients is given (Table 5).

Table 5

Coding table

Code	Coefficients	**Code**	Coefficients	**Code**	Coefficients
00000	-1, -1, -1	**01001**	-1, -1, 0	**10010**	-1, -1, 1
00001	0, -1, -1	**01010**	0, -1, 0	**10011**	0, -1, 1
00010	1, -1, -1	**01011**	1, -1, 0	**10100**	1, -1, 1
00011	-1, 0, -1	**01100**	-1, 0, 0	**10101**	-1, 0, 1
00100	0, 0, -1	**01101**	0, 0, 0	**10110**	0, 0, 1
00101	1, 0, -1	**01110**	1, 0, 0	**10111**	1, 0, 1
00110	-1, 1, -1	**01111**	-1, 1, 0	**11000**	-1, 1, 1
00111	0. 1, -1	**10000**	0. 1, 0	**11001**	0. 1, 1
01000	1, 1, -1	**10001**	1, 1, 0	**11010**	1, 1, 1

To assign 3 polynomial coefficients, it is enough to use 5 bits. Table 5 specifies codes for all variations of the coefficients. The row number of the table specifies the code (5 bits), which encodes 3 consecutive polynomial coefficients, starting with the coefficient with a smaller number.

4.2 Converting a polynomial $(Z/qz)[X](x^n–x–1)$ into an array of bytes and vice versa

The algorithm is used for public keys and encryption results.

The objective is to allocate for the public key and the encryption result minimum of memory to allow storage its on a device with a small memory.

If in the standard q=2048 for all parameters, then for the new algorithm the value q>=48t+3 and the prime number (2t is the number of polynomial non-zero coefficients).

2 methods of packing q.

1 method. Calculate the minimum $Q=2^k>q$. In packing for each polynomial element take k bits.

Advantage – simple packaging-unpacking operation.

Disadvantages:

the public key size can be reduced;

the k value is different for different q, that is, the packaging-unpacking procedures depend on q.

2 method. Set a polynomial as a large number in the q system. That is, to calculate the polynomial value:

$$h_0+h_1*q+h_2*q^2+\ldots+h_{n-1}*q^{n-1}.$$

Advantages:

public key takes a minimum of memory;

packing (calculating the polynomial value – the Gorner scheme)–unpacking (the definition of the number "digits" in a given numbers system) procedures do not depend on q.

q – this is the algorithm parameter, that is, the values q^i for i=2…n–1 can be calculated once, this will greatly accelerate both the packing algorithm and the unpacking algorithm.

Disadvantage: We must use arithmetic of long numbers.

Unpacking – the operation is inversely related to the selected packaging option.

Table 6 shows the values of the public key lengths when using the first and second methods, as well as the time (the number of processor tacts) for key unpacking for some parameters of NTRU Prime. The first column defines the parameter number in the Table B Parameters [2]. Numbering from 0. Parameter № 64 (n=739, q=9829) is given for comparison with the data given in [2]. Parameter №74 (n=761, q=4591) is chosen for comparison with the data given in [3].

Table 6

Public Key. The dependence of the length and time for the unpacking operation depending on the method

№			Lengths (Bits)			Time (tacts)			
	N	*Q*	Len1	Len2	Δ (%)	Pack1	Unpack1	Pack2	Unpack2
0	439	6833	5707	5593	2	3100	3154	1685683	8263155
1	457	6037	5941	5740	4	1356	1380	750852	3706680
4	467	3911	5604	5573	1	1055	1076	701441	3506980
5	463	6529	6019	5868	3	1273	1310	727928	3679286
6	463	6841	6019	5899	2	1270	1288	732257	3692521
8	479	5689	6227	5976	4	1297	1315	767563	3899634
9	479	6089	6227	6022	3	1300	1318	772980	3902594
10	491	6287	6383	6196	3	1358	1376	812741	4107416
15	503	2879	6036	5781	4	1122	1140	780414	3946185
17	523	3331	6036	5886	3	1119	1137	793815	3998625
64	739	9829	10346	9802	6	1902	1920	1880312	9626583
74	761	4591	9893	9258	7	2044	2063	1832085	9494444

Conclusions:

1. The length for the second method is less than the length for the first method, not more than 8 %.

2. The packing-unpacking time for option 1 is much less than the time of the corresponding operation for the second option. The transformation time for the second option is even greater than the time for encryption and decryption operations.

Developers recommendation: use the first option.

5. Polynomials multiplication operation

The multiplication operation is performed during encryption (one operation) and decryption (2 operations). It is this operation that takes most time among the remaining operations, so its optimization is paying much attention.

All multiplication operations are performed for polynomials $(Z/3z)[X](x^n–x–1)$ and $(Z/qz)[X](x^n–x–1)$ with each other. As a result, we obtain a polynomial $(Z/qz)[X](x^n–x–1)$. The multiplication on the polynomial $f=3F+1$ (decryption operation) can easily be replaced by the multiplication operation by F. Really, $f*h=(3F+1)*h=3*F*h+h$. After calculating $F*h$ multiplication by 3 and adding h are performed very quickly due to the use of AVX operations. As our researches have shown, it is more efficient than calculating directly $f*h$. Therefore, the polynomial multiplication operation with $(Z/3z)[X](x^n–x–1)$ on the polynomial $(Z/qz)[X](x^n–x–1)$ is considered below.

To optimize the multiplication operation, the following was investigated:

—method of specifying the polynomial in $(Z/3z)[X](x^n–x–1)$,

—different multiplication algorithms.

5.1. Method of specifying the polynomial in $(Z/3z)[X](x^n–x–1)$

The polynomial after unpacking has coefficients 0, 1, –1. To specify, you can use an array in which to specify all coefficients, or numbers arrays that have values 1 and –1. When performing a multiplication operation, the most significant coefficients are significant, so the second method of assignment is more accepted.

5.2. Different multiplication algorithms

We have investigated all multiplication methods that are recommended in [2, 3] and other methods. When implementing various methods, minimization of transition operations was performed and the properties of the modern processors cache were taken into account. The possibilities of using parallel computations through the use of AVX operations and multi-core processors were explored.

The multiplication result must be reduced by modulus q (prime number) and by modulus of polynomial $x^n–x–1$. As our studies have shown for the reduction by modulus q to use the Barrett reduction method is most effectively [5].

For reduced by modulus $x^n–x–1$, polynomial $x^n–x–1$ properties are used.

The following is a summary of the various multiplication and calculating modules methods.

5.2.1. Experimental research of multiplication algorithms

The following methods were studied:

- A1 – "School" method – it is provided for comparison and verification of the results correctness;
- A2 – Toom-Cook's algorithm. It is implemented according to the [2, 3] recommendations. Even without interpolation, time characteristics are worse than the rest of the algorithms;
- A3 – FFT, Fast Fourier transform (algorithm with pre-calculations). Time characteristics approximately coincide with the time characteristics for the Toom-Cook's algorithm;
- A4 – Our algorithm that takes into account the special structure of a polynomial with coefficients (0, –1, 1) for which numbers are given, using AVX commands;
- A5 – Algorithm A4, which uses 2 threads.

– A6 – Algorithm A4, which uses 4 threads.

The results of the experimental research for some parameters from NTRU Prime are shown in Table 7. The first column defines the parameter number in the Table B Parameters [2]. Numbering from 0. Parameter № 64 (n=739, q=9829) is given for comparison with the data given in [2]. Parameter №74 (n=761, q=4591) is chosen for comparison with the data given in [3].

Table 7

Polynomials multiplying algorithms. Time indicators

№	*N*	A1	A2	A3	A4	A5	A6
0	439	403796	152191	143679	40179	26588	18436
1	457	451344	157148	147812	37692	26336	18492
4	467	471340	157100	148196	24852	17636	14556
5	463	459120	157064	148352	46188	27444	20008
6	463	471340	157100	148196	24852	17636	14556
8	479	494224	157144	148020	36664	22468	18096
9	479	494156	157204	148228	39408	26168	18236
10	491	519168	157096	148556	41048	25784	19084
15	503	545388	157168	148120	19464	16376	13952
17	523	545436	157192	148056	22608	18176	14704
64	**739**	**1162612**	**157136**	**148944**	**89676**	**52344**	**32400**
74	**761**	**1229540**	**157204**	**148852**	**38560**	**27264**	**20148**

Conclusions on the multiplication methods:

1. The use of complex algorithms that do not take into account the special structure of the polynomial with coefficients (–1, 0, 1) makes no sense.

2. The polynomial with coefficients (–1, 0, 1) is better to be specified using non-zero elements indices.

3. The use of threads in the case of module performance on the multi-core processor makes a sense.

4. According to [2] for the parameters (p=739, q=9829) the number of tacts for the multiplication operation is 51488, we have 32400.

5. According to [3] for the parameters (p=761, q=4591) the number of tacts for the multiplication operation is 28682, we have 20148.

After completing the multiplication operation, we obtain the polynomial, whose coefficients do not exceed $q*n$, and the degree of the polynomial is $2n–1$. This polynomial must first be reduced by modulus $x^n–x–1$, after that we obtain the polynomial of degree n, and then each of n coefficients are reduced by modulus q.

5.2.2. Reduced by modulus $x^n–x–1$

For reduction by modulus $x^n–x–1$ it is sufficient to polynomial coefficients with indices $0…n–1$ to add (subtract) corresponding coefficients with indices $n...2n–2$.

To simultaneously reduction the coefficients block using AVX operations.

5.2.3. Reduced by modulus q. Barrett reduction

To accelerate the method, the constants, which depend only on the values of n, q, are computed one time when setting parameters.

Barrett bk, bc constants pre-calculation:

bk is chosen such that $2^{bk}>p*q$

$bc=2^{bk}/q$ is calculated.

For each polynomial coefficient h_i, it must perform the following calculations:

$$h_i=h_i–((h_i*bc)>>bk)*q.$$

To simultaneously reduction by modulus q the coefficients block using AVX operations.

6. Encryption and decryption

6.1. Encryption

The encryption algorithm coincides with Algorithm 23 (ANSI X 9.98 [1]). Next, the notation is used from Algorithm 23.

6.1.1. Encryption algorithm optimization

1. For strings *M* and *sData*, intersecting memory is used. This allows you to reduce the amount of memory required by the message length and reduce the time it takes to copy the string for encryption (Step 5 and Step 9).

2. To accelerate the formation of *Mtrin* (Step 8), the bit string is processed in portions of 3 bytes, which allows you to immediately get 8 polynomial coefficients. The case is handled correctly when the length of the string is not multiple 3.

3. To exclude the need to convert a public key into a byte string (Step 9), it is stored in the container in the form of a byte string and in the form of a polynomial.

4. For blinding polynomial generation (Step 10) a BPGM method is used, which optimization will be described below.

5. To calculate *r*h* (Step 11), the multiplication function is used for one or multi-core processor according to the execution environment. The maximum number of cores that the function uses is 4.

6. Step 12 and Step 13 are executed as one step.

7. The polynomial generation to mask *mask* (Step 14, MGF).

8. Steps 15–18 are executed as one step in which the modules are formed and the numbers of values –1, 0, 1 are calculated, their correctness is checked and the ciphertext value is calculated.

9. The result is the bytes array that we obtain by converting a polynomial into bytes array according to the package algorithm.

6.1.2. Optimization of the blinding polynomial calculation algorithm

Blinding polynomial calculations are performed according to the Blinding Polynomial Generation Method (BPGM) – Algorithm 18 [1].

The algorithm uses the index generation function (IGF) by which the bit string (IGF state *s*) creates, with the length *minCallsR*Hlen*.

6.1.3. Formation of IGF state *s*

We consider 3 options for creating bit string.

For option 1, use of the hash function as proposed in the standard [1]. To optimize the implementation of the *s* creation function, a constant part is formed that is used at each step of the hash calculation. When forming a hash, AVX commands are used.

For option 2, when implementing the *s* creation function, the initial string and its length are defined as for option 1, but instead of multiple recalling of the hash function, the multiple encryption function call for the SALSA-2.0 algorithm is used. For the next step of encrypting, the result for the previous step is selected. The number of steps compared to the Algorithm for option 1 is reduced due to the fact that the length of the initial state is longer than the length of the hash.

For option 3, instead of the hash function, the encryption function for the SNOW-2.0 algorithm is used (see option 2).

6.1.4. Calculation of coefficients

To optimize the calculation of the polynomial coefficients in the first step, a completely array of coefficients is formed. At the second step, the possibility of their application is checked. If necessary, the bit string is expanded.

6.1.5. MGF algorithm optimization

The algorithm uses the *minCallMask* parameter to generate a bit string. The bit string formation optimization is made due to the fact that the constant part is used at each step of the hash calculation is calculated only once. When forming a hash, AVX commands are used. For fast conversion of a byte string in a polynomial, a pre-computed table is used, the entry point of which is byte and each row includes 5 coefficients.

6.1.6. Parallel computing using

In the encryption algorithm, 2 branches are executed in parallel.

The first branch includes Steps 4-8 of the encryption algorithm (Algorithm 23 [1]).

The second branch includes Steps 9-10 (Algorithm 23 [1]).

To implement parallel branches, Open MP is used.

6.2. Decryption

The decryption algorithm is executed according to Algorithm 23 (ANSI X 9.98 [1]). Next, the notation is used from Algorithm 23 [1].

6.2.1. Decryption algorithm optimization

1. Steps 1-3 are combined in one step. The multiplication algorithm is used to multiply polynomials. The number of non-zero elements of received polynomial is counted simultaneously with its formation.

2. Steps 4-6 are combined in one step, that allows to form in one cycle a byte row to calculate *c0R4*.

3. Step 7 optimization.

4. To optimize Step 8, the cycle is deployed to simultaneously receive 4 bytes of string.

5. Steps 9, 10, 11 are combined in one step. That allowed in one cycle to form a byte string.

6. Step 12 of the algorithm actually determines the message after the decryption. Further, the speculative execution of the code that uses these data may be continued. The following steps can be performed in parallel with the use of the obtained data. If in result of additional checks will be obtained a negative result, the code execution after the use of decrypted data should be determined to be invalid. In case of successful additional verification, the executed code is accepted as valid. Additional checks include Steps 13-17.

7. For Step 13, you do not need to convert the public key into a byte string, it is stored in this format.

8. Step 14 optimization to form a blinding polynomial.

9. Step 15 polynomials multiplying.

6.3. Time characteristics of encryption and decryption functions

The results of the experimental research for some parameters from NTRU Prime are shown in Table 8. The first column defines the parameter number in the Table B Parameters [2]. Numbering from 0. Parameter № 64 (n=739, q=9829) is given for comparison with the data given in [2]. Parameter №74 (n=761, q=4591) is chosen for comparison with the data given in [3].

For encryption and decryption operations, there are 3 modes for creating a random string to create a blinding polynomial (hash, salsa2.0, snow2.0). For both modes, we get the best results for the last option.

The last 2 columns determine the time it takes to decrypt and verify the decryption correctness. In the case of parallel execution of the verification operation with other encryption-decryption operations, you can balance the time required for encryption and decryption.

The last row in the table specifies the results obtained for the parameters specified in [3]. These parameters correspond to a cryptographic stability of more than 200.

Authors [3] received the results:
Encryption: 59600 and decryption 97452 cycles respectively.

Table 8

Time characteristics of encryption and decryption functions

№	N	Q	Encrypt			Decrypt			Decrypt	
			Hash	Salsa2.0	Snow2.0	Hash	Salsa2.0	Snow2.0	Decrypt	Check
0	439	6833	56364	40456	35408	81356	64280	60016	29048	28508
1	457	6037	62712	38200	33712	87948	61448	56868	30500	28496
4	467	3911	52340	32016	29252	72760	51528	48532	26908	23320
5	463	6529	64336	39124	35288	90484	63256	59072	32352	29220
6	463	6841	66240	40628	36268	94884	66448	61464	32336	30884
8	479	5689	62800	38396	34144	87748	61556	57320	31000	28360
9	479	6089	64444	39072	34796	90624	62596	58352	32168	28948
10	491	6287	67464	41304	36188	94220	65984	61500	31368	30600
15	503	2879	50460	30536	27844	87748	61556	47116	31000	28360
17	523	3331	54192	33112	30812	90624	62596	51224	32168	28948
64	739	9829	104916	61597	53876	145516	100152	93224	48636	46488
74	761	4591	73456	44120	40032	100140	71480	67728	37580	30920

In the case of the hash using, our encryption operation implementation loses the specified ones in the paper by 25%, and the decryption operation – 5%. When using encryption algorithms, our implementation wins 30% for encryption algorithm and 29% for decryption algorithm.

In [4] – one of the algorithms submitted to the competition (Kyber), the following performance data after optimization using AVX2 are given:

Encryption – 119652;

Decryption – 125736.

Compared to our results for the best option, we get the winning: 65% for encryption and 44% for decryption.

Conclusions

In view of the above, the following conclusions can be made.

1. In the cryptosystem «NTRU Prime IIT Ukraine» as the main cryptographic transformation, as in NTRU Prime, unlike NTRU, the transformation is used in the finite field. The above makes it impossible to conduct a series of potential attacks regarding the cryptographic system «NTRU Prime IIT Ukraine» and eliminates the potential weaknesses present in the NTRU cryptosystem. They are mainly related to the existence of non-trivial subfields or factor rings of the factor (truncated) polynomials ring.

2. In the cryptosystem «NTRU Prime IIT Ukraine» polynomials F and r are arbitrary t-small, they have $2t$ non-zero coefficients (+1, –1), whereas in NTRU, each of these polynomials has exactly t nonzero coefficients equal to 1 and –1 respectively. The same is true for the polynomial g used in the cryptosystem «NTRU Prime IIT Ukraine», which is an arbitrary small polynomial with $2t$ nonzero coefficients (+1, –1). Specified allows to expand the size of the key space in comparison with NTRU without losing the efficiency of algorithms implementation for the keys formation and implementation of encryption and decryption algorithms.

3. During optimization great attention was paid to multiplication operation, as it is the most time consuming. Usage of complex multiplying algorithms, which don't take into account special polynomial structure with coefficients (–1, 0, 1) doesn't make a sense. The polynomial with coefficients (–1, 0, 1) is better to be specified using non-zero elements indices. The use of threads in case of reduction by modulus on the multi-core processor makes a sense. Usage of AVX2 operations for reduction by modulus polynomial $x^n - x - 1$ and prime q and for Barrett algorithm for reduction by modulus q are effective and accelerates multiplication speed.

4. Three algorithms of blinding polynomial formation were studied (hash, salsa2.0, snow2.0), the best time rates were obtained for Snow 2.0.

References:

1. American National Standard for Financial Services Lattice-Based Polynomial Public Key Establishment Algorithm for the Financial Services Industry ANSI X9.98–2010, 2010. 284 p.

2. Daniel J. Bernstein, Chitchanok Chuengsatiansup, Tanja Lange, Christine van Vredendaal: NTRU Prime. Access mode: https://ntruprime.cr.yp.to/ntruprime-20160511.pdf.

3. Daniel J. Bernstein, Chitchanok Chuengsatiansup, Tanja Lange, Christine van Vredendaal NTRU Prime: reducing attack surface at low cost. Access mode: https://eprint.iacr.org/2016/461.pdf.

4. Joppe Bos, Leo Ducas, Eike Kiltz, Tancrede Lepoint, Vadim Lyubashevsky, John M. Schanck CRYSTALS Kyber: a CCA-secure module-lattice-based KEM. Access mode: https://eprint.iacr.org/2017/634.pdf.

5. P. Barrett Communications, Authentication and Security Using Public Key Encryption A Design for Implementation. Master's thesis, Oxford University, September 1984.

6. Kachko E. G. Investigation of inversion calculation methods in the NTRU algorithm / E. G. Kachko, D. S. Balagura, K. A. Pogrebniak, Yu. I. Gorbenko // Radiotekhnika. 2012. Issue 171. P. 58–63. (in Russian).

7. Post-Quantum Cryptography. [Electronic resource] Access mode: https://csrc.nist.gov/projects/post-quantum-cryptography.

8. Kachko O. G. The optimization of NTRU-like algorithm for asymmetric encryption with "inconvenient parameters" / O. G. Kachko, L. V Makutonina, O. S. Akolzina // Mathematical and computer modeling. Series: Engineering, 15 (2017), 79–85. (in Ukrainian).

9. Kachko O. Asymmetric encryption algorithm optimization based on using NTRU Prime mathematics / O. Kachko, Yu. Gorbenko, M. Yesina, O. Akolzina // Radiotekhnika. 2017. Issue 191. P. 5-10.

Акціонерне товариство
«Інститут інформаційних технологій»,
Харківський національний
університет радіоелектроніки,
Харківський національний
університет імені В.Н.Каразіна

Надійшла до редколегії 11.02.2018

УДК 004.056.55

О.О. КУЗНЕЦОВ, д-р техн. наук, І.Д. ГОРБЕНКО, д-р техн. наук,
Ю.І. ГОРБЕНКО, канд. техн. наук, А.М. ОЛЕКСІЙЧУК, д-р техн. наук, В.А. ТИМЧЕНКО

МАТЕМАТИЧНА СТРУКТУРА ПОТОКОВОГО ШИФРУ *СТРУМОК*

Вступ

Важливим механізмом криптографічного захисту інформації є потокове симетричне шифрування [1, 2]. Воно застосовується для забезпечення послуги конфіденційності та цілісності (як додаткової послуги) інформації під час обробки інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах [1].

Останніми роками вимоги до сучасних алгоритмів потокового шифрування істотно зросли [3 – 5]: з одного боку, потрібно забезпечувати високу швидкість криптографічного перетворення (понад 10 Гбит/с), з іншого – необхідно ефективно протистояти новітнім методам криптографічного аналізу, в тому числі із застосуванням квантових методів обчислення.

Отже розробка, дослідження та поступове впровадження нових методів потокового симетричного шифрування є актуальною та надзвичайно важливою науково-прикладною проблемою національного рівня.

Зазвичай операція потокового шифрування є побітовою операцією XOR між ключовим потоком і повідомленням. У ISO/IEC 18033-4:2011 описано вихідні функції для різних потокових шифрів, та певні генератори псевдовипадкових чисел, які призначено для захисту інформації з обмеженим доступом, зокрема, для забезпечення конфіденційності інформації під час її обробки [6]. Метою цієї роботи є викладення основних результатів з розробки нового генератору псевдовипадкових чисел (ключового потоку), який позначено «Струмок», та який пропонується у якості кандидата на національний стандарт симетричного шифрування в Україні [7 – 18]. Генератор «Струмок» забезпечує високу швидкість формування ключового потоку (понад 10 Гбіт/с), яка перевищує більшість відомих алгоритмів, та придатний до застосування в постквантовому середовищі [7 – 9].

Загальні параметри потокового шифру

В основі алгоритму *Струмок* лежить класична схема підсумовуючого генератора [1, 2, 6, 7], подібна генератору *SNOW-2.0*, який визначено в ISO/IEC 18033-4:2011 [6]. Алгоритм *Струмок* використовує 256-бітний вектор ініціалізації *IV* та 256-бітний або 512-бітний секретний ключ *K* і забезпечує високий та надвисокий рівень стійкості із врахуванням можливого застосування квантового криптографічного аналізу. Криптоалгоритм орієнтований на 64-розрядні обчислювальні системи, отже розмір слова визначено рівним 64 бітам.

Основними структурними компонентами генератору є регістр зсуву з лінійним зворотнім зв'язком (linear feedback shift register, *LFSR*) та скінченний автомат (finite-state machine, *FSM*), в якому виконується нелінійне перетворення. Вхідні дані використовуються для ініціалізації змінної стану $S_i (i \geq 0)$, яка складається з вісімнадцяти 64-бітових блоків, до складу яких входить дві компоненти:

– 16 змінних $s^{(i)}$ – комірок регістра зсуву з лінійним зворотнім зв'язком: $s^{(i)} = (s_{15}^{(i)}, s_{14}^{(i)}, ..., s_0^{(i)})$;

– два регістри скінченного автомату $r^{(i)}$: $r^{(i)} = (r_2^{(i)}, r_1^{(i)})$.

На виході отримуємо ключовий потік (гаму шифру), який формується з 64-бітових слів Z_i. Схематичне зображення функціонування генератора ключових потоків *Струмок* в довільний момент часу *i* наведено на рис. 1. Змінну часової залежності *i* не наведено.

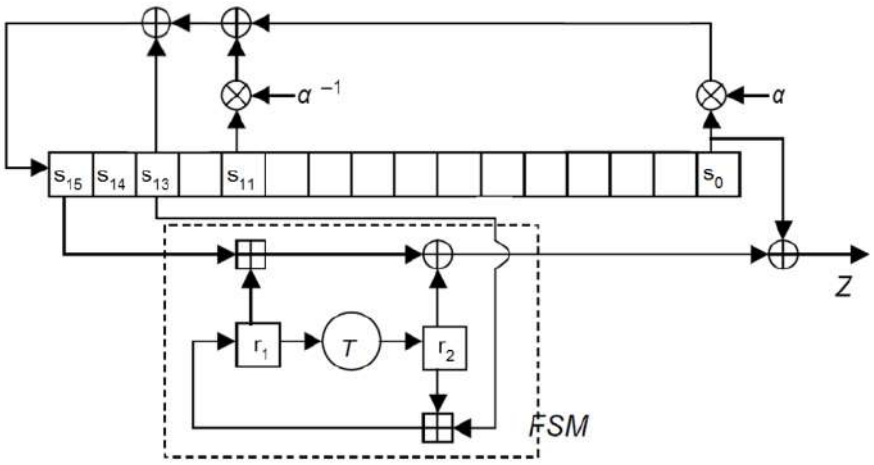


Рис. 1. Схематичне зображення генератору ключових потоків *Струмок* у режимі генерації гами шифру (ключового потоку)

Відводи зворотного зв'язку у LFSR будується за примітивним над полем $GF(2^{64})$ поліномом

$$f(x) = x^{16} + x^{13} + \alpha^{-1}x^{11} + \alpha,$$

де α є коренем примітивного над полем $GF(2^8)$ поліному

$$g(z) = z^8 + \beta^{170}z^7 + \beta^{166}z^6 + \beta^2 z^5 + \beta^{224}z^4 + \beta^{70}z^3 + \beta^2.$$

В свою чергу поле $GF(2^8)$ будується за примітивним над полем $GF(2)$ поліномом

$$p(y) = y^8 + y^4 + y^3 + y^2 + 1,$$

а коефіцієнти поліному $g(z)$ подаються через ступінь примітивного елементу β поля $GF(2^8)$, тобто β – корінь поліному $p(y)$.

Таким чином, маємо вежу полів:

$$GF(2) \subset GF(2^8) \subset GF(2^{64}) \subset GF(2^{1024}),$$

де

– поле $GF(2^{1024})$ задається відводами зворотного зв'язку LFSR як факторкільце $GF(2^{64})[x]/(f(x))$,

– поле $GF(2^{64})$ задається як факторкільце $GF(2^8)[z]/(g(z))$,

– поле $GF(2^8)$ задається як факторкільце $GF(2)[y]/(p(y))$.

Отже період вихідної послідовності *LFSR* є максимальним і дорівнює $2^{1024} - 1$.

Структурно в алгоритмі *Струмок* можна виділити три основні функції:

– функція ініціалізації *Init*, яка приймає в якості вхідних даних ключ *K* (256 біт або 512 біта) і вектор ініціалізації *IV* (256 біт), і виробляє початкове значення змінної стану

$$S_0 = (s^{(0)}, r^{(0)});$$

– функція наступного стану *Next*, яка приймає на вхід змінну стану

$$S_i = (s^{(i)}, r^{(i)})$$

і виробляє наступне значення змінної стану

$$S_{i+1} = (s^{(i+1)}, r^{(i+1)}).$$

Функція *Next* може виконуватися в двох режимах, в залежності від способу виконання ітерації – як частини реалізації або як частини нормального режиму генерації вихідних даних;

– функція ключового потоку *Strm*, що приймає на вході змінну стану $S_i = (s^{(i)}, r^{(i)})$ і виробляє на виході 64-бітний ключовий потік Z_i.

Функція ініціалізації внутрішнього стану *Init*

Функція ініціалізації внутрішнього стану *Init* описується наступним чином.

Вхід: 256 або 512-бітний ключ *K*, 256-бітний вектор ініціалізації *IV*.

Вихід: початкове значення змінної стану $S_0 = (s^{(0)}, r^{(0)})$.

Ключ для версії потокового шифру *Струмок-256* можна представити у вигляді чотирьох 64-бітних слів

$$K = \left(K_3, K_2, K_1, K_0\right)$$

а для 512-бітного ключа – у вигляді восьми 64-бітних слів

$$K = \left(K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0\right),$$

де K_3 та K_7, відповідно для 256 и 512 біт, найбільш значущі слова, а K_0 – найменш значущі.

Вектор ініціалізації можна представити у вигляді чотирьох 64-бітних слів

$$IV = \left(IV_3, IV_2, IV_1, IV_0\right),$$

де IV_3 – найбільш значуще слово, а IV_0 – найменш значуще.

1. В 16 комірок LFSR заноситься значення ключа.

Для версії з 256-бітним ключем виконуються операції:

$$s_{15}^{(-33)} = \neg K_0,\ s_{14}^{(-33)} = K_1,\ s_{13}^{(-33)} = \neg K_2,\ s_{12}^{(-33)} = K_3,\ s_{11}^{(-33)} = K_0,\ s_{10}^{(-33)} = \neg K_1,\ s_9^{(-33)} = K_2,\ s_8^{(-33)} = K_3,$$
$$s_7^{(-33)} = \neg K_0,\ s_6^{(-33)} = \neg K_1,\ s_5^{(-33)} = K_2 \oplus IV_3,\ s_4^{(-33)} = K_3,\ s_3^{(-33)} = K_0 \oplus IV_2,\ s_2^{(-33)} = K_1 \oplus IV_1,$$
$$s_1^{(-33)} = K_2,\ s_0^{(-33)} = K_3 \oplus IV_0.$$

Для версії з 512-бітним ключем *K* виконуються операції:

$$s_{15}^{(-33)} = K_0,\ s_{14}^{(-33)} = \neg K_1,\ s_{13}^{(-33)} = K_2,\ s_{12}^{(-33)} = K_3,\ s_{11}^{(-33)} = \neg K_7,\ s_{10}^{(-33)} = K_5,\ s_9^{(-33)} = \neg K_6,$$
$$s_8^{(-33)} = K_4 \oplus IV_3,\ s_7^{(-33)} = \neg K_0,\ s_6^{(-33)} = K_1,\ s_5^{(-33)} = K_2 \oplus IV_2,\ s_4^{(-33)} = K_3,\ s_3^{(-33)} = K_4 \oplus IV_1,$$
$$s_2^{(-33)} = K_5,\ s_1^{(-33)} = K_6,\ s_0^{(-33)} = K_7 \oplus IV_0.$$

2. Виконуються 32 ініціюючих такти без генерації ключового потоку, тобто чотири повних циклів. Формально це подається наступним чином:

$$S_{-1} = Next^{32}(S_{-33}, INIT),$$

що означає 32 ітерації з виконання функції *Next* у режимі ініціалізації *INIT*, $S_{-33} = (s^{(-33)}, r^{(-33)})$ – обраховані на попередньому кроці значення змінної стану.

3. Розраховується початкове значення змінної стану $S_0 = (s^{(0)}, r^{(0)})$ за правилом: $S_0 = Next(S_{-1})$, тобто шляхом виконання функції *Next* у звичайному режимі.

4. Виводиться вихідне значення $S_0 = (s^{(0)}, r^{(0)})$.

Функція наступного стану *Next*

Функція стану *Next* описується наступним чином.

Вхід: Змінна стану $S_i = (s^{(i)}, r^{(i)})$, обраний режим (звичайний, або режим ініціалізації).

Вихід: Наступне значення змінної стану $S_{i+1} = (s^{(i+1)}, r^{(i+1)})$.

1. Виконується нелінійна підстановка для оновлення значення регістру $r_2^{(i+1)}$ скінченного автомату. Для цього розраховується значення функції T: $r_2^{(i+1)} = T(r_1^{(i)})$.

2. Оновлюється значення регістру $r_1^{(i+1)}$ скінченного автомату. Для цього розраховується значення

$$r_1^{(i+1)} = r_2^{(i+1)} +_{64} s_{13}^{(i)},$$

де $+_{64}$ позначає операцію додавання цілих чисел за модулем 2^{64} (у схемі шифру на рис. 1 цю операцію позначено як ⊞).

3. Оновлюється значення 15 комірок *LFSR*

$$s_j^{(i+1)} = s_{j+1}^{(i)}$$

для всіх $j = 0, 1, \ldots, 14$.

4. Оновлюється значення 16-ї комірки *LFSR*. Якщо встановлено звичайний режим функції *Next*, значення цієї комірки обчислюється за правилом

$$s_{15}^{(i+1)} = (s_0^{(i)} \otimes \alpha) \oplus (s_{11}^{(i)} \otimes \alpha^{-1}) \oplus s_{13}^{(i)}.$$

Якщо встановлено режим ініціалізації *INIT* функції *Next*, значення обчислюється за правилом

$$s_{15}^{(i+1)} = FSM(s_{15}^{(i)}, r_1^{(i)}, r_2^{(i)}) \oplus (s_0^{(i)} \otimes \alpha) \oplus (s_{11}^{(i)} \otimes \alpha^{-1}) \oplus s_{13}^{(i)}.$$

Операції множення $\otimes$ на α та на α^{-1} та сутність функції FSM пояснюються далі.

5. Обчислюється та виводиться значення змінної стану $S_i = (s^{(i)}, r^{(i)})$.

Схематичне зображення генератору ключових потоків *Струмок* при виконанні функції *Next* у режимі ініціалізації представлено на рис. 2.

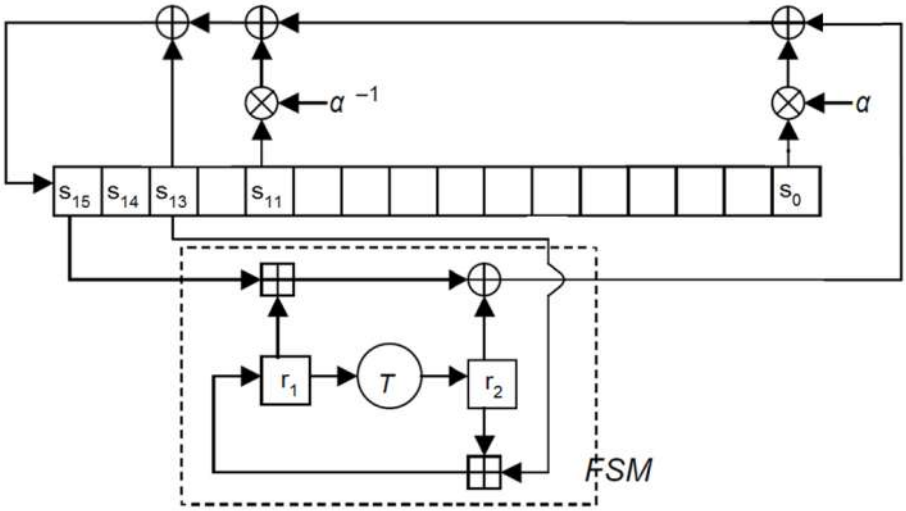


Рис. 2. Схематичне зображення генератору ключових потоків Струмок у режимі ініціалізації функції *Next*

Функція ключового потоку *Strm*

Функція ключового потоку *Strm* описується наступним чином.

Вхід: Змінна стану $S_i = (s^{(i)}, r^{(i)})$.

Вихід: 64-бітовий ключовий потік Z_i.

1. Обчислюється значення

$$Z_i = FSM(s_{15}^{(i)}, r_1^{(i)}, r_2^{(i)}) \oplus s_0^{(i)}.$$

2. Виводиться вихідне значення Z_i.

Функція скінченного автомату *FSM*

Функція скінченного автомату позначається як *FSM* (*x*, *y*, *z*) та описується наступним чином.

Вхід: три 64-бітових рядка *x*, *y* і *z*.

Вихід: 64-бітовий рядок *q*.

1. Обчислюється значення $q = (x +_{64} y) \oplus z$.

2. Виводиться вихідне значення *q*.

Функція нелінійної підстановки *T*

Функція нелінійної підстановки T реалізує перестановку елементів скінченного поля $GF(2^{64})$ за допомогою компонентів національного стандарту блокового симетричного криптоперетворення ДСТУ 7624:2014 [19].

Вхід: 64-бітовий рядок w.

Вихід: 64-бітовий рядок $T = T(w)$.

1. Вхідний 64-бітовий рядок w розбивається на підблоки w_j по 8 біт:

$$w = (w_7, w_6, w_5, w_4, w_3, w_2, w_1, w_0),$$

2. Для кожного підблоку w_j виконується підстановка з алгоритму ДСТУ 7624:2014 за допомогою чотирьох табличних перетворень $\pi_0, \pi_1, \pi_2, \pi_3$ (див. додаток Б). Виконання функції T за допомогою цих перетворень схематично (на прикладі підблоків w_j, що подано у шіснадцятковому вигляді) зображено на рис. 3.

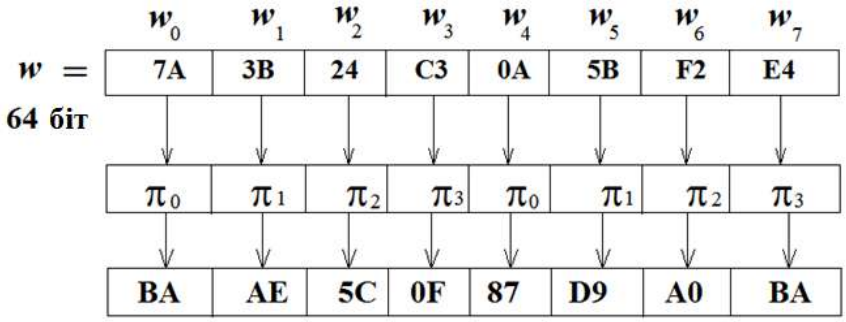


Рис. 3. Схематичне зображення виконання процедури підстановки $T = T(w)$

У результаті формується вихідний вектор $r = (r_7, r_6, r_5, r_4, r_3, r_2, r_1, r_0)$:

$$r_j = \pi_{j \bmod 4}\left[w_j\right],$$

де $j = 0, 1, \ldots, 7$.

3. Обчислюється вектор

$$q = (q_7, q_6, q_5, q_4, q_3, q_2, q_1, q_0)$$

за правилом

$$\begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \\ q_7 \end{pmatrix} = \begin{pmatrix} 01 & 01 & 05 & 01 & 08 & 06 & 07 & 04 \\ 04 & 01 & 01 & 05 & 01 & 08 & 06 & 07 \\ 07 & 04 & 01 & 01 & 05 & 01 & 08 & 06 \\ 06 & 07 & 04 & 01 & 01 & 05 & 01 & 08 \\ 08 & 06 & 07 & 04 & 01 & 01 & 05 & 01 \\ 01 & 08 & 06 & 07 & 04 & 01 & 01 & 05 \\ 05 & 01 & 08 & 06 & 07 & 04 & 01 & 01 \\ 01 & 05 & 01 & 08 & 06 & 07 & 04 & 01 \end{pmatrix} \cdot \begin{pmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \\ r_6 \\ r_7 \end{pmatrix},$$

де елементи матриці (подано у шістнадцятковому вигляді) та векторів r і q інтерпретуються як елементи скінченного поля $GF(2^8)$, яке задане як факторкільце $GF(2)[y]/(p(y))$.

Цю операцію можна записати у скороченому вигляді (як у ДСТУ 7624:2014):

$$q_i = (\upsilon >>> i) r^T,$$

де $\upsilon = (01, 01, 05, 01, 08, 06, 07, 04)$, $>>> i$ – операція циклічного зсуву на i розрядів праворуч, $i = 0, 1, ..., 7$,

$$r^T = \left(r_0, r_1, r_2, r_3, r_4, r_5, r_6, r_7\right)^T.$$

4. Виводиться вихідне значення q, яке інтерпретується як 64-бітовий рядок.

Швидке обчислення вектору $\left(q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7\right) = Q$ реалізується за правилом

$$Q^T = T_0[w_0] \oplus T_1[w_1] \oplus T_2[w_2] \oplus T_3[w_3] \oplus T_4[w_4] \oplus T_5[w_5] \oplus T_6[w_6] \oplus T_7[w_7],$$

де

$$T_0[a] = \begin{pmatrix} 01 \\ 04 \\ 07 \\ 06 \\ 08 \\ 01 \\ 05 \\ 01 \end{pmatrix} \cdot \pi_0\left[a\right],\ T_1[a] = \begin{pmatrix} 01 \\ 01 \\ 04 \\ 07 \\ 06 \\ 08 \\ 01 \\ 05 \end{pmatrix} \cdot \pi_1\left[a\right],\ T_2[a] = \begin{pmatrix} 05 \\ 01 \\ 01 \\ 04 \\ 07 \\ 06 \\ 08 \\ 01 \end{pmatrix} \cdot \pi_2\left[a\right],\ T_3[a] = \begin{pmatrix} 01 \\ 05 \\ 01 \\ 01 \\ 04 \\ 07 \\ 06 \\ 08 \end{pmatrix} \cdot \pi_3\left[a\right],$$

$$T_4[a] = \begin{pmatrix} 08 \\ 01 \\ 05 \\ 01 \\ 01 \\ 04 \\ 07 \\ 06 \end{pmatrix} \cdot \pi_0\left[a\right],\ T_5[a] = \begin{pmatrix} 06 \\ 08 \\ 01 \\ 05 \\ 01 \\ 01 \\ 04 \\ 07 \end{pmatrix} \cdot \pi_1\left[a\right],\ T_6[a] = \begin{pmatrix} 07 \\ 06 \\ 08 \\ 01 \\ 05 \\ 01 \\ 01 \\ 04 \end{pmatrix} \cdot \pi_2\left[a\right],\ T_7[a] = \begin{pmatrix} 04 \\ 07 \\ 06 \\ 08 \\ 01 \\ 05 \\ 01 \\ 01 \end{pmatrix} \cdot \pi_3\left[a\right].$$

Застосування таблиць-констант $T_i[a]$, $i = 0,1,...,7$ дозволяє значно зменшити кількість операцій, зокрема функція нелінійної підстановки обчислюється за сім операцій XOR над 64-бітовими рядками.

Множення на α в арифметиці поля $GF(2^{64})$

Множення на α в арифметиці поля $GF(2^{64})$ реалізується за допомогою таблиці передобчислень Mul_α з 256 рядків по 64 бітів в кожному.

Вхід: 64-бітовий рядок w, що представляє елемент поля $GF(2^{64})$.

Вихід: 64-бітовий рядок $w' = w \otimes \alpha$, що представляє елемент поля $GF(2^{64})$.

1. Обчислюється значення

$$w' = (w << 8) \oplus Mul_\alpha[w >> 56], \qquad (1)$$

де $w << 8$ є результатом зсуву ліворуч (в бік старших розрядів) 64-бітового рядка w на вісім розрядів із заповненням молодших розрядів нульовими значеннями;

$w >> 56$ є результатом зсуву праворуч (в бік молодших розрядів) 64-бітового рядка w на 56 розрядів із заповненням старших розрядів нульовими значеннями. Вісім молодших розрядів вектору $w >> 56$ інтерпретуються як елемент поля $GF(2^8)$ для індексації таблиці передобчислень Mul_α;

Mul_α – таблиця-константа з 256 рядків по 64 біта в кожному (таблиця передобчислень);

$Mul_\alpha[c]$ – 64-бітне значення таблиці передобчислень у рядку з індексом c, де c представляє елемент поля $GF(2^8)$, $Mul_\alpha[c]$ представляє елемент поля $GF(2^{64})$.

2. Виводиться вихідне значення w'.

Множення на α^{-1} в арифметиці поля $GF(2^{64})$

Множення на α^{-1} в арифметиці поля $GF(2^{64})$ реалізується за допомогою таблиці передобчислень $Mul_{\alpha^{-1}}$ з 256 рядків по 64 бітів в кожному.

Вхід: 64-бітовий рядок w, що представляє елемент поля $GF(2^{64})$.

Вихід: 64-бітовий рядок $w' = w \otimes \alpha^{-1}$, що представляє елемент поля $GF(2^{64})$.

1. Обчислюється значення

$$w' = (w >> 8) \oplus Mul_{\alpha^{-1}}[w \,\&\, \gamma], \tag{2}$$

де $w >> 8$ є результатом зсуву праворуч (в бік молодших розрядів) 64-бітового рядка w на 8 розрядів із заповненням старших розрядів нульовими значеннями;

$w \,\&\, \gamma$ є результатом побітової кон'юнкції 64-бітового рядка w та 64-бітового рядка γ, який у шістнадцятковому поданні має вигляд $\gamma = 00000000000000FF$. Вісім молодших розрядів вектору $w \,\&\, \gamma$ інтерпретуються як елемент поля $GF(2^8)$ для індексації таблиці передобчислень $Mul_{\alpha^{-1}}$;

2. Виводиться вихідне значення w'.

Значення таблиць-констант Mul_α, $Mul_{\alpha^{-1}}$

Для швидкого шифрування застосовуються таблиці передобчислень Mul_α, $Mul_{\alpha^{-1}}$. Це дозволяє значно зменшити кількість операцій для обробки блоку вхідних даних.

Поліном, що задає зворотний зв'язок LFSR має вигляд $f(x) = x^{16} + x^{13} + \alpha^{-1}x^{11} + \alpha$, де α і α^{-1} належать полю $GF(2^{64})$, причому $\alpha = z$ є коренем примітивного над полем $GF(2^8)$ поліному $g(z)$. Таким чином, у кожній комірці LFSR зберігається 64-бітна послідовність w, яку представимо у вигляді восьми підблоків w_j по вісім біт у кожному:

$$w = (w_7, w_6, w_5, w_4, w_3, w_2, w_1, w_0),$$

які інтерпретуються як коефіцієнти поліному

$$w(z) \in GF(2^8)[z] / (g(z)).$$

Якщо $\alpha = z$ є коренем примітивного над $GF(2^8)$ поліному $g(z) = z^8 + g_7 z^7 + ... + g_1 z + g_0$, тоді маємо:

$$\begin{aligned} w(z) \cdot \alpha &= (w_7 z^8 + w_6 z^7 + ... + w_1 z^2 + w_0 z) \bmod g(z) \equiv \\ &\equiv (w_6 + w_7 g_7) z^7 + (w_5 + w_7 g_6) z^6 + ... + (w_0 + w_7 g_1) z + (w_7 g_0) z^0 = \\ &= w_{<<8}(z) + w_{>>56}(z) \cdot g'(z), \end{aligned}$$

де поліноми $w_{<<8}(z)$, $w_{>>56}(z)$ та $g'(z)$ мають вигляд:

$$w_{<<8}(z) = w_6 z^7 + w_5 z^6 + ... + w_0 z,$$

$$w_{>>56}(z) = w_7,$$

$$g'(z) = g_7 z^7 + g_6 z^6 + ... + g_1 z + g_0.$$

Двійкове подання коефіцієнтів поліномів $w_{<<8}(z)$ і $w_{>>56}(z)$ утворює розглянуті у (1) двійкові послідовності $w << 8$ і $w >> 56$. Отже, обчислення $w(z) \cdot \alpha$ в арифметиці поля

$GF(2^{64})$ відповідає формулі (1), де 256 значень таблиці $Mul_{\alpha}[w_7]$ розраховуються як 64-бітні послідовності при двійковому поданні коефіцієнтів $(w_7g_7, w_7g_6, ..., w_7g_1, w_7g_0)$ поліному

$$w_{>>56}(z)\cdot g'(z) = w_7(g_7z^7 + g_6z^6 + ... + g_1z + g_0)$$

для кожного з 256 можливих значень $w_7 \in GF(2^8)$.

Якщо $\alpha = z$ є коренем примітивного над $GF(2^8)$ поліному $g(z) = z^8 + g_7z^7 + ... + g_1z + g_0$, тоді маємо:

$$\alpha^8 = g_7\alpha^7 + ... + g_1\alpha + g_0\alpha^0,$$

або

$$\alpha^7 = g_7\alpha^6 + ... + g_1\alpha^0 + g_0\alpha^{-1},$$

отже

$$g_0^{-1}\alpha^7 + g_0^{-1}g_7\alpha^6 + ... + g_0^{-1}g_1\alpha^0 = \alpha^{-1} = z^{-1}.$$

Тоді

$$w(z)\alpha^{-1} = w_7z^6 + w_6z^5 + ... + w_1z^0 + w_0z^{-1} =$$
$$w_7z^6 + w_6z^5 + ... + w_1z^0 + w_0(g_0^{-1}z^7 + g_0^{-1}g_7z^6 + ... + g_0^{-1}g_1z^0) =$$
$$= (w_0g_0^{-1})z^7 + (w_7 + w_0g_0^{-1}g_7)z^6 + ... + (w_1 + w_0g_0^{-1}g_1)z^0 =$$
$$= w_{>>8}(z) + w_0(z)\cdot g''(z),$$

де поліноми $w_{>>8}(z)$, $w_0(z)$ та $g''(z)$ мають вигляд:

$$w_{>>8}(z) = w_7z^6 + w_6z^5 + ... + w_2z + w_1,$$
$$w_0(z) = w_0,$$
$$g''(z) = g_0^{-1}z^7 + g_0^{-1}g_7z^6 + ... + g_0^{-1}g_2z + g_0^{-1}g_1.$$

Двійкове подання коефіцієнтів поліномів $w_{>>8}(z)$ і $w_0(z)$ утворює розглянуті у (2) двійкові послідовності $w>>8$ і γ. Таким чином, обчислення $w(z)\cdot\alpha^{-1}$ в арифметиці поля $GF(2^{64})$ відповідає формулі (2), де 256 значень таблиці $Mul_{\alpha^{-1}}[w_0]$ розраховуються як 64-бітні послідовності при двійковому поданні коефіцієнтів $(w_0g_0^{-1}, w_0g_0^{-1}g_7, ..., w_0g_0^{-1}g_2, w_0g_0^{-1}g_1)$ поліному

$$w_0(z)\cdot g''(z) = w_0(g_0^{-1}z^7 + g_0^{-1}g_7z^6 + ... + g_0^{-1}g_2z + g_0^{-1}g_1)$$

для кожного з 256 можливих значень $w_0 \in GF(2^8)$.

Дослідження швидкості програмної реалізації

Для дослідження швидкості формування ключового потоку ми реалізували в рівних умовах найбільш відомі симетричні криптоперетворення. Список алгоритмів, джерело специфікації та короткі відомості наведено у табл. 1.

Таблиця 1

Криптоалгоритми, обрані для порівняння

Шифр	Джерело специфікації	Розмір стану, біт	Розмір ключа, біт	Розмір IV, біт
AES	FIPS-197, CRYPTREC, ISO/IEC 18033-4	128	128, 256	256
Калина	ДСТУ 7624:2014	128, 256, 512	128, 256, 512	128, 256, 512
HC	eSTREAM	128, 256	128, 256	128, 256

MICKEY	eSTREAM	160	128	128
RABBIT	ISO/IEC 18033-4, eSTREAM	513	128	64
SALSA-20	eSTREAM	512	128	64
SNOW2.0	ISO/IEC 18033-4	512	128, 256	128, 56
SOSEMANUK	eSTREAM	512	128	128
TRIVIUM	eSTREAM, ISO/IEC 29192-3	288	80	80
Enocoro	ISO/IEC 29192-3	272	80, 128	64
CRYPTMT3	eSTREAM	128	128	64
DECIMv2	ISO/IEC 18033-4, eSTREAM	288	128	128
RC4	Список розсилки Cypherpunks	256	256	–
KCIPHER-2	ISO/IEC 18033-4, CRYPTREC	640	128	128
GRAIN	eSTREAM	128	128	96
MUGI	ISO/IEC 18033-4	128	128	128
Струмок-256	Цей документ	1024	256	256
Струмок-512		1024	512	512

Результати тестування за критерієм шифрування довгих потоків [20] наведено в табл. 2.

Таблиця 2

Результати оцінки швидкодії шифрів

Шифр	Intel Core i7-6820HQ 2.7Gh	Intel Core i7-5500u 2.4Gh	Intel Pentium P6200 2.13Gh
AES-128	2,48	1,75	1,12
AES-256	1,66	1,18	0,80
Калина-128	2,56	1,79	0,83
Калина-256	1,71	1,21	0,57
Калина-512	1,42	0,99	0,46
HC-128	11,46	7,69	4,25
HC-256	5,13	3,88	2,03
MICKEY-128	0,07	0,05	0,03
RABBIT	3,65	2,77	1,64
SALSA-20	3,02	2,06	1,41
SNOW2.0-128	8,76	5,43	3,67
SNOW2.0-256	8,72	5,54	3,59
SOSEMANUK	4,07	2,56	1,82
TRIVIUM	3,89	2,78	1,89
CryptMT3	5,92	4,63	4,04
DECIM-128	0,01	0,01	0,01
RC4	3,58	3,21	1,67
KCIPHER-2	0,40	0,40	0,31
GRAIN	0,01	0,01	0,00
MUGI	3,62	2,98	2,58
Струмок-256	13,31	10,04	5,10
Струмок-512	13,70	9,74	5,08

Як видно із даних таблиці, генератор ключових потоків *Струмок* дозволяє формувати псевдовипадкові послідовності із швидкістю понад 10 Гбіт/с. За цим показником він випереджає майже всі найбільш поширені шифри, зокрема і алгоритм *SNOW2.0*.

Висновки

Генератор ключового потоку *Струмок* у своїй концептуальній схемі подібний до SNOW2.0. Але розробники SNOW2.0 зосереджувалися на використанні 32-розрядних обчислювальних систем, тоді як *Струмок* призначений для використання в більш потужних 64-розрядних обчислювальних системах. У зв'язку з цим в алгоритмі *Струмок* підвищується швидкість формування псевдовипадкової послідовності, що використовується 64-розрядними словами, для зберігання потоку ключів шифрування. Проведені порівняльні тести показали, що алгоритм *Струмок* на 32-розрядних обчислювальних системах також демонструє хороші результати роботи. Використання попереднього обчислення збільшує швидкість алгоритму, оскільки в процесі генерації ключові потоку немає необхідності в складних калькуляціях.

В алгоритмі *Струмок* збільшені, в порівнянні з *SNOW2.0*, довжини секретного ключа та вектору ініціалізації. Це дозволяє надійно застосовувати потоковий шифр навіть з із врахуванням квантових методів криптографічного аналізу. Отже за сукупністю властивостей *Струмок* може розглядатися у якості кандидата на національний стандарт симетричного шифрування в Україні.

Список литературы:

1. Ferguson N. and Schneier B. Practical Cryptography. John Wiley & Sons. 2003. 432 p.
2. Menezes A.J., P.C. van Oorschot, Vanstone S.A. Handbook of Applied Cryptography. CRC Press, 1997, 794 p.
3. Koblitz N. and Menezes A.J. A Riddle Wrapped in an Enigma. Internet: https://eprint.iacr.org/2015/1018.pdf, Oct. 20, 2015 [Aug. 21, 2016].
4. Bernstein D., Buchmann J. and Dahmen E.. Post-Quantum Cryptography. Springer-Verlag, Berlin-Heidleberg, 2009, 245 p.
5. Moody D. Post-Quntum Cryptography: NIST's Plan for the Future // The Seventh International Conference on Post-Quntum Cryptography, Japan, 2016. [On-line]. Internet: https://pqcrypto2016.jp/data/pqc2016_nist_announcement.pdf [March 8, 2016].
6. ISO/IEC 18033-4:2011. Information technology – Security techniques – Encryption algorithms – Part 4: Stream ciphers. On-line]. Internet: http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=54532 [Dec., 2012].
7. Kuznetsov O., M. Lutsenko and D. Ivanenko, "Strumok stream cipher: Specification and basic properties," 2016 Third International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T), Kharkiv, 2016, pp. 59-62.
8. Kuznetsov O., Gorbenko Y. and Kolovanova I. Combinatorial properties of block symmetric ciphers key schedule, 2016 Third International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T), Kharkiv, 2016, pp. 55-58.
9. Gorbenko I., Kuznetsov A., Lutsenko M. and Ivanenko D. The research of modern stream ciphers // 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T), Kharkov, 2017, pp. 207-210.
10. Kuznetsov A., Svatovskij I., Kiyan N. and Pushkar'ov A. Code-based public-key cryptosystems for the post-quantum period // 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T), Kharkov, 2017, pp. 125-130.
11. A. Kuznetsov, I. Kolovanova and T. Kuznetsova. Periodic characteristics of output feedback encryption mode // 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T), Kharkov, 2017. pp. 193-198.
12. Kuznetsov O., Gorbenko Y., Andrushkevych A. and Belozersev I. Analysis of block symmetric algorithms from international standard of lightweight cryptography ISO/IEC 29192-2. 2017 // 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T). Kharkov, 2017. pp. 203-206.
13. Izbenko Y., Kovtun V. and Kuznetsov A. The Design of Boolean Functions by Modified Hill Climbing Method // Sixth International Conference on Information Technology: New Generations, Las Vegas, NV, 2009, pp. 356-361.
14. Kuznetsov A., Serhiienko R. and Prokopovych-Tkachenko D. Construction of cascade codes in the frequency domain // 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T). Kharkov, 2017. pp. 131-136.
15. Andrushkevych A., Kuznetsova T., Bilozertsev I. and Bohucharskyi S. The block symmetric ciphers in the post-quantum period // Third International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T), Kharkiv, 2016, pp. 43-46.

16. Gorbenko I.D., Dolgov V.I., Rublinetskii V.I., Korovkin K.V. Methods of Information Protection in Communications Systems and Methods of Their Cryptoanalysis // Telecommunications and Radio Engineering. 1998. Volume 52, Issue 4, pp. 89-96.

17. Gorbenko I., Ponomar V. Examining a possibility to use and the benefits of post-quantum algorithms dependent on the conditions of their application // EasternEuropean Journal of Enterprise Technologies. Vol 2, No 9 (86) (2017), pp. 21-32.

18. Stasev Yu.V., Kuznetsov A.A. Asymmetric code-theoretical schemes constructed with the use of algebraic geometric codes // Kibernetika i Sistemnyi Analiz, No. 3, pp. 47-57, May-June 2005.

19. ДСТУ 7624:2014. Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення. [On-line]. Internet: http://shop.uas.org.ua/ua/informacijni-tehnologii-kriptografichnij-zahist-informacii-algoritm-simetrichnogo-blokovogo-peretvorennja.html

20. eSTREAM Optimized Code HOWTO. [On-line]. Internet: http://www.ecrypt.eu.org/stream/perf/ [Nov. 1, 2005].

Харківський національний
університет імені В.Н. Каразіна

Надійшла до редколегії 12.02.2018

УДК 621.391:519.2

А.М. ОЛЕКСІЙЧУК, д-р техн. наук, С.М. ІГНАТЕНКО

АЛГОРИТМИ ОЦІНЮВАННЯ СТІЙКОСТІ SNOW 2.0-ПОДІБНИХ ПОТОКОВИХ ШИФРІВ НАД КІЛЬЦЯМИ ЛИШКІВ ВІДНОСНО КОРЕЛЯЦІЙНИХ АТАК

Вступ

Потоковий шифр SNOW 2.0 [1] запропонований у 2002 році як альтернатива попередньої (більш слабкої) версії – SNOW. На сьогодні цей шифр є стандартизованим [2] та являє собою один з найбільш швидких програмно орієнтованих потокових шифрів.

Найбільш потужними з відомих атак на SNOW 2.0 є кореляційні атаки, сутність яких полягає у складанні та розв'язанні певних систем лінійних рівнянь зі спотвореними правими частинами [3 – 6].

Метою даної статті є відповідь на запитання про те, чи можна підвищити стійкість SNOW 2.0 відносно відомих кореляційних атак шляхом (повної) заміни у схемі генератора гами цього шифру порозрядного булевого додавання арифметичним додаванням за модулем 2^{32}, а також заміни нелінійної підстановки у схемі генератора іншим (швидким) перетворенням.

В п. 1 наведено означення класу SNOW 2.0-подібних потокових шифрів над кільцем лишків за модулем 2^N та описану загальну схему побудови кореляційних атак на них, аналогічних відомим атакам на SNOW 2.0. В п. 2 – 4 на основі аналізу відомих методів розв'язання систем лінійних рівнянь зі спотвореними правими частинами над кільцем лишків за модулем 2^N [7 – 12] наведено алгоритми оцінювання обчислювальної складності зазначених кореляційних атак на шифри, що розглядаються, а п. 5 – відповідні чисельні оцінки їх стійкості відносно цих атак. Наприкінці статті сформульовано стислі висновки.

В цілому отримані результати свідчать про помітну перевагу, з погляду стійкості відносно відомих кореляційних атак, SNOW 2.0-подібних шифрів над кільцями лишків у порівнянні з традиційними SNOW 2.0-подібними шифрами. Поряд з тим, питання про стійкість розглянутих у цій статті шифрів відносно інших (зокрема, алгебраїчних) атак потребує окремого дослідження.

1. Кореляційні атаки на SNOW 2.0-подібні шифри над кільцями лишків за модулем 2^N

Розглянемо генератор гами SNOW 2.0-подібного потокового шифру, який складається з лінійного регістру зсуву (ЛРЗ) над кільцем $R_N = \mathbf{Z}/(2^N)$ та підстановки $\sigma: R_N \to R_N$, поєднаних між собою, як зазначено на рис. 1. Вважатимемо, що многочлен зворотного зв'язку ЛРЗ $g(z) = z^n - (c_{n-1}z^{n-1} + \ldots + c_0)$ над кільцем R_N є многочленом максимального періоду (який дорівнює $2^{N-1}(2^n - 1)$ [13]), а ЛРЗ виробляє лінійну рекурентну послідовність $x_0, x_1, \ldots$, знаки якої пов'язані співвідношенням $x_{i+n} = c_{n-1}x_{i+n-1} + \ldots + c_0 x_i$, $i = 0, 1, \ldots$. Генератор гами являє собою скінченний автономний автомат з множиною внутрішніх станів $R_N{}^n \times R_N{}^2$, функцією переходів

$$h((z_{n-1}, z_{n-2}, \ldots, z_0), u, v) = ((z_n, z_{n-1}, \ldots, z_1), z_\mu + v, \sigma(u)),$$

та функцією виходів

$$f((z_{n-1}, z_{n-2}, \ldots, z_0), u, v) = z_0 + z_{n-1} + u + v,$$

де $z_0, \ldots, z_{n-1}, u, v \in R_N$, $x_n = c_{n-1}x_{n-1} + \ldots + c_0 x_0$. Отже, знак гами в i-му такті визначається за початковим станом $((x_{n-1}, x_{n-2}, \ldots, x_0), u_0, v_0)$ генератора за допомогою таких рекурентних співвідношень:

$$\gamma_i = x_i + x_{i+n-1} + u_i + v_i\,,\ u_{i+1} = x_{i+\mu} + v_i\,,\ v_{i+1} = \sigma(u_i)\,,\ i = 0, 1, \ldots. \qquad (1)$$

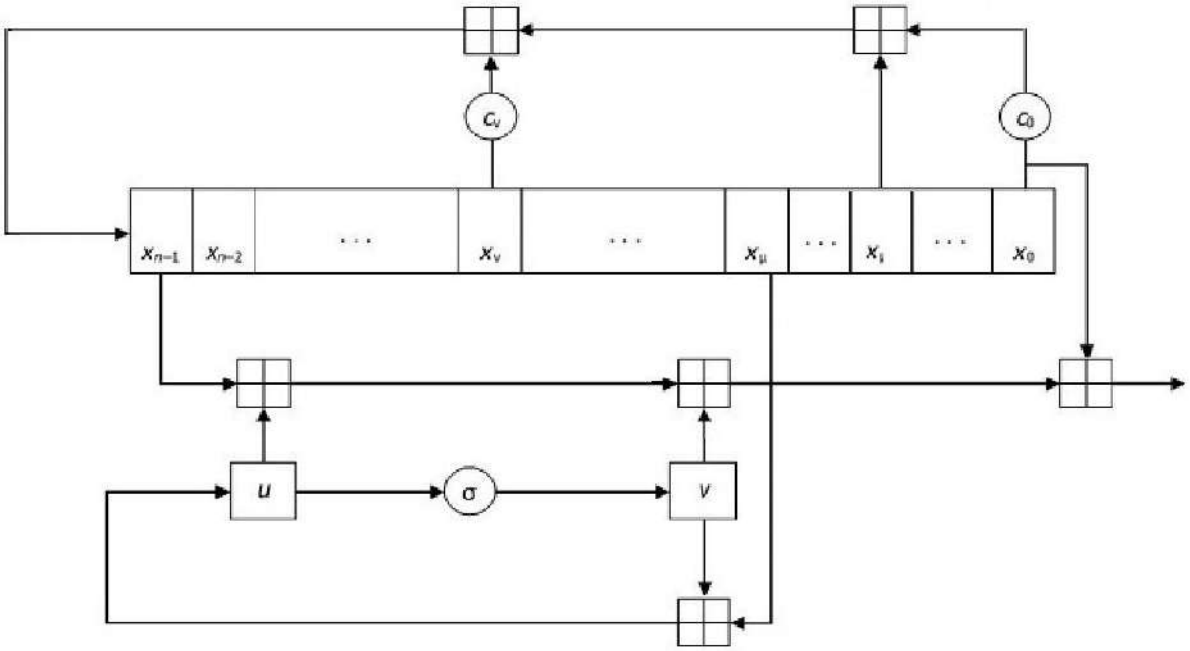


Рис. 1. Схема генератора гами SNOW 2.0-подібного шифру

Зауважимо, що головною відмінністю генератора, що розглядається, від генераторів гами звичайних SNOW 2.0-подібних потокових шифрів [14] є застосування операції + додавання в кільці R_N замість операції $\oplus$ порозрядного додавання двійкових векторів за модулем 2.

Відомі на сьогодні кореляційні атаки на шифр SNOW 2.0 [3 – 6] базуються на тому, що сума (за модулем 2) знаків гами в будь-яких суміжних тактах є результатом спотворення знаку певної лінійної рекуренти з характеристичним многочленом $g(z)$. Для генератора, що розглядається, на підставі співвідношень (1) справедливі такі рівності:

$$\gamma_{i+1} - \gamma_i = x_{i+1} + x_{i+n} - x_i - x_{i+n-1} + x_{i+\mu} + \xi_i\,,\ i = 0, 1, \ldots, \qquad (2)$$

де

$$\xi_i = \sigma(u_i) - u_i\,,\ i = 0, 1, \ldots. \qquad (3)$$

Вважаючи, що змінні $u_0, u_1, \ldots$ є незалежними випадковими величинами з рівномірним розподілом на кільці R_N та виражаючи знаки $x_i, x_{i+1}, x_{i+\mu}, x_{i+n-1}, x_{i+n}$ лінійної рекуренти через початковий стан ЛРЗ на рис. 1, отримаємо систему лінійних рівнянь (2) зі спотвореними правими частинами над кільцем R_N, де спотворення є випадковими величинами (3). Кореляційні атаки, що розглядаються, полягають у розв'язанні цієї системи рівнянь (СР) за допомогою відомих методів [7 – 12].

Позначимо $\sigma'(u) = \sigma(u) - u\,,\ u \in R_N$. Тоді

$$p(z) = \mathbf{P}\{\xi_i = z\} = 2^{-N}\,|\{u \in R_N : \sigma'(u) = z\}|\,,\ z \in R_N\,,\ i = 0, 1, \ldots. \qquad (4)$$

Отже, з погляду спроможності генератора протистояти кореляційним атакам, що базуються на розв'язанні СР (2), найкращим способом вибору підстановки σ є такий, коли розподіл (4) є рівномірним або, що те ж саме, відображення σ' є підстановкою на кільці R_N. Поряд з тим, при $N \geq 2$ таких підстановок σ не існує [15]. Проте існують підстановки σ, для яких розподіл (4) відрізняється від рівномірного розподілу ймовірностей на кільці R_N лише у двох точках:

$$p(0)=0,\ \ p(2^{N-1})=2^{1-N},\ p(z)=2^{-N},\ z\in R_N\setminus\{0,\,2^{N-1}\}. \tag{5}$$

Як приклад, зазначимо підстановку σ, значення якої в точці $z\in R_N$ дорівнює циклічному зсуву двійкового запису числа z в бік старших розрядів [15].

Отже, далі вважатимемо, що підстановка σ вибрана таким чином, що розподіл ймовірностей (4) має вигляд (5). Базуючись на результатах робіт [7 – 12], оцінимо обчислювальну складність розв'язання цієї СР за допомогою відомих методів.

2. Метод максимуму правдоподібності

Запишемо перші m рівнянь системи (2) у вигляді

$$Ax=b, \tag{6}$$

де A є (відомою) $m\times n$-матрицею над кільцем R_N, b є вектором з координатами

$$b_i=A_i a+\xi_i,\ i\in\overline{1,m}, \tag{7}$$

де $A_1,\dots,A_m$ – рядки матриці A, $a=(a_1,\dots,a_n)^{\mathrm{T}}$ – невідомий вектор-стовпець над кільцем R_N, який співпадає з початковим станом ЛРЗ на рис. 1, $\xi_1,\dots,\xi_m$ – незалежні випадкові величини, розподілені за законом (4).

Для будь-якого $x\in {R_N}^n$ позначимо $\varepsilon(x)=b-Ax$. Нагадаємо (див., наприклад, [7, 16]), що розв'язання СР (2) методом максимуму правдоподібності (ММП) полягає в знаходженні "оцінки" a^* вектора a за правилом $\mathbf{P}\{\xi=\varepsilon(a^*)\}=\max\limits_{x\in R^n}\mathbf{P}\{\xi=\varepsilon(x)\}$, де $\xi=(\xi_1,\dots,\xi_m)$. Якщо вектор a є рівномірно розподіленим на множині ${R_N}^n$, то ММП має найменшу середню ймовірність помилки серед усіх методів розв'язання СР (2) (див., наприклад, [16]).

Як випливає з результатів робіт [7, 11], для відновлення вектора a за допомогою ММП з імовірністю не менше $1-\delta$, $\delta\in(0,\,1/2)$, необхідно виконати не менше ніж

$$T=nm_0 2^{Nn}(6N^2-N) \tag{8}$$

двійкових операцій, де

$$m_0=\frac{nN(1-\delta)-h(\delta)}{\Delta(p_\xi)}\ln 2,$$

$$h(\delta)=-\delta\log_2\delta-(1-\delta)\log_2(1-\delta),\ \Delta(p_\xi)=2^{-N}\sum_{z\in R_N}(2^N p(z)-1)^2.$$

3. Послідовний метод

Цей метод запропоновано в [8] і полягає у послідовному відновленні двійкових розрядів координат невідомого вектора a шляхом розв'язання систем лінійних рівнянь зі спотвореними правими частинами, які отримуються з вхідної СР (2) за допомогою канонічних гомоморфізмів кільця R_N в кільця $R_i=\mathbf{Z}/(2^i)$, $i\in\overline{0,N-1}$. Необхідною умовою застосовності методу є відмінність розподілу випадкових величин $\xi_j(\mathrm{mod}2^i)$, $j\in\overline{1,m}$, від рівномірного розподілу ймовірностей на кільці R_i для деякого $i\in\overline{0,N-1}$.

Як випливає з формул (4), (5), для будь-яких $j\in\overline{1,m}$, $z\in R_{N-1}$ справедлива рівність $\mathbf{P}\{\xi_j(\mathrm{mod}2^{N-1})=z\}=2^{-(N-1)}$. Отже, випадкові величини $\xi_j(\mathrm{mod}2^i)$, $j\in\overline{1,m}$, є рівномірно розподіленими на кільці R_i для кожного $i\in\overline{0,N-1}$, і послідовний метод є незастосовним для побудови кореляційних атак на генератор гами, що розглядається.

4. Узагальнений алгоритм BKW та його модифікації

Узагальнений алгоритм BKW [11] є природним узагальненням (на випадок довільного скінченного кільця) одного з найкращих на сьогодні алгоритмів розв'язання систем лінійних рівнянь зі спотвореними правими частинами над полем з двох елементів [17, 18]. Цей алгоритм складається з двох етапів, на першому з яких за вхідною СР (2) над кільцем R_N певним чином будується нова система лінійних рівнянь зі спотвореними правими частинами від $n_1 \le n-3$ змінних. Потім, на другому етапі отримана система рівнянь розв'язується за допомогою ММП.

В [10, 12] запропоновано використовувати замість традиційного ММП його модифікації, які базуються, відповідно, на швидкому перетворенні Фур'є та швидкому перетворенні Ферма деяких допоміжних функцій. Показано, що за певних умов це дозволяє помітно зменшити трудомісткість узагальненого алгоритму BKW.

Для оцінювання складності узагальненого алгоритму та його модифікацій введемо низку додаткових позначень. Для будь-якого $n_1 \in \overline{1, n-3}$ позначимо

$$u = \left\lceil \frac{\log(n-n_1)}{2} \right\rceil,\ v = \left\lceil \frac{2(n-n_1)}{\log(n-n_1)} \right\rceil,\ k = 2^{u-1}. \tag{9}$$

Далі, позначимо $p_\xi^{(k)} = (p^{(k)}(z) : z \in R_N)$ розподіл ймовірностей випадкової величини $\xi_1 + \cdots + \xi_{k/2} - (\xi_{k/2+1} + \cdots + \xi_k)$, де $\xi_1, ..., \xi_k$ є незалежними випадковими величинами, розподіленими за законом (4). Покладемо

$$N_k = \{z \in R_N : p^{(k)}(z) > 0\},\ p_{\max} = \max_{z \in R_N} p^{(k)}(z),\ p_{\min} = \min_{z \in N_k} p^{(k)}(z),$$

$$D(p^{(k)} \| \omega) = \sum_{z \in N_k} p^{(k)}(z) \log(2^N p^{(k)}(z)),\ D(\omega \| p^{(k)}) = -2^{-N} \sum_{z \in N_k} \log(2^N p^{(k)}(z)),$$

$$D_a = \sum_{z \in N_k} p^{(k)}(z) \log^2(2^N p^{(k)}(z)) - D(p^{(k)} \| \omega)^2,\ D = 2^{-N} \sum_{z \in N_k} \log^2(2^N p^{(k)}(z)) - D(\omega \| p^{(k)})^2,$$

$$m_1 = \frac{2n_1 \ln(2^{N+1}\delta^{-1})(\log p_{\max} - \log p_{\min})^2}{(D(p^{(k)} \| \omega) + D(\omega \| p^{(k)}))^2},\ m_2 = \left(\frac{u_\alpha \sqrt{D_a} + u_\beta \sqrt{D_x}}{D(p^{(k)} \| \omega) + D(\omega \| p^{(k)})} \right)^2,$$

де u_α, u_β – квантилі стандартного нормального розподілу, $\alpha, \beta > 0$, $\alpha + (2^{Nn_1} - 1)\beta \le \delta/2$, $\delta \in (0, 1/2)$. Нарешті, покладемо

$$t = \min\{m_1, m_2\},\ l = (u + \left\lceil \ln(2t\delta^{-1}) \right\rceil - 1)q^v, \tag{10}$$

$$m(n_1) = lt. \tag{11}$$

На підставі результатів робіт [9 – 12] трудомісткості узагальненого алгоритму BKW та його модифікацій можна оцінити за допомогою алгоритмів 1, 2, 3.

Алгоритм 1 (обчислення трудомісткості узагальненого алгоритму BKW)

Вхідні дані:

– натуральні числа $N \ge 2$, $n \ge 3$;

– число $\delta \in (0, 1/2)$.

Алгоритм обчислень.

Для кожного $n_1 \in \overline{1, n-3}$:

1. Обчислити значення (9), (10), (11).

2. Обчислити $T_{\mathrm{BKW}}(n_1) = 2^{Nn_1+1} n_1 t + ult$.

Результат: число n_1 таке, що $T_{\mathrm{BKW}}(n_1) = \min\{T_{\mathrm{BKW}}(s) : s \in \overline{1, n-3}\}$ та відповідні значення $T_{\mathrm{BKW}}(n_1)$, $m(n_1)$.

Алгоритм 2 (обчислення трудомісткості модифікації узагальненого алгоритму BKW, що базується на швидкому перетворенні Фур'є).

Вхідні дані:

– натуральні числа $N \geq 2$, $n \geq 3$;

– число $\delta \in (0, 1/2)$.

Алгоритм обчислень.

Для кожного $n_1 \in \overline{1, n-3}$:

1. Обчислити значення (9) – (11).
2. Обчислити

$$T'(n_1, t) = \; 5 \cdot 2^{(n_1+2)N+1} N n_1 \log(2^{N+1} N n_1 t) + 2^{2N} t((n_1+1)N(6N-5) + 5(N-1)),$$

$$T'_{\mathrm{BKW}}(n_1) = T'(n_1, t) + ult.$$

Результат: число $n_1{*}$ таке, що $T'_{\mathrm{BKW}}(n_1{*}) = \min\{T'_{\mathrm{BKW}}(n_1) : n_1 \in \overline{1, n-3}\}$ та відповідні значення $T'_{\mathrm{BKW}}(n_1{*})$, $m(n_1{*})$.

Алгоритм 3 (обчислення трудомісткості модифікації узагальненого алгоритму BKW, що базується на швидкому перетворенні Ферма).

Вхідні дані:

– натуральні числа $N \geq 2$, $n \geq 3$;

– число $\delta \in (0, 1/2)$.

Алгоритм обчислень.

Для кожного $n_1 \in \overline{1, n-3}$:

3. Обчислити значення (9), (10), (11).
4. Обчислити

$$T''(n_1, t) = 26 \cdot 2^{N(n_1+1)} N n_1 + 2^{2N} t((n_1+1)N(6N-5) + 5(N-1) + 7 \cdot 2^{N-1} + 2),$$

$$T''_{\mathrm{BKW}}(n_1) = T''(n_1, t) + ult.$$

Результат: число $\tilde{n}_1{*}$ таке, що $T''_{\mathrm{BKW}}(\tilde{n}_1{*}) = \min\{T''_{\mathrm{BKW}}(n_1) : n_1 \in \overline{1, n-3}\}$ та відповідні значення $T''_{\mathrm{BKW}}(\tilde{n}_1{*})$, $m(\tilde{n}_1{*})$.

Зауважимо, що наведені алгоритми можна застосовувати до оцінювання обчислювальної складності розв'язання будь-яких систем лінійних рівнянь зі спотвореними правими частинами над кільцем лишків за модулем 2^N.

5. Оцінки складності кореляційних атак на SNOW 2.0-подібні потокові шифри

В таблиці наведено оцінки обсягу матеріалу, потрібного для розв'язання СР (2) із заданою достовірністю, а також обчислювальної складності розв'язання цієї СР за допомогою ММП, узагальненого алгоритму BKW та його модифікацій.

Символом T в таблиці позначено нижню межу (8) часової складності ММП; символи $T_{\mathrm{BKW}}(n_1)$, $T'_{\mathrm{BKW}}(n_1{*})$ та $T''_{\mathrm{BKW}}(\tilde{n}_1{*})$ позначають трудомісткості узагальненого алгоритму BKW та його модифікацій із застосуванням швидкого перетворення Фур'є та швидкого перетворення Ферма відповідно, а символи $m(n_1)$, $m(n_1{*})$ і $m(\tilde{n}_1{*})$ позначають обсяг матеріалу (кількість рівнянь в системі (2)), потрібного для успішного застосування узагальненого алгоритму BKW та його модифікацій з використанням швидкого перетворення Фур'є та швидкого перетворення Ферма відповідно.

При проведенні розрахунків використано інформацію про розподіл ймовірностей $p_\xi^{(k)}$, що на підставі формули (5) має такий вигляд:

$$p_\xi^{(k)}(0) = 2^{-N}(1+2^{-(N-1)(k-1)}),\ p_\xi^{(k)}(2^{N-1}) = 2^{-N}(1-2^{-(N-1)(k-1)}),\ p_\xi^{(k)}(z) = 2^{-N},$$
$$z \in R_N \setminus \{0, 2^{N-1}\}.$$

Результати оцінювання стійкості SNOW 2.0-подібних шифрів над кільцями лишків відносно кореляційних атак

Параметр	$n=64,\ N=8$	$n=16,\ N=32$
$\log T$	542,01	568,03
n_1	17	7
n_1*	21	7
$\widetilde{n}_1*$	21	7
$\log T_{\text{BKW}}(n_1)$	200,93	329,26
$\log T'_{\text{BKW}}(n_1*)$	199,20	304,65
$\log T''_{\text{BKW}}(\widetilde{n}_1*)$	192,69	300,72
$\log m(n_1)$	199,04	299,72
$\log m(n_1*)$	191,04	299,72
$\log m(\widetilde{n}_1*)$	191,04	299,72

Як видно з таблиці, за умов (4), (5) для розв'язання СР (2) від $n=64$ невідомих над кільцем $R_N = \mathbf{Z}/(2^8)$ за допомогою ММП необхідно не менше ніж $2^{542,01}$ двійкових операцій. При цьому для відновлення будь-яких $n_1=17$ невідомих з цієї системи рівнянь за допомогою узагальненого алгоритму BKW потрібно лише $2^{200,93}$ операцій та $2^{199,04}$ рівнянь, а при застосуванні швидкого перетворення Ферма – тільки $2^{192,69}$ операцій та $2^{191,04}$ рівнянь. Отже, складність найкращої (з відомих на сьогодні) кореляційних атак на SNOW 2.0-подібний шифр, що розглядається, складає $\left\lceil \frac{64}{21} \right\rceil \cdot 2^{192,69} = 2^{194,69}$ операцій при обсязі матеріалу $\left\lceil \frac{64}{21} \right\rceil \cdot 2^{191,04} = 2^{193,04}$ знаків вихідної послідовності генератора.

При $n=16,\ N=32$ (параметри шифру SNOW 2.0) найкраща з відомих кореляційних атак на шифр потребує $\left\lceil \frac{16}{7} \right\rceil \cdot 2^{300,72} = 2^{302,31}$ операцій та $\left\lceil \frac{16}{7} \right\rceil \cdot 2^{299,72} = 2^{301,31}$ знаків гами (при цьому довжина ЛРЗ генератора складає 512 біт). Зауважимо також, що найкраща з відомих кореляційних атак на оригінальний шифр SNOW 2.0 має обчислювальну складність $2^{164,15}$ операцій та потребує $2^{163,59}$ знаків гами [6].

Висновки

Отримані результати свідчать про можливість безпосереднього застосування методів робіт [7 – 12] до вирішення задачі оцінювання стійкості потокових шифрів над кільцями лишків за модулем 2^N відносно кореляційних атак. Вони надають також можливість цілеспрямовано вибирати компоненти зазначених шифрів для підвищення їх стійкості.

Модифікації узагальненого алгоритму BKW, побудовані на основі швидких перетворень Фур'є або Ферма [10, 12], мають меншу часову складність у порівнянні з традиційною версією цього алгоритму [11]. Зокрема, застосування швидкого перетворення Ферма на другому

етапі узагальненого алгоритму BKW зменшує складність кореляційної атаки на розглянуті версії SNOW 2.0-подібних шифрів у $2^{8,24}$ – $2^{28,54}$ разів в залежності від параметрів n та N.

Заміна в схемі генератора гами шифру SNOW 2.0 порозрядного булевого додавання арифметичним додавання за модулем 2^N приводить (за умови належного вибору підстановки σ) до суттєвого підвищення стійкості шифру відносно відомих кореляційних атак. Зокрема, найкраща з таких атак на розглянуту версію шифру потребує $2^{302,31}$ операцій та $2^{301,31}$ знаків гами, в той час як найкраща з відомих атак на SNOW 2.0 [6] має обчислювальну складність $2^{164,15}$ та потребує $2^{163,59}$ знаків гами.

Список літератури:

1. Ekdahl P., Johansson T. A new version of the stream cipher SNOW // Selected Areas in Cryptography SAC 2002. LNCS 2295. Springer-Verlag. P. 47-61.

2. ISO/IEC 18033-4: 2011(E). Information technology Security techniques Encryption algorithm Part 4: Stream ciphers 2011. 92 p.

3. Nyberg K., Wallen J. Improved linear distinguishers for SNOW 2.0 // Fast Software Encryption FSE 2006. LNCS 4047. Springer-Verlag. P. 144-162.

4. Maximov A., Johansson T. Fast computation for large distribution and its cryptographic application // Advanced in Cryptology ASIACRYPT 2005. LNCS 3788. Springer-Verlag. P. 313-332.

5. Lee J.-K., Lee D.H., Park S. Cryptanalysis of SOSEMANUC and SNOW 2.0 using linear masks // Advanced in Cryptology ASIACRYPT 2008. LNCS 5350. Springer-Verlag. P. 524-538.

6. Zhang B., Xu C., Meier W. Fast correlation attacks over extension fields, large-unit linear approximation and cryptanalysis of SNOW 2.0 // http://eprint.iacr.org/2016/311.

7. Алексейчук А.Н., Игнатенко С.М. Оценки эффективности универсальных методов восстановления искаженных линейных рекуррент над кольцом вычетов по модулю 2^N // Зб. наук. праць ІПМЕ НАН України. 2003. Вип. 20. С. 40–48.

8. Алексейчук А.Н., Игнатенко С.М. Метод оптимизации алгоритмов решения систем линейных уравнений с искаженной правой частью над кольцом вычетов по модулю 2^N // Реєстрація, зберігання і обробка даних. 2005. № 1. Т. 7. С. 11–23.

9. Алексейчук А.Н., Игнатенко С.М. Нижняя граница вероятности восстановления истинного решения системы линейных уравнений с искаженной правой частью над кольцом вычетов по модулю 2^N // Захист інформації. 2006. № 4. С. 5-12.

10. Игнатенко С.М. Модификация метода максимума правдоподобия решения систем линейных уравнений с искаженной правой частью над кольцом вычетов по модулю 2^N // Захист інформації. 2007. № 1. С. 63-72. 11. Олексійчук А.М., Ігнатенко С.М., Поремський М.В. Системи лінійних рівнянь зі спотвореними правими частинами над скінченними кільцями // Математичне та комп'ютерне моделювання. Сер.: Техн. науки. 2017. Вип. 15. С. 150-155.

12. Олексійчук А.М., Ігнатенко С.М. Застосування швидкого перетворення Фур'є для розв'язання задачі LPN над скінченими фробеніусовими кільцями // Захист інформації. 2017. № 4. С. 271-277.

13. Кузьмин А.С., Куракин В.Л., Нечаев А.А. Псевдослучайные и полилинейные последовательности // Труды по дискретной математике. Москва : ТВП. Т. 1. 1997. С. 139-202.

14. Олексійчук А.М. Достатня умова стійкості SNOW 2.0-подібних потокових шифрів відносно певних атак зі зв'язаними ключами // Захист інформації. 2016. Т. 18. № 3. С. 261-268.

15. Vaudenay S. On the Lai-Massey scheme // Advanced in Cryptology ASIACRYPT'99. Springer-Verlag. 1999. P. 8-19.

16. Чечёта С.И. Введение в дискретную теорию информации и кодирования: учебное издание. Москва : МЦНМО, 2011. 224 с.

17. Blum A., Kalai A., Wasserman H. Noise-tolerant learning, the parity problem, and the statistical query model // J. ACM. 2003. Vol. 50. № 3. P. 506-519.

18. Bogos S., Tram′er F., Vaudenay S. On solving LPN using BKW and variants. Implementation and analysis // http://eprint.iacr.org/2015/049.

Інститут спеціального зв'язку та захисту інформації
Національного технічного університету України
«Київський політехнічний інститут імені Ігоря Сікорського»

Надійшла до редколегії 07.03.2018

УДК 004.056.55

О.Г. КАЧКО, канд. техн. наук, Ю.І. ГОРБЕНКО, канд. техн. наук, О.С. АКОЛЬЗІНА

АНАЛІЗ АТАК СПЕЦІАЛЬНОГО ТИПУ ЩОДО NTRU-ПОДІБНОГО АЛГОРИТМУ

Вступ

Розвиток та створення квантового комп'ютеру спричинили необхідність пошуку квантово стійких криптографічних механізмів та формування вимог до них. Так, NIST США восени 2017 року прийняв на конкурс постквантових 69 пакетів з кандидатами механізмів асиметричного криптоперетворення (електронні підписи (ЕП), асиметричні шифри(АСШ) та протоколи інкапсуляції ключів(ПІК)) [1], які пройшли попередні тестування. В процесі підготовки кандидатів розробниками враховано, що для практичного застосування механізми криптоперетворення мають задовольняти вимогам криптографічної стійкості, швидкодії та в певній мірі мають бути мало ресурсними. Серед висунутих вимог щодо криптографічної стійкості особливе значення мають спеціальні вимоги щодо каналів витоку по стороннім каналам та можливостям їх перекриття [2, 3]. З'ясувалось, що ця проблема є недостатньо добре вивченою, хоча і надзвичайно актуальною.

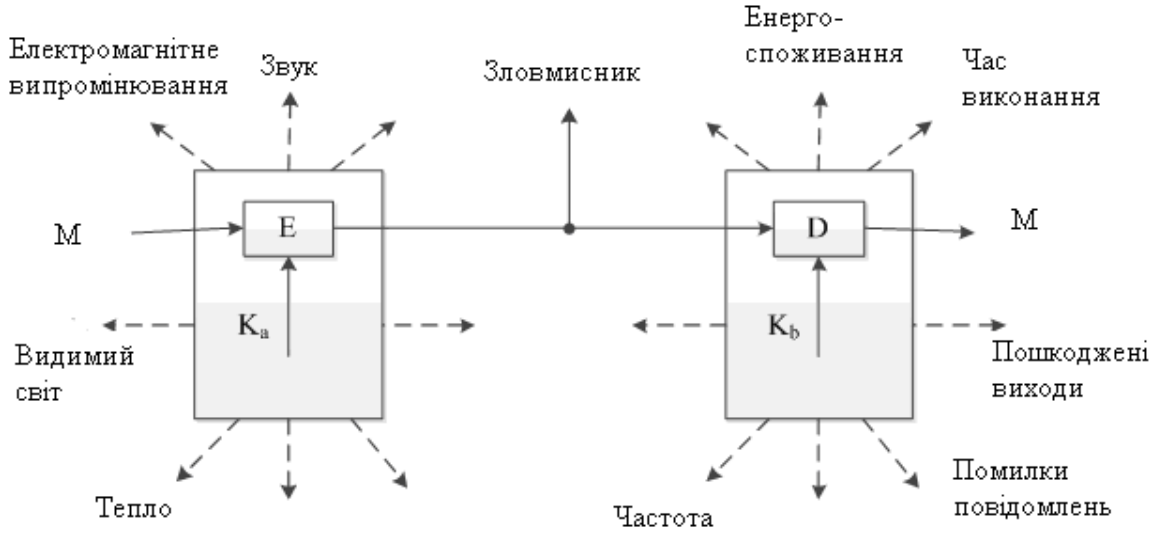
Мета статі – огляд та класифікація атак по стороннім каналам, а також викладення узагальнених підходів до перекриття або ослаблення впливу таких атак в постквантовий період в АСШ NTRU Prime IIT Ukraine [14 – 15].

1. Огляд та класифікація атак спеціального типу

Атаки спеціального виду (side-chanel attacks) можна віднести до атак аналітичного типу. Реалізація цих атак направлена на пошук вразливостей у практичній реалізації криптосистеми, в першу чергу засобу криптографічного захисту інформації (КЗІ). У [2, 3] запропоновано класифікацію спеціальних атак за такими ознаками:

- контроль над обчислювальним процесом;
- спосіб доступу до системи чи засобу;
- метод безпосереднього здійснення атаки.

На рисунку наведено модель, яка пояснює атаки спеціального виду [2, 3]. Розглянемо одну із можливих варіантів класифікації атак.



Криптографічна модель відносно атак спеціального виду

Класифікація спеціальних атак по степеню впливу на обчислювальний процес. Аналіз існуючих джерел [2, 3] показав, що по степеню впливу на обчислювальний процес спеціальні атаки можна поділити:

- на пасивні, коли зловмисник отримує необхідну інформацію без помітного впливу на систему, але система при цьому продовжує функціонувати як і раніше;
- активні, коли зловмисник реалізує деякий вплив на систему, у результаті якого змінюється поведінка системи, але зміни такого роду можуть бути «прозорими» для системи, на яку відбувається напад. При цьому зловмисник у змозі визначати та використати інформацію про систему.

Класифікація спеціальних атак по способу доступу до системи. В залежності від можливості доступу до апаратно-програмного чи апаратного засобу КЗІ можна виділити такі класи атак [2, 4 – 8]:

- агресивні (англ. *invasive*) – коли здійснюється спроба розкриття системи зловмисником та отримання прямого доступу до внутрішніх компонентів;
- напів агресивні (англ. *semi-invasive*) – коли вплив на внутрішні компоненти засобу КЗІ здійснюється без посереднього контакту;
- не агресивні (англ. *non-invasive*) – коли використовується тільки зовнішня інформація – наприклад час обчислення чи споживання енергії. Тобто безпосереднього впливу на систему, що досліджується, немає.

Класифікація спеціальних атак по методу здійснення атаки. Спеціальні атаки, в залежності від методів, які використовуються для аналізу отриманої інформації, можна поділити на [2, 3]:

- прості (англ. *simple side channel attack*) – коли здійснюється дослідження прямої залежності між процесами в пристрої та отриманої зловмисником інформації, а результатом атаки є виділення корисної інформації, наприклад, від рівня шумів;
- диференційні (англ. *differential side channel attack*) – коли використовуються статистичні методи дослідження залежностей між вхідними даними та інформацією, яка отримана під час спостереження. Як правило, при цьому здійснюються велика кількість вимірювань та спеціальна обробка сигналу і корекція помилок.

В процесі здійснення атак на реалізацію засобу КЗІ може здійснюватись аналіз усіх зовнішніх параметрів засобу, а також усі можливі методи порушення його нормального функціонування, аж до його руйнування з метою отримання секретного ключа.

При виконанні атак за часом [2, 3] вимірюється час виконання алгоритму криптоперетворення. У реалізаціях асиметричних алгоритмів час виконання операцій також може залежати як від оброблюваних даних, так і від ключа криптоперетворення (ЕП, АСШ, ПІК). При використанні апаратного рішення у вигляді автомата з жорсткою логікою, навіть час складання за деяким модулем може змінюватися у залежності від реалізації ланцюгів перенесення.

Атаки на реалізацію можуть ґрунтуватись на аналізі всіх споживаних потужностей сучасних обчислювальних пристроїв КЗІ, особливо таких, що побудовані на використанні елементів схемотехніки ТТЛ (англ. TTL), ТТЛШ (англ. TTL(S)), а також частково і КМОП (англ. CMOS). Вона також залежить від оброблюваних даних. Тому у зловмисника з’являється можливість отримати інформацію про внутрішній стан автомату, у тому числі секретний ключ, наприклад шляхом аналізу енергоспоживання при АСШ чи ЕП. Так, атака, що описана у [9], дозволяє на основі аналізу енергоспоживання обчислити вагу Хеммінга (кількість одиничних бітів) оброблюваного блоку. Ця інформація, а також знання виключно відкритих текстів (без знання шифртексту), дає зловмисникові можливість відтворити таємний ключ шифрування.

Крім того, якщо у порушника є можливість порушувати нормальну роботу пристрою (наприклад, вносити збої), то за допомогою спеціальних методів можна відновити практично будь-який секретний параметр системи.

Основною метою фізичної атаки є дослідження особливостей реалізації пристрою КЗІ (мікросхеми), що потрібно для отримання інформації відносно особистого або таємного ключів, наприклад, шляхом дослідження області всередині кристалу ПЛІС. Як правило, такі атаки орієнтовані на специфічні області ПЛІС, які в режимі нормального функціонування є не доступними.

2. Загальні пропозиції відносно протидії атакам спеціального виду

В основу захисту від атак спеціального виду можуть бути покладеними такі методи.

2.1. Фіксована кількість звернень до геш-функції

В роботі [10] показано атаку спеціального виду за часом, яка може розкрити секретний ключ NTRU. Ця атака можлива завдяки тому, що у розшифруванні різних шифротекстів використовується різна кількість звернень до геш-функції. Методом протидії таким атакам є використання механізму доповнення. Розмір доповнення повинен відповідати необхідному рівню криптостійкості. Так, в [10] використовується схема доповнення NAEP, а розмір доповнення дорівнює розміру геш значення, яке задовольняє умові

$$Hlen = \begin{cases} 160 & k \le 112 \\ 256 & k > 112 \end{cases} \quad (1)$$

де k – рівень криптостійкості.

За умови виконання (1) можна сподіватись, що криптоперетворення і, як наслідок, криптосистема, може бути захищеною від атак за часом.

2.2. Рандомізація даних

Метод рандомізації зводиться до «засліплення» даних [11 – 14]. По суті воно зводиться до зміни вхідних даних в деякий непередбачуваний стан. Залежно від характеристик функції «засліплення» вона може виключити деякі або всі витоки корисної інформації. Основною властивістю вхідних даних є їх псевдовипадковість. У криптосистемі «NTRU Prime IIT Ukraine» застосовується засліплюючий поліном, що запобігає витоку інформації про секретний ключ.

2.3. Незалежність від значень

Якщо усі перетворення із особистим ключем та поліномом засліплення при зашифруванні та розшифруванні не залежать від значень засліплюючого поліному та особистого ключа, то про них не можливо по стороннім каналам дізнатися будь-яку інформацію.

Також, якщо в операції множення не використовується значення секретного ключа, то не можливо отримати інформацію про секретний ключ аналізуючи операцію множення по стороннім каналам.

3. Вплив заходів стійкості на кількість ключів NTRU-подібного алгоритму

Аналіз показав, що в будь-якому разі ключі криптоперетворення повинні задовольняти властивостям випадкових послідовностей. До таких властивостей належать: випадковість, рівноймовірність та незалежність. В «NTRU Prime IIT Ukraine» [14, 15] це забезпечується за рахунок фіксованих значень кількостей ненульових елементів у секретних ключах f та g. Так, кількість 1, -1, 0 приблизно є рівною.

У табл. 1 у якості прикладу наведено конкретні значення параметрів для першого, середнього та останнього набору параметрів згідно [16].

Таблиця 1

Приклади параметрів NTRU Prime IIT Ukraine

Параметри				
n	q	t	рівень стійкості k	
439	6833	142	112	1
727	5827	121	205	2
1021	8819	183	298	3

У [14, 15] визначено наступне співвідношення (1, -1, 0) для секретних ключів f та g: для f: кількість 1 та -1 позначається як df та дорівнює $df = 2t$, для g кількість одиниць дорівнює $dg_1 = n/3+1$, кількість -1 дорівнює $dg_{-1} = n/3$.

Таблиця 2

Стійкість NTRU Prime IIT Ukraine

	Рівень стійкості		
	1	2	3
$df = 2t$	184	242	366
$dg_1 = n/3+1$	147	243	341
$dg_{-1} = n/3$	146	242	340

Для того щоб порахувати кількість можливих ключів, визначимо наступну формулу.

Нехай задані n_1 елементів першого типу, n_2 елементів другого типу, ... n_k елементів k-го типу, усього n елементів. Перестановки з повторенням – це варіанти їх розміщення по різним місцям. Їх кількість позначається як $P_n(n_1, n_2, ..., n_k)$.

В цьому випадку кількість перестановок з повторенням:

$$P_n(n_1, n_2, ..., n_k) = \frac{n!}{n_1! n_2! ... n_k!} \tag{2}$$

Ключі в NTRU представляють собою перестановки з повторенням довжини n, що складаються з елементів трьох типів $(1, 0, -1)$. У табл. 3 наведено значення кількості можливих ключів, які отримані при застосуванні формули (2).

Таблиця 3

Кількість ключів NTRU Prime IIT Ukraine

	Рівень стійкості		
	1	2	3
для f	$0,3*10^{193}$	$0,9*10^{344}$	$0,3*10^{482}$
для g	$0,5*10^{207}$	$0,9*10^{344}$	$0,1*10^{485}$

Якщо немає обмеження на кількість 1, -1 для ключів, наприклад як для схеми Crystals-Kyber [17], то для підрахунку треба використовувати формулу розміщення з повторенням:

$$A_n^{'m} = n^m, \tag{3}$$

де n – для ключів це кількість елементів, тобто 3, а m – кількість позицій, тобто розмір ключа.

У табл. 4 наведені значення кількості секретних ключів при відсутності обмежень на кількість коефіцієнтів.

Таблиця 4

Кількості секретних ключів без обмеження на кількість коефіцієнтів

	Рівень стійкості		
Кількість секретних ключів	1	2	3
	$0,3*10^{210}$	$0,7*10^{347}$	$0,1*10^{488}$

Аналіз показав, що при введенні обмежень розмір простору ключів зменшується. У табл. 5 наведені значення у скільки разів зменшується кількість ключів, якщо ввести обмеження на коефіцієнти згідно наведеному вище.

Таблиця 5

Зменшення розміру ключового простору

	Рівень стійкості		
	1	2	3
для f	10^{17}	$0,8*10^{3}$	$0,3*10^{6}$
для g	$0,6*10^{3}$	$0,8*10^{3}$	10^{3}

Таким чином, обмеження на кількість ненульових коефіцієнтів призводить до зменшення кількості ключів від 17-ти до 3-х десяткових порядків. Однак, ця міра є необхідною задля захисту перспективних криптоперетворень постквантового періоду від атак по стороннім каналам.

3. Висновки

В результаті проведених досліджень можна зробити наступні висновки:

1. У загальному випадку певну інформацію про особистий (таємний) ключі можна отримати по таким параметрам як час, енергоспоживання, та будь-яким іншим фізичним показникам обчислювального приладу.

2. Класифікацію спеціальних атак по стороннім каналам можна провести за такими основними ознаками: контролем над обчислювальним процесом, способом доступу до системи чи засобу та методом безпосереднього здійснення атаки.

3. При виконанні атак за часом вимірюється час виконання алгоритму шифрування. У реалізаціях асиметричних алгоритмів час виконання операцій також може залежати як від оброблюваних даних, так і від ключа шифрування. Основною ознакою, яка дозволяє здійснити атаку по значенню часу виконання криптоперетворення стороннім є, наприклад, асиметрія в числі символів (1, -1).

4. Для захисту криптосистеми «NTRU Prime IIT Ukraine» від атак за часом пропонується під час шифрування здійснювати фіксовану кількість звернень до геш-функції, а також здійснювати засліплення даних (що вносить додаткову випадковість). Також усі перетворення, що здійснюються з секретними параметрами, не повинні залежати від конкретних значень цих параметрів.

5. Для забезпечення випадковості, рівноймовірності та незалежності ключових даних можна використовуються поліноми з фіксованою кількістю символів (1, -1, 0). Однак такі обмеження призводять до зменшення кількості ключів від 17-ти порядків до 3-х десяткових порядків. Але використання ключів з фіксованою кількістю символів (1, -1, 0) ключів дозволяє в суттєвій мірі зменшити можливості криптоаналітика по здійсненню атак по спеціальним (стороннім) каналам.

Список літератури:

1. Електронний режим доступу: https://csrc.nist.gov/projects/post-quantum-cryptography.

2. Васильцов І. В. Атаки спеціального виду на криптопристрої та методи боротьби з ними ; за наук. ред. проф. В.П. Широчина. Кременець : Видавничий центр «КОГПІ», 2009. 264 с.

3. Kocher P. Differential Power Analysis/ P. Kocher, J. Jaffe, J. Benjamin // Proc. of Advances in Cryptology (CRYPTO '99). LNCS. 1999. T. 1666. P.388-397.

4. Горбенко Ю. І., Пасічник Р. О., Коряков І. В., Скуліш Є. Д. Організація атак спеціального виду на КРП в групі точок ЕК // Зб. наук. праць національної академії СБУ №4. 2011. С. 193-205.

5. Chnorr C.P. A Hierarchy of Polynomial Time Lattice Basis Reduction Algorithms / C.P. Schnorr // Theoretical Computer Science 53. 1987. P.201-224.

6. Smith K. J. Methodologies for Power Analysis Attacks on Hardware Implementations of AES: Master's thesis, Department of Computer Engineering, Rochester Institute of Technology / K. J. Smith. N: 2009. 109p.

7. D.AZTEC.2. Alternatives to RSA. – Access mode: http://www.ecrypt.eu.org/ecrypt1/documents/D.AZTEC.2-1.2.pdf.

8. Peeters E. Power and Electromagnetic Analysis: Improved Model, Consequences and Comparisons / E. Peeters, F.-X. Standaert, J.-J. Quisquater // Integr. VLSI J. vol. 40. 2007. P. 52-60.

9. Горбенко І. Д., Горбенко Ю. І. Прикладна криптологія. Харків : ХНУРЕ ;Форт, 2012. 868 с.

10.Oswald, E. Randomized addition-subtraction chains as a countermeasure against power attacks / E. Oswald, M. Aigner // Cryptographic Hardware and Embedded Systems – CHES 2001, LNCS, vol.2162. Springer. 2001. P.39-50.

11. Moeller B. Securing elliptic curve point multiplication against side-channel attacks / B. Moeller // Information Security – ISC 2001, LNCS, vol.2200. Springer. 2001. P.324-334.

12. Hasan M. Power analysis attacks and algorithmic approaches to their countermeasures for Koblitz curve cryptosystems / M. Hasan // IEEE Trans. Comput. 2001. Vol.50, no.10. P.1071-1083.

13. Lee M.K. Sliding window method for NTRU / M.K. Lee, J.W. Kim, J.E. Song, K. Park // Applied Cryptography and Network Security – ACNS 2007, LNCS. vol.4521. Springer. 2007. P.432-442.

14. Качко О.Г., Єсіна М.В., Акользіна О.С. Оптимізація алгоритму направленого шифрування NTRU Prime IIT Ukraine з урахуванням відомих атак // Радіотехніка. 2017. Вип. 191. С.11-23.

15. Горбенко І.Д., Качко О.Г., Єсіна М.В. Аналіз алгоритму направленого шифрування NTRU Prime // Радіотехніка. 2017. Вип. 191. С..5-10.

16. Bernstein D.J., Chuengsatiansup Ch., Lange T., van Vredendaal Ch. NTRU Prime // Cryptology ePrint Archive: https://ntruprime.cr.yp.to/ntruprime-20160511.pdf.

17. Joppe Bos, Leo Ducas, Eike Kiltz. CRYSTALS – Kyber: a CCA-secure module-lattice-based KEM // https://eprint.iacr.org/2017/634.

Акціонерне товариство
«Інститут інформаційних технологій»;
Харківський національний
університет радіоелектроніки;
Харківський національний
університет імені В.Н. Каразіна

Надійшла до редколегії 05.03.2018

УДК 004.056.55

А.С. КИЯН, М.С. ЛУЦЕНКО, А.А. КУЗНЕЦОВ, д-р техн. наук

ПЕРВИННИЙ АНАЛІЗ ТА ДОСЛІДЖЕННЯ КОДОВИХ СХЕМ ЕЛЕКТРОННОГО ЦИФРОВОГО ПІДПИСУ ТА НАПРАВЛЕНОГО ШИФРУВАННЯ З NIST PQC

Вступ

Сьогодні ми живемо на межі зміни епох інформаційної безпеки, оскільки поява повномасштабного квантового комп'ютера, винахід якого анонсовано на найближчі 10 – 15 років, є викликом захищеності сучасних криптографічних систем [1 – 3]. У більшості з існуючих криптографічних механізмів і протоколів задача пошуку секретного ключа за відомим відкритим ключем пов'язана з вирішенням відомої та складної математичної задачі (наприклад, дискретного логарифмування, факторизації тощо). Однак квантові обчислення суттєво прискорюють рішення багатьох математичних задач.

Національний інститут стандартів і технологій (NIST) звернулися до громадськості та оголосили про початок конкурсу для відбору претендентів на стандарти постквантових алгоритмів (Post-Quantum Cryptography, PQC), рішення щодо яких планується прийняти в 2020 – 2022 роках [3].

Нині в межах конкурсу дослідники з усього світу вже представили 82 проекти, 23 з яких обґрунтовують схеми електронного цифрового підпису (ЕЦП), 59 – шифрування та інкапсуляцію ключів. Роботи у сфері постквантової криптографії ведуться у п'яти різних напрямках [1 – 3]:

- криптографія, заснована на геш-функціях (Hash-based cryptography);
- криптографія, заснована на кодах виправляючих помилки (Code –based cryptography);
- криптографія, заснована на решітках(Lattice-based cryptography);
- криптографія, заснована на багатовимірних квадратичних системах (Multivariate cryptography);
- шифрування з відкрити ключем.

Варто відзначити, що найбільше досліджень проводять в області криптографії, заснованої на решітках (всього подано 28 проектів) та на кодах (20 претендентів на стандартизацію) [4].

Особливістю конкурсу, оголошеного NIST, є те, що на нього можуть бути подані алгоритми, які базуються на математичних методах, які є недостатньо випробуваними. Тому дослідження таких алгоритмів щодо їх стійкості до класичного та квантового криптоаналізу потребує значних витрат часу. Вищезгаданий факт обумовлює актуальність всебічного вивчення представлених проектів, їх порівняльний аналіз, а також оцінку їх захищеності. В межах даної роботи ми обмежимося дослідженням алгоритмів ЕЦП та направленого шифрування, що засновані на кодах, проведемо їх первинний аналіз та систематизацію.

Характеристика проектів кодових схем ЕЦП

На сьогодні авторами представлено три різних схеми формування та перевірки ЕЦП, алгоритми яких базуються на кодових криптосистемах: pqsigRM, RaCoSS, RankSign. Розглянемо поступово кожну з цих схем.

Схема pqsigRM. Схему pqsigRM було розроблено групою дослідників з Кореї: Wijik Lee, Young-Sik Kim,Yong-Woo Lee та Jong-Seon No [5]. Вона ґрунтується на коді Ріда – Мюллера (РМ), покращуючи схему підпису на основі кодів Гоппа, розроблену свого часу Courtois, Finiasz та Sendrier (CFS) [1]. Перевагами даного алгоритму є контрольований час підписання. У порівнянні з CFS час підпису не залежить від можливості виправлення помилок t. Також час підпису та рівень безпеки контролюється завдяки налаштуванню параметрів. Управління

відношенням між часом підпису та рівнем безпеки здійснюють завдяки змінам параметрів N і w, де N – очікувана кількість ітерацій, w – параметр ваги похибок. Обмеження pqsigRM – це відносно великий розмір відкритого ключа, оскільки код РМ не квазіциклічний, розмір відкритого ключа дорівнює (n – k) × k (де n, k – параметри коду). У проекті представлено експерименти щодо ефективності підпису для різних рівнів безпеки, а саме 128, 196,256, які, як відомо, позначають, що з метою подолання захисту необхідно здійснити 2^{128},2^{196} та 2^{256} операцій відповідно [5].

Схема RacoSS. Назва цього алгоритму розшифровується як Random Code-based Signature Scheme, що в перекладі означає: «Випадкова схема підпису, заснована на кодуванні» [4]. Вона є здобутком спільної роботи японських дослідників (Partha Sarathi Roy, Rui Xu, Kazuhide Fukushima, Shinsaku Kiyomoto, Tsuyoshi Takagi) та вченого з американського університету (Kirill Morozov). Представлено дві версії реалізації цієї схеми: довідкова та оптимізована, перша з яких призначена для покращення розуміння функціонування алгоритму, а друга-для демонстрації продуктивності.

Авторами зазначаються такі переваги RacoSS:

– RaCoSS виявився стійким та екзистенційно непідробним в умовах атаки обраного повідомлення;
– підпис має невеликий розмір у порівнянні з іншими схемами підпису на основі кодування, за виключенням схеми підпису CFS з 81-бітовою безпекою. Але, розміри ключів CFS значно більші, ніж потребує RaCoss;
– процеси, виконувані у алгоритмі (формування ключів, перевірка та формування підпису), можуть бути легко прискорені паралельними обчисленнями.

Незважаючи на всі переваги схеми, вона має також суттєвий недолік: діапазон підпису обмежений [4].

Схема RankSign. Криптосистема RankSign була представлена у 2014 року [4]. Її розробниками виступили Nicolas Aragon, Olivier Ruatta, Philipp e Gaborit, Gilles Zémor та Adrien Hauteville. Ця схема підпису заснована на коді в ранговій метриці. Загальною ідеєю є використання коду LRPC (який є еквівалентом для MDPC в метриці Хеммінга або NTRU в евклідовій метриці) як лазівки для обчислення помилки пов'язаної з повідомленням. Головна проблема цієї криптосистеми полягала у тому, що ймовірність розрізнення підпису та випадкового вектора дорівнювала 2/q (де q – степінь основи поля Fq), тобто повинно використовуватися дуже велике значення q. Через це на конкурс було представлено модифіковану версію RankSign, де додатково відбувається додавання невеликої випадкової помилки до підпису, тобто це дозволяє зменшити спроможність зловмисника розрізнити підписи. Схема підпису має невеликі параметри і є відносно швидкою. Оскільки нам потрібно взяти велике значення q, всі відомі комбінаторні атаки є недієвими для порушення стійкості RankSign. Таким чином, найкращі атаки на неї ґрунтуються на розрахунках Грейбнера. У оцінці безпеки не враховується просторова складність цих алгоритмів, оскільки, зараз не існує квантового прискорення для них, автори очікують, що параметри будуть досить стійкими [4].

Порівняльний аналіз представлених алгоритмів ЕЦП

Порівняльний аналіз представлених алгоритмів ЕЦП доцільно провести з точки зору їх швидкодії та довжини параметрів. У табл. 1 наведено значення основних параметрів для різних версій алгоритмів з різними рівнями забезпечуваної безпеки. З метою продемонструвати значення більш наглядно довжини наведено у байтах.

Продемонструємо дані, наведені у таблиці, за допомогою графіків (рис. 1 – 3) в логарифмічному масштабі. Сутність використання такого масштабу полягає в перетворенні довжин даних наступним чином: $x = \log_{10} X$, де: X – параметр, такий як довжина

відкритого або особистого ключа, довжина шифротекста, яка підлягає масштабуванню; x – результат обчислення десяткового логарифма над масштабованим значенням.

Таблиця 1

Характеристика основних криптографічних параметрів ЕЦП

№	Назва	Версії	Рівень безпеки	Секретний ключ, байт	Відкритий ключ, байт	Підпис, байт
1	pqsigRM	/pqsigRM-4-12	1	1382118	336804	260
		/pqsigRM-6-12	3	334006	501176	516
		/pqsigRM-6-13	5	2105344	2144166	1028
2	RacoSS	Reference	-	703000	99600	586
		Optimized	-	703000	99600	586
3	RankSign	RankSign I	1;$q=2^{32}$	-	80640	11008
		RankSign II	1; $q=2^{24}$	-	96768	12000
		RankSign III	3; $q=2^{32}$	-	155520	17280
		RankSign IV	5; $q=2^{32}$	-	228480	23424

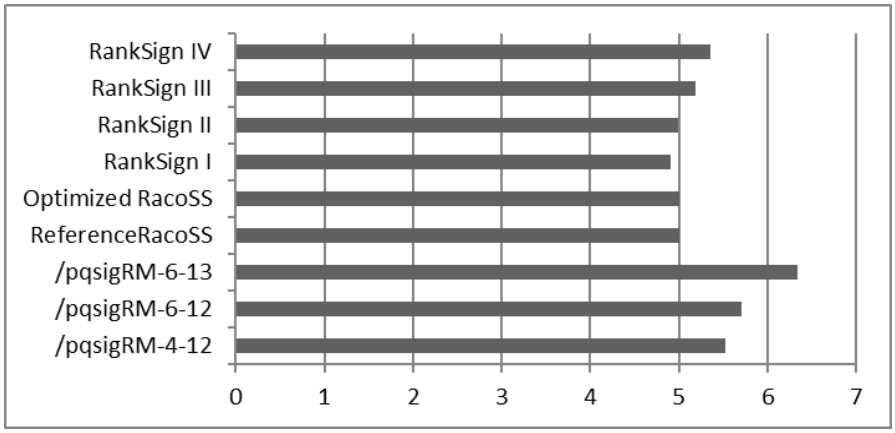


Рис. 1. Порівняння довжин відкритого ключа (в байтах, логарифмічний масштаб) різних схем ЕЦП

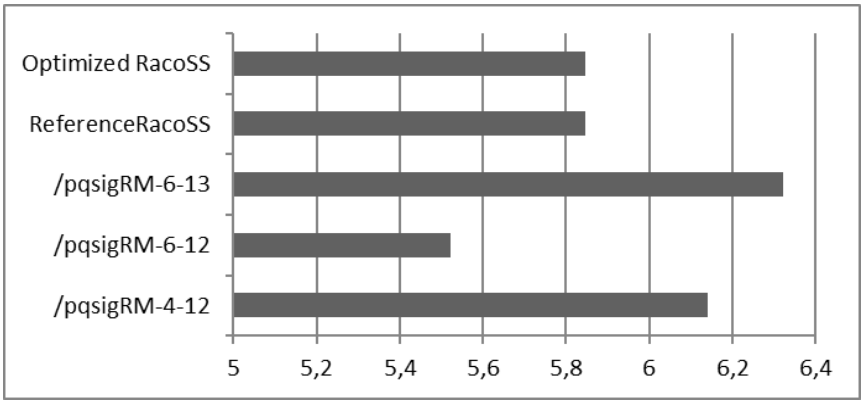


Рис. 2. Порівняння довжин секретного ключа (в байтах, логарифмічний масштаб) різних схем ЕЦП

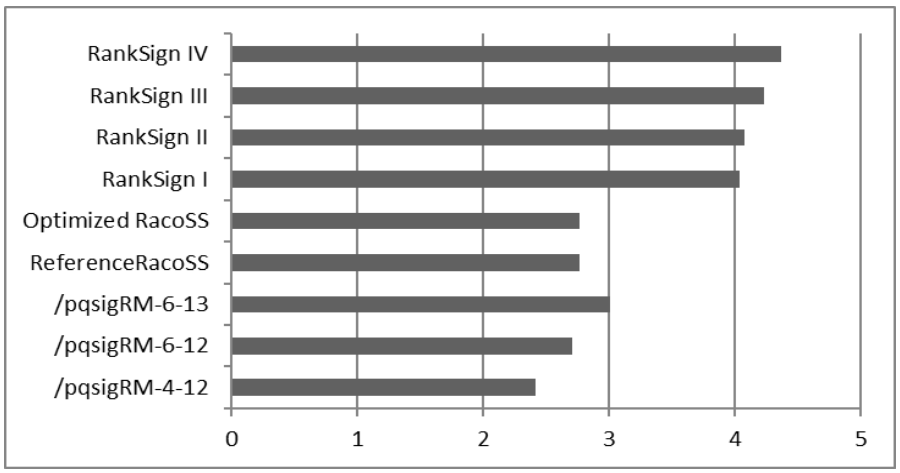


Рис. 3. Порівняння довжин сформованого підпису (в байтах, логарифмічний масштаб) різних схем ЕЦП

Аналізуючи отримані дані, можна зазначити, що для версії алгоритму pqsigRM-6-13 довжини відкритого та секретного ключів найбільші, при цьому довжина шифртексту для даної схеми приймає одне з найменших значень. Дослідити довжину секретного ключа схеми RankSign не вдалося, оскільки вона не передбачає використання секретного ключа. Найбільша довжина шифртексту відповідає алгоритму RankSign і зі зростанням рівня безпеки, що надає ця схема, довжина шифртексту збільшується, як і довжина відкритого ключа.

У табл. 2 наведено показники швидкості генерації ключів, формування і верифікації підпису, а також зазначена обчислювальна платформа, яка використовувалась при випробувані схем. Дані швидкості, що були надані у мілісекундах, зведені до кількості циклів з урахуванням особливостей конкретної обчислювальної платформи.

Таблиця 2

Показники швидкодії алгоритмів ЕЦП

№	Назва	Обчислювальна платформа	Версії	Генерація ключових даних,цикл	Формування підпису, цикл	Верифікація підпису, цикл
1	pqsigRM	Intel(R) Xeon(R) CPU E5-2698 v4 (2,2 ГГц)	/pqsigRM-4-12	9641836	15194705	81178
			/pqsigRM-6-12	1983428	77735436	116906
			/pqsigRM-6-13	22668519	1557210	540378
2	RacoSS	Intel Core i7-4770K CPU (3,50 ГГц)	Reference	24815000000	60900000	31850000
			Optimized	840000000	22680000	213150000
3	RankSign	Intel(R) Core™ i7-4700HQ (3,4 ГГц)	RankSign I	190000000	18600000	7300000
			RankSign II	432000000	33100000	13600000
			RankSign III	537000000	43100000	17500000
			RankSign IV	1030000000	67800000	28200000

Представимо отримані результати за допомогою графічного зображення (рис. 4). Оскільки показники для різних схем різняться в десятки разів, для кращого сприйняття дані наведені у логарифмічному масштабі.

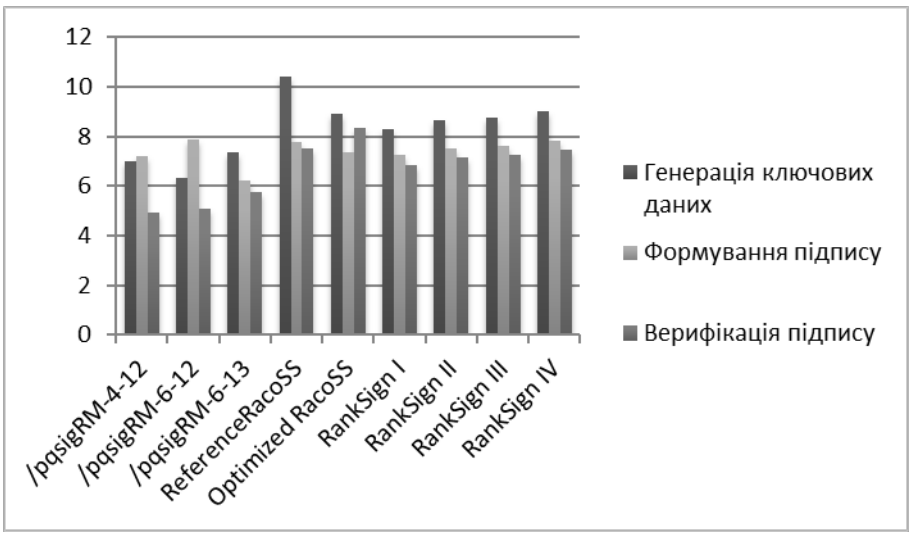


Рис. 4. Показники на швидкості усіх етапах алгоритмів

З точки зору швидкодії очевидно, що більш продуктивним буде той алгоритм, показники для якого більші. Аналізуючи гістограми, бачимо, що оптимізована версія RacoSS (Optimized RacoSS) є найбільш швидкою з усіх представлених алгоритмів. Тоді як схема підпису pqsigRM для різних своїх версій продемонструвала порівняні показники, що є на порядок меншими за швидкість RankSign та RacoSS.

Характеристики проектів направленого шифрування

По результатах аналізу наданих на конкурс проектів було виділено п’ять кодових схеми направленого шифрування: BIG QUAKE [6], HQC [7], LEDApkc [8], LOCKER [4] та McNie [4].

Схема шифрування BIG QUAKE. У рамках проекту запропоновано схему шифрування з відкритим ключем, яка перетворюється в механізм інкапсуляції ключів [6]. Автори проекту (Alain Couvreur, Magali Bardet, Elise Barelli, Olivier Blazy, Rodolfo Canto-Torres, Philippe Gaborit, Ayoub Otmani, Nicolas Sendrier, Jean-Pierre Tillich) передбачають використання у даній схемі двійкових кодів Гоппа. BIQ QUAKE побудовано як і схему Нідеррайтера, але у порівнянні з оригінальною схемою ця пропозиція уникає обчислення бієкції між словами фіксованої довжини та словами із постійною вагою. Це дозволяє уникнути громіздких обчислень, що стосуються великих цілих чисел, і робить схему більш придатною для вбудованої системи з обмеженими обчислювальними ресурсами [6].

Схема шифрування HQC. Авторами схеми шифрування виступили Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loïc Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Edoardo Persichetti, Gilles Zémor [7]. Назва HQC є абревіатурою Hamming Quasi-cyclic, що передбачає використання квазіциклічного коду Хеммінга. HQC – криптосистема з відкритим ключем на основі коду з декількома корисними властивостями:

- за конструкцією HQC дозволяє отримати гібридну схему шифрування з сильними гарантіями безпеки (IND-CCA2) і високою економічністю;
- на відміну від більшості криптосистем, що базуються на кодах, припущення про те, що сімейство кодів, що використовуються, не розрізняється серед випадкових кодів, більше не потрібно;
- схема має аналіз вірогідності помилок дешифрування.

Основними перевагами HQC над існуючими криптосистемами, заснованими на кодах, як стверджують автори, є:

– зменшення її IND-CPA до добре зрозумілих проблем теорії кодування: проблема дешифрування квазіциклічного синдрому;
– стійкість проти атак, спрямованих на відновлення прихованої структури коду, що використовується;
– закриті оцінки невдалого розшифрування.

Серед обмежень криптосистеми можна виділити низькі оцінки шифрування. Можливо зашифрувати 256 біт відкритого тексту, як того вимагає NIST, але збільшуючи об'єм потрібно збільшувати також і параметри [7].

Схема LEDApkc. Цей проект представила група італійських дослідників: Marco Baldi, Alessandro Barenghi, Franco Chiaraluce, Gerardo Pelosi, Paolo Santini [8]. LEDApkc – це криптосистема з відкритим ключем, побудована на кшталт криптосистеми Мак-Еліса на основі лінійних виправляючих помилки кодів. Зокрема, ця схема використовує переваги перевірки парності з низькою щільністю, що забезпечує високу швидкість перетворень та компактність ключових пар. Серед переваг схеми LEDApkc можна виділити наступні:
– побудована на NP-повній проблемі;
– компактні ключові пари (не більше 23 кбітів), секретні ключі мінімального розміру;
– потребує лише операцій додавання та множення у полі $F_{2\,[x]}$;
– повна патентована, автономна, відкрита кодова база даних, написана на ANSI-C99, її легко інтегрувати в існуючі криптографічні бібліотеки [8].

Схема LOCKER. Головними розробникам схеми шифрування LOCKER є Nicolas Aragon, Olivier Blazy, Jean-Christophe Deneuville, Philipp e Gaborit, Adrien Hauteville, Olivier Ruatta, Jean-Pierre Tillich та Gilles Zémor [4]. Пропозиція заснована на варіаціях підходу LRPC. Схема ефективна з точки зору величини параметрів та обчислюваної складності, яка користується властивостями рангової метрики. LOCKER має імовірність відмови, але ця імовірність обґрунтована і може бути дуже низькою від 2^{-64} до 2^{-128}.

Запропонована схема дуже ефективна, як з точки зору розміру ключів, так і обчислювальної складності. Також позитивним моментом є те, що вибір параметрів носить універсальний характер.

Схема McNie. Авторами гібридної схеми, що об'єднає елементи криптосистеми McElice та Niderreiter, є корейські вчені Lucky Galvez, Jon-Lark Kim, Myeong Jae Kim, Young-Sik Kim, Nari Lee [4]. У порівнянні з іншими схемами шифрування McNie забезпечує значно менші розміри відкритих ключів, які збільшуються з більш поступовим темпом зі збільшенням рівня безпеки.

McNie може використовувати різноманітні види відомих блокових кодів в якості вхідних даних, навіть незважаючи на те, що криптосистема McEliece на основі цих кодів була порушена. Причина в тому, що McNie використовує випадковий код, що є більш безпечним, ніж у криптосистемі McEliece. Також завдяки використанню випадкового коді McNie захищений від структурних атак та атак з набором декодованої інформації [4].

Порівняльний аналіз представлених схем направленого шифрування

Порівняльний аналіз представлених на конкурс схем шифирування, як і схем формування ЕЦП, доцільно буде здійснити за двома критеріями: довжинами основних криптографічних параметрів, а також показниками щвидкодію, яку забезпечують кожен з алгоритмів.

У табл. 3 наведено показники довжин секретного, відкритого ключа, а також шифртексту для різних версій представлених схем шифрування, що надають різні рівні безпеки.

Таблиця 3

Характеристика основних криптографічних параметрів схем шифрування

№	Назва	Версії	Рівень безпеки		Секретний ключ, байт	Відкритий ключ, байт	Шифртекст, байт
1	Big Quake	Big Quake-1	1		14772	25482	201
		Big Quake-3	3		30860	84132	406
		Big Quake-5	5		41804	149800	492
2	HQC	Basic- I	1; $P_{пом} \le 2^{-64}$		40	2819	5662
		Basic- II	1; $P_{пом} \le 2^{-96}$		40	3009	6002
		Basic- III	1; $P_{пом} \le 2^{-128}$		40	3125	6234
		Advanced- I	3; $P_{пом} \le 2^{-64}$		40	5115	1021
		Advanced- II	3; $P_{пом} \le 2^{-128}$		40	5499	10982
		Advanced- III	3; $P_{пом} \le 2^{-192}$		40	5884	11752
		Paranoic- I	5; $P_{пом} \le 2^{-64}$		40	7417	14818
		Paranoic-II	5; $P_{пом} \le 2^{-128}$		40	7989	15962
		Paranoic- III	5; $P_{пом} \le 2^{-192}$		40	8503	16990
		Paranoic- IV	5; $P_{пом} \le 2^{-256}$		40	8897	17778
3	LEDApkc	LEDApkc-1	1	$n_0=2$	3480	668	6960
				$n_0=3$	4688	844	7032
				$n_0=4$	6408	1036	8544
		LEDApkc-3	2-3	$n_0=2$	7200	972	14400
				$n_0=3$	10384	1196	15576
				$n_0=4$	13152	1364	17536
		LEDApkc-5	4-5	$n_0=2$	12384	1244	24768
				$n_0=3$	18016	1548	27024
				$n_0=4$	22704	1772	30272
4	LOCKER	LOCKER I	1		-	736,625	800,625
		LOCKER II	3		-	1047,875	1111,875
		LOCKER III	5		-	1190,375	1252,875
		LOCKER IV	1		-	997,375	1061,375
		LOCKER V	3		-	1248,875	1312,875
		LOCKER VI	5		-	1377,625	1441,625
		LOCKER VII	1		-	1545,875	1609,875
		LOCKER VIII	3		-	1881,125	1945,125
		LOCKER IX	5		-	2139,125	2203,125
5	McNie	McNie-1	1		401	417	422
		McNie-3	3		512	539	651
		McNie-5	5		601	647	781

Представимо дані таблиці за допомогою схематичного зображення (рис. 5 – 7). На графіках позначено тільки по три версії кожного алгоритму для кращого візуального сприйняття. Було обрано ті варіанти схем шифрування, що забезпечують різні рівні безпеки.

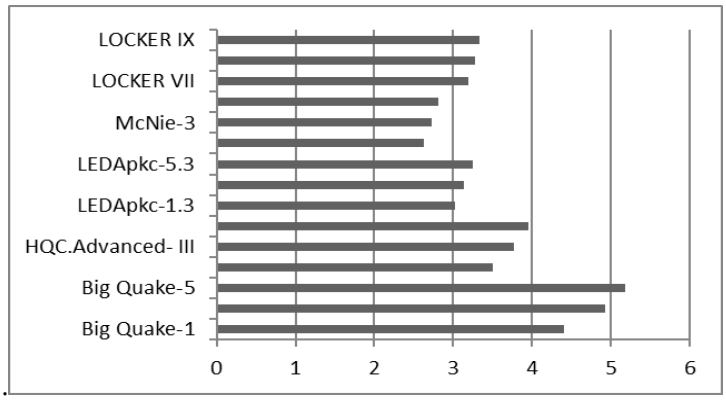


Рис. 5. Порівняння довжин відкритих ключів (в байтах, логарифмічний масштаб) різних схем шифрування

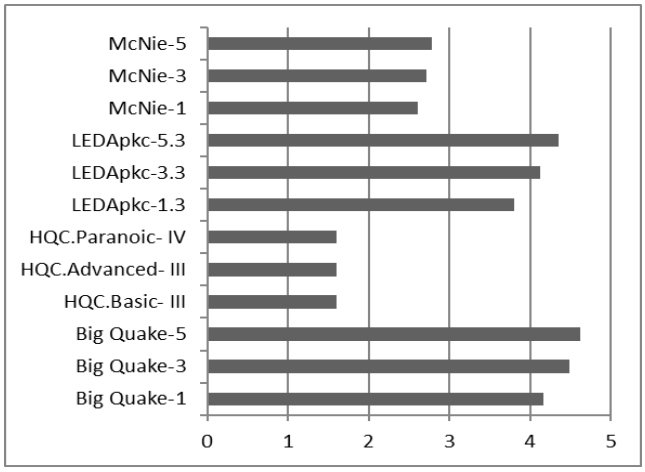


Рис. 6. Порівняння довжин секретних ключів (в байтах, логарифмічний масштаб) різних схем шифрування

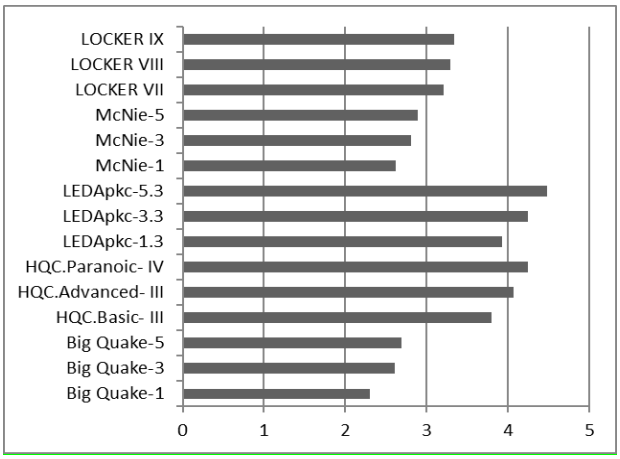


Рис. 7. Порівняння довжин шифртексту (в байтах, логарифмічний масштаб) різних схем шифрування

Аналізуючи отримані результати, варто зазначити, що довжини відкритого ключа та шифртексту для алгоритму Big Quake є найбільшими. McNie напроти демонструє найменші показники для усіх трьох параметрів, при тому що здатен забезпечувати п’ятий рівень безпеки, як і інші схеми.

Наступним кроком дослідимо швидкодію розглянутих вище схем шифрування, що представлені на конкурс. Вивчається швидкість трьох етапів алгоритмів: генерація ключових

даних, шифрування та розшифрування. Отримані результати представлені у табл. 4. Дані, що були надані у мілісекундах, було зведено до кількості циклів, які потребує виконання алгоритму з урахуванням особливостей використовуваної обчислювальної платформи.

Таблиця 4

Показники швидкодії схем шифрування

№	Назва	Обчислювальна платформа	Версії	Генерація ключових даних,цикл	Шифрування, цикл	Дешифрування, цикл
1	Big Quake	Intel(R) Xeon™ E3-1240v5 (3,5 ГГц)	Big Quake-1	268000000	4305000	4935000
			Big Quake-3	8641500000	10500000	31885000
			Big Quake-5	16509500000	15610000	47950000
2	HQC	Intel(R) Core™ i7-4770 (3.4ГГц)	Basic- I	578000	1224000	1938000
			Basic- II	612000	1292000	2074000
			Basic- III	646000	1360000	2142000
			Advanced- I	1258000	2618000	3842000
			Advanced- II	1360000	2822000	4114000
			Advanced-III	1462000	3026000	4352000
			Paranoic- I	2210000	4692000	6664000
			Paranoic-II	2584000	5440000	7548000
			Paranoic-III	2652000	5610000	7990000
			Paranoic-IV	2788000	5984000	850000
3	LEDApkc	AMD Ryzen 5 1600 CPU (3.2 ГГц)	1	144960000	9952000	66784000
				67072000	9300000	80576000
				57568000	12608000	90560000
			2-3	635168000	38592000	200160000
				321248000	41792000	184256000
				232896000	45376000	191200000
			4-5	1788288000	108672000	369152000
				956512	119296000	374176000
				668480000	127520000	503136000
4	LOCKER	IntelR Core TMi7-4700HQ CPU(3,4 ГГц)	LOCKER-I	2710000	550000	2570000
			LOCKER-II	3190000	540000	1080000
			LOCKER-III	3580000	600000	3770000
			LOCKER-IV	3720000	710000	2860000
			LOCKER-V	4360000	860000	4320000
			LOCKER-VI	4680000	750000	4060000
			LOCKER-VII	8440000	1350000	4780000
			LOCKER-VIII	9480000	1390000	5000000
			LOCKER-IX	10400000	1490000	6600000
5	McNie	Intel Core i7-4790 (3.6 ГГц)	McNie-1	280800000	2444400	5788800
			McNie-2	511200000	3492000	8445600
			McNie-3	668880000	4287600	10342800

Продемонструємо наведені результати у графічному вигляді (рис. 8 – 11). Варто зазначити, що через те, що дані для різних алгоритмів різняться в сотні разів, графіки було побудовано з використанням логарифмічного масштабу.

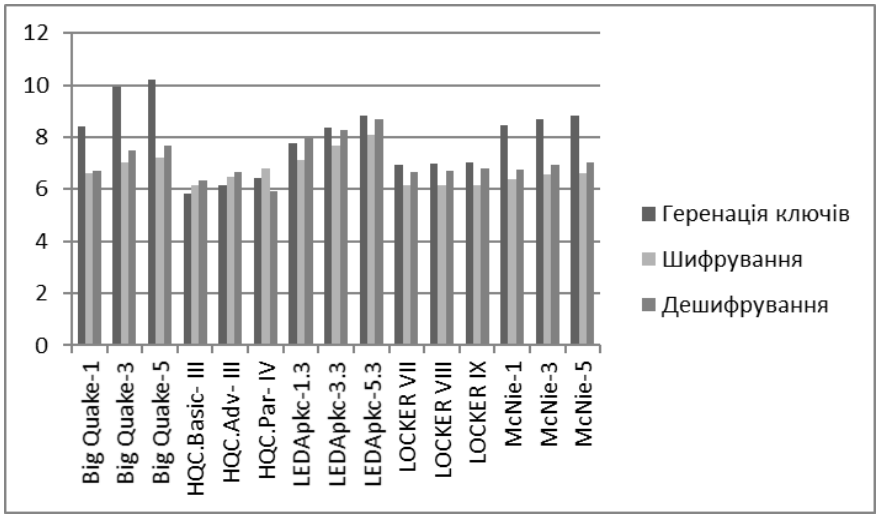


Рис. 8. Гістограма показників швидкодії схем шифрування (логарифмічний масштаб)

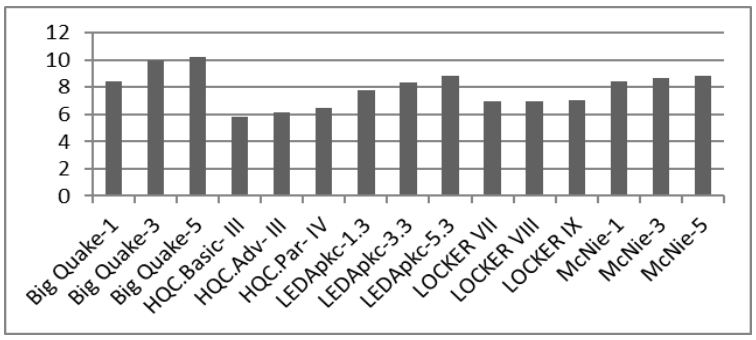


Рис. 9. Порівняння показників швидкості генерації ключових даних різних схем шифрування (логарифмічний масштаб)

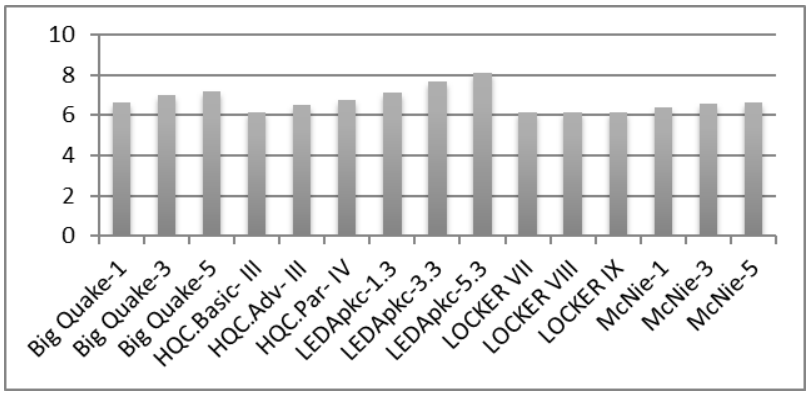


Рис. 10. Порівняння показників швидкості шифрування різних схем шифрування (логарифмічний масштаб)

Big Quake надає найбільшу швидкість генерації ключів. Приблизно порівняні результати показали алгоритми McNie та LEDApkc, тоді як версії алгоритму HQC забезпечують найменшу швидкодію з усіх розглянутих схем. Отже, Big Quake надає найбільшу швидкість генерації ключів, тоді як версії алгоритму HQC забезпечують меншу швидкодію, порівняно з іншими алгоритмами.

Аналізуючи дані, бачимо, що швидкість шифрування досить висока у всіх схем шифрування, але LEDApkc-5.3 забезпечує найкращі показники.

Швидкість дешифрування порівняна у алгоритмів McNie, Big Quake та HQC, тоді як показники LEDApkc є найкращими.

Швидкість дешифрування порівняна у алгоритмів McNie, Big Quake та HQC, тоді як показники LEDApkc є найкращими.

Отже, з точки зору швидкодії, найефективнішою з представлених схем є схема LEDApkc у всіх її варіантах, а HQC, у свою чергу, показала найгірші результати.

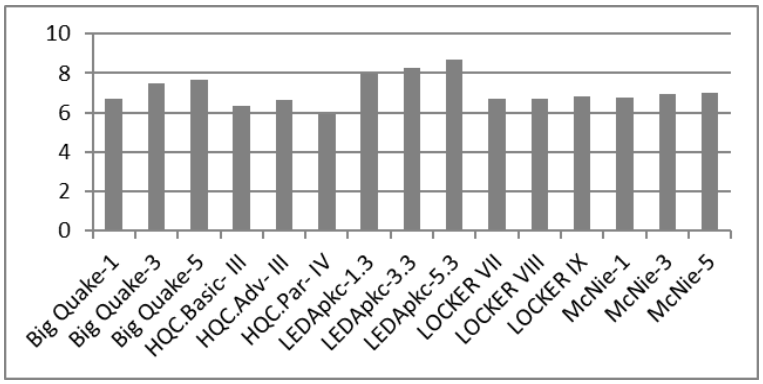


Рис. 11. Порівняння показників швидкості дешифрування різних схем шифрування (логарифмічний масштаб)

Висновки

Висока імовірність появи повномасштабного квантового комп'ютера в найближчі роки обумовлює зростання актуальності досліджень в області постквантової криптографії з метою стандартизації нових алгоритмів, що будуть здатні залишатися стійкими в умовах застосування квантових обчислень. Національний інститут стандартів і технологій США оголосив про початок конкурсу для відбору претендентів на стандарти постквантових алгоритмів, рішення щодо яких планується прийняти в 2020 – 2022 роках. На сьогодні у рамках конкурсу подано 82 проекти.

Криптографія, що ґрунтується на кодах, нині вважається одним з найперспективніших напрямків. Це підтверджується тим, що з 82 проектів, представлених на конкурс, 20 базуються саме на кодах. Серед них три схеми формування та верифікації електронного цифрового підпису, п'ять схем шифрування та дванадцять механізмів інкапсуляції ключів.

В цій роботі алгоритми формування підпису і схеми направленого шифрування порівняно за двома критеріями: за довжинами основних криптографічних параметрів та за показниками швидкодії, які забезпечує кожен з алгоритмів. З точки зору швидкодії, найефективнішою виявилась схема направленого шифрування LEDApkc та алгоритм формування електронного цифрового підпису RacoSS. Використовуючи перший критерій порівняння, найкращі показники продемонстрували pqsigRM та McNie для формування ЕЦП та шифрування відповідно.

Слід відмітити, що наведені оцінки спираються на дані, що були надані безпосередньо авторами проектів, тобто вважати їх повністю об'єктивними неможливо. Для всебічного дослідження з метою стандартизації вичерпний аналіз триватиме роками і це є перспективним напрямком подальших робіт.

Список літератури:

1. Bernstein D., Buchmann J. and Dahmen E. Post-Quantum Cryptography. Springer-Verlag, Berlin-Heidleberg, 2009. 245 p.

2. Koblitz N. and Menezes A.J.. A Riddle Wrapped in an Enigma. [Електронний ресурс] URL: https://eprint.iacr.org/2015/1018.pdf, Oct. 20, 2015 [Aug. 21, 2016]

3. Moody D. Post-Quntum Cryptography: NIST's Plan for the Future. The Seventh International Conference on Post-Quntum Cryptography, Japan, 2016. [Електронний ресурс] URL: https://pqcrypto2016.jp/data/pqc2016_nist_announcement.pdf [March 8, 2016].

4. Computer Security Resource Center [Електронний ресурс] URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions

5. A modified RM code-based post-quantum digital signature algorithm [Електронний ресурс]. URL: https://sites.google.com/view/pqsigrm/home

6. Binary Goppa QUAsi-cyclic Key Encsapulation [Електронний ресурс] URL: https://bigquake.inria.fr/

7. Hamming Quasi-Cyclic [Електронний ресурс]. URL: http://pqc-hqc.org/

8. LEDApkc Public Key Cryptosystem [Електронний ресурс]. URL: https://www.ledacrypt.org/LEDApkc/

Харківський національний
університет імені В.Н.Каразіна

Надійшла до редколегії 10.03.2018

УДК 004.056.55

М.С. ЛУЦЕНКО, А.С. КИЯН, Т.Ю. КУЗНЕЦОВА, А.А. КУЗНЕЦОВ, д-р техн. наук

АНАЛИЗ И СРАВНИТЕЛЬНЫЕ ИССЛЕДОВАНИЯ КОДОВЫХ СХЕМ ИНКАПСУЛЯЦИИ КЛЮЧЕЙ, ПРЕДСТАВЛЕННЫХ НА КОНКУРС NIST PQC

Введение

В конце 2016 года Национальным институтом стандартов и технологий (National Institute of Standards and Technology, NIST) США объявлен конкурс постквантовой криптографии (Post-Quantum Cryptography, PQC) [1], в частности алгоритмов электронной цифровой подписи, направленного шифрования и схем инкапсуляции ключей. Среди перспективных направлений исследований особое место занимают кодовые криптосистемы (Code-Based Public-Key Cryptosystems), позволяющие эффективно реализовывать все три группы алгоритмов.

Цель работы – анализ и сравнительные исследования кодовых схем инкапсуляции ключей, представленных на конкурс NIST PQC. Рассмотрены все 12 алгоритмов инкапсуляции ключей, представленных на конкурс: BIKE [2], Classic McElice [3], DAGS [4], Edon-K [5], LAKE [6], LedaKem [7], Lepton [8], NTS-KEM [9], Ouroboros-R [10], QC –MDPC KEM [11], RLCE-KEM [12], RQC [13]. Для первичной оценки криптографической стойкости проведен анализ соответствия алгоритмов инкапсуляции ключей современным требованиям к криптосистемам с открытым ключом, а именно – обеспечению свойств неразличимости [14, 15]. Свойство неразличимости шифротекста определяет криптостойкость алгоритма к атаке на основе подобранного открытого текста. Обеспечение такого свойства неразличимости на основе открытого теста (IND-CPA) считается основным требованием для большинства доказуемо защищенных криптосистем с открытым ключом [14, 15], хотя некоторые схемы также обеспечивают криптографическую стойкость против атак на основе подобранного шифротекста и адаптивных атак на основе подобранного шифротекста. Такие свойства неразличимости обозначаются, как IND-CCA1 и IND-CCA2 соответственно [14, 15].

Для оценки уровня криптостойкости в классической и постквантовой криптографии для каждого алгоритма и его вариации используются обеспечиваемый уровень криптостойкости и соответствующие основные параметры преобразований. При описании требований к алгоритмам, подаваемым на конкурс, были определены уровни криптографической стойкости:

- Уровень 1: любая атака, которая взламывает IND-CCA-стойкий алгоритм, должна требовать вычислительных ресурсов, сравнимых или превышающих требуемые для поиска ключа на блочном шифре с 128-битным ключом (например, AES-128);
- Уровень 3: если существует атака на IND-CCA-криптостойкий алгоритм, то для проведения такой атаки должны обеспечиваться вычислительные ресурсы, соизмеримые или превышающие требуемые для поиска ключа на блочном шифре с 192-битным ключом (например, AES-192);
- Уровень 5: любая атака, которая нарушает криптостойкость IND-CCA-стойкой схемы, должна требовать вычислительных ресурсов, сравнимых или превышающих требуемые для поиска ключа на блочном шифре с 256-битным ключом (AES-256).

В работе также проведен сравнительный анализ показателей быстродействия неоптимизированных версий алгоритмов. Проводимый анализ носит первичный характер, все результаты оценки показателей стойкости и быстродействия основаны на экспериментах, проведенных авторами (разработчиками) алгоритмов.

Схемы инкапсуляции ключей: краткая характеристика

BIKE (BIt Flipping Key Encapsulation) – схема инкапсуляции ключей, представленная группой ученых – Николя Арагон, Пауло Баррето, Слим Беттайб, Лойк Биду, Оливье Блази, Жан-Кристоф Деневиль, Филлип Габорит, Шей Герон, Тим Ганейсу, Карлос Агилар Мелхор,

Рафаэль Мисоцки, Эдоардо Персикетти, Николя Сенриер, Жан-Пьер Тиллих, Жиль Земор – из университетов стран Франции, США, Израиля, Германии [2]. Квазициклические коды с проверкой на четность (QC-MDPC) с умеренной плотностью. Могут быть декодированы с использованием техники бит флиппинг. Алгоритм обладает IND-CPA криптостойкостью, из-за применения техники бит флиппинга ожидается и обеспечение IND-CCA стойкости. Авторами представлены три модификации алгоритма: BIKE-1, BIKE-2 и BIKE-3. Схема BIKE-1 основана на вариации алгоритма Мак-Элиса. В BIKE-1 обеспечивается ускоренная генерация ключей. Открытый ключ имеет удвоенную длину, по сравнению с BIKE-2. В основе алгоритма BIKE-2 лежит криптосистема Нидеррайтера с проверочной матрицей на четность. Вариация BIKE-3 по основным преобразованиям напоминает BIKE-1, но обладает значительно большим, по мнению авторов, запасом криптостойкости. Для каждой из вариации приведены входные параметры в зависимости от обеспечиваемого уровня криптостойкости (обеспечиваются 1, 3 и 5 уровни).

Classic McElice – схема предложенная, группой исследователей, в которую входят: Даниэль Дж. Бернстайн, Тун Чжоу, Таня Ланге, Инго фон Маурич, Рафаэль Мисоцки, Рубен Нидерхаген, Эдоардо Персикетти, Кристиан Петерс, Питер Швабе, Николя Сенриер, Якуб Сезер; из стран США, Япония, Нидерланды, Германия, Франция [3]. Вариация алгоритма Мак-Элиса, основанная на двоичных кодах Гоппы. Данный алгоритм инкапсуляции ключей разработан для обеспечения безопасности IND-CCA2 на очень высоком уровне криптостойкости. Авторы предполагают, что алгоритм может найти эффективное применение даже в системах с ограниченными вычислительными возможностями и ресурсами и при этом сохранить эффективную криптостойкость.

DAGS (Key Encapsulation from DyAdic GS Codes) – исследователи из университетов Нидерландов, США, Сенегала, Франции, Бразилии, предоставили на конкурс алгоритм инкапсуляции ключей. В группу разработчиков входят: Густаво Банегас, Паоло С. Л. М. Барретo, Брайс Одилон Бойддже, Пьер-Луис Кайрел, Гилберт Ндоллане Диона, Крис Гай, Шейх Тикумба Гуи, Ричард Хаусслер, Жан-Бело-Кламти, Усмане Ндиайе, Дюк Три Нгуен [4]. Алгоритм DAGS использует квазидвоичный (QD) подход с использованием обобщенных кодов Сривастава. Авторы утверждают, что это первый алгоритм, основанный на структурированных алгебраических кодах, обеспечивающих не только IND-CPA криптостойкость, но и IND-CCA. В целом, предложенная схема предлагает большую гибкость в обмене ключами. Предположительно алгоритм может найти применение в приложениях для Интернета.

Edon-K – схема инкапсуляции, представленная норвежскими учеными Данилой Глигороски, Кристианом Гьёстином [5]. Этот алгоритм основан на схеме Мак-Элиса, но использует другое семейство кодов. Эти коды определены над другим полем и не основываются на метрике Хэмминга. Такой подход позволяет значительно сокращать длину открытых ключей, по сравнению со схемой Мак-Элиса. Код, используемый в Edon-K, является суперкодом с ранговой проверкой на четность (LRPC) очень малого ранга (1 или 2). Соответствующая матрица проверки на четность для суперкода такого низкого ранга может быть легко получена для открытого ключа. Алгоритм Edon-K обеспечивает уровни криптостойкости 1 и 3. Edon-K предназначен для обеспечения безопасности CCA2 без необходимости дополнительного (потенциально дорогостоящего) преобразования CPA-CCA.

LAKE (Low rAnk parity check codes Key Exchange) – еще один алгоритм, представленный группой ученых из Франции. Авторами алгоритма выступили Николя Арагон, Оливье Блази, Жан-Кристоф Деневиль, Филипп Габорит, Адриен Хаутвиль, Оливье Руатта, Жан-Пьер Тиллих, Жиль Земор [6]. Алгоритм основан на кодах с проверкой на четность Ideal-LRPC и механизме инкапсуляции ключей IND-CPA (KEM). Схема имеет некоторую вероятность ошибки при деинкапсуляции, которая, по мнению авторов, может быть легко устранена. Существует три вариации схемы для обеспечения трех уровней криптостойкости 1, 3 и 5. Предложенная схема очень эффективна, как с точки зрения выбранных размеров основных параметров (ключей и шифротекста), так и вычислительной сложности.

LedaKem (Low dEnsity coDe-bAsed key encapsulation mechanism) – ученые Марко Балди, Алессандро Баренги, Франко Чиаралесе, Херардо Пелоси, Паоло Сантини из Италии представили алгоритм инкапсуляции ключей [7]. Схема основывается на криптосистеме Нидеррайтера с линейной коррекцией ошибок. LEDAkem использует преимущества использования квазициклических кодов четности с низкой плотностью (QC-LDPC), обеспечивающих высокие скорости декодирования и малые длины ключей. Следует отметить крайне малую длину получаемого шифротекста – 64 байта, даже при уровне криптостойкости 5. Схема обладает IND-CCA криптостойкостью. Представленные вариации алгоритма с различными входными параметрами обеспечивают необходимую криптостойкость для уровней 1, 3, 5. В свою очередь, для каждой вариации определены по три подварианта с разным количеством циркулянтных блоков (n_0).

Lepton (LEarning PariTy with Noise) – китайский алгоритм инкапсуляции, представленный авторами Ю Ю, Цзян Чжан [8]. Алгоритм Lepton основан на определении четности с помехами (Learning Parity with Noise). Авторы представили две версии алгоритма. В данной схеме присутствует вероятность ошибки, которая колеблется для разных вариаций алгоритма от 2^{-87} до 2^{-148}. Первый Lepton.CPA направлен на достижение CPA-безопасности и основан на Ring-CLPN (Compact Learning Parity with Noise). Второй вариант Lepton.CCA представляет собой схему KEM для достижения CCA-безопасности, которая получается путем применения преобразования Фудзисаки – Окамото над Lepton.CPA.

NTS-KEM – исследователи из Великобритании, подали на конкурс алгоритм инкапсуляции. Авторы – Мартин Альбрехт, Карлос Сид, Кеннет Г. Патерсон, Цзэн Юнг Тхай, Мартин Томлинсон [9]. NTS-KEM можно рассматривать как вариант схемы шифрования с открытым ключом Мак-Элиса – двоичные линейные коды Гоппы используются в криптосистеме Нидеррайтера. NTS-KEM обеспечивает безопасность IND-CCA (как KEM) в модели случайного предсказателя, используя преобразование, похожее на преобразования Фудзисаки-Окамото или Дента. Авторы представили три версии алгоритма для обеспечения трех уровней криптостойкости.

Ouroboros-R – алгоритм, представленный исследователями – Карлос Агилар Мелхор, Жан-Кристоф Деневиль, Николя Арагон, Филипп Габорит, Слим Беттейб, Адриен Хаутвиль, Лойк Биду, Жиль Земо; Франция [10]. Используемый квазициклический код позволяет ускорить процесс декодирования при увеличении длины шифротекста. Алгоритм имеет некоторые схожие черты с NTRU-подобными схемами. Ouroboros также имеет вероятность отказа (как и другие протоколы NTRU), в связи с используемым алгоритмом декодирования. Ouroboros-R обладает криптостойкостью IND-CPA в соответствии с предположениями 2-QCRSD и 3-QCRSD.

QC–MDPC KEM – алгоритм разработали Ацуши Ямада, Эдвард Итон, Кассем Калач, Филип Лафранс, Алекс Родитель; Канада [11]. Алгоритм основан на криптосистеме Мак-Элиса. В QC-MDPC KEM используется квазициклическая проверка на четность с умеренной плотностью. Авторами указано, что алгоритм может быть недостаточно быстрым, по сравнению с другими алгоритмами. Алгоритм обеспечивает IND-CPA криптостойкость.

RLCE-KEM – схема инкапсуляции ключей исследователя Юн Ван из США [12]. В алгоритме используется схема шифрования Мак-Элиса на основе случайного линейного кода (RLCE). Преимущество схемы RLCE заключается в том, что ее криптостойкость не зависит от какой-либо конкретной структуры базовых линейных кодов. Считается, что безопасность RLCE зависит от NP-сложности декодирования случайных линейных кодов. Автором представлено несколько вариаций алгоритма, которые обеспечивают три уровня криптостойкости.

RQC (Rank Quasi-Cyclic) – алгоритм сформирован группой французских ученых, в состав которой входят: Карлос Агилар Мелхор, Николя Арагон, Слим Беттейб, Лойк Биду, Оливье Блази, Жан-Кристоф Деневиль, Филипп Габорит, Жиль Земор [13]. Схема RQC основана на квазциклическом коде. Используемый подход для инкапсуляции ключей позволяет

гарантировать IND-CCA2 криптостойкость и обеспечивает высокие показатели эффективности. Предлагаются различные значения параметров для уровней безопасности 1, 3 и 5. Авторы указывают, что алгоритм имеет нулевую вероятность отказа декодирования.

В работе проанализированы основные параметры преобразования для всех приведенных выше алгоритмов на основе характеристик, указанных авторами. В табл. 1 приведены значения основных параметров для различных версий алгоритмов и их вариации, в соответствии с обеспечиваемым уровнем криптостойкости. Для алгоритма BIKE длины личного и открытого ключей шифротекста авторы привели в битах. Для наглядности эти параметры нормированы в байты.

Таблица 1

Характеристика основных криптографических параметров

№	Название	Версия	Уровень безопасности	Личный ключ, байт	Открытый ключ, байт	Шифротекст, байт
1	BIKE	BIKE-1	1	266,25	2540,75	2540,75
			3	287	5473,25	5473,25
			5	548	8187,25	8187,25
		BIKE-2	1	266,25	1270,375	1270,375
			3	412	2736,625	2736,625
			5	548	4093,625	4093,625
		BIKE-3	1	251,25	2756,75	2756,75
			3	396	5420,75	5420,75
			5	565,25	9032,75	9032,75
2	Classic McElice	mceliece8192128	1	14080	1357824	240
		mceliece6960119	3	13908	1047319	226
3	DAGS	DAGS_1	1	432640	6760	552
		DAGS_3	3	1284096	8448	944
		DAGS_5	5	2230272	11616	1616
4	Edon-K	EDON-K128 ref	1	32	2576	2336
		EDON-K192 ref	3	32	2192	2736
5	LAKE	LAKE I	1	-	3149	-
		LAKE II	3	-	4717	-
		LAKE III	5	-	6313	-
6	LedaKem		1, $n_0 = 2$	668	3480	32
			1, $n_0 = 3$	844	4688	32
			1, $n_0 = 4$	1036	6408	32
			2-3, $n_0 = 2$	972	7200	48
			2-3, $n_0 = 3$	1196	10384	48
			2-3, $n_0 = 4$	1364	13152	48
			4-5, $n_0 = 2$	1244	12384	64
			4-5, $n_0 = 3$	1548	18016	64
			4-5, $n_0 = 4$	1772	22704	64
7	Lepton. CPA	Light I	1	1045	32	1585
		Light II	1	1045	40	1966
		Moderate I	1	2052	38	2465
		Moderate II	1	2052	48	2765
		Moderate III	3	2052	56	2973
		Moderate IV	5	2052	74	3989
		Paranoid I	5	4128	70	5303
		Paranoid II	5	4128	80	5557
	Lepton.	Light I	1	1045	1077	1617

№	Название	Версия	Уровень безопас-ности	Личный ключ, байт	Открытый ключ, байт	Шифро-текст, байт
	CCA	Light II	1	1045	1085	1998
		Moderate I	1	2052	2090	2497
		Moderate II	1	2052	2100	2751
		Moderate III	3	2052	2018	3005
		Moderate IV	5	2052	2126	4021
		Paranoid I	5	4128	4198	5335
		Paranoid II	5	4128	4208	5589
8	NTS-KEM	NTS-KEM(12,64)	1	9216	319488	128
		NTS-KEM(13,80)	3	17524	929760	162
		NTS-KEM(13,136)	5	19890	1419704	253
9	Ouroboros-R	Ouroboros-R I	1	1180	1180	1180
		Ouroboros-R II	3	1490	1490	1490
		Ouroboros-R III	5	2128	2128	2128
10	QC –MDPC KEM	QC –MDPC KEM 58	58	-	4801	-
		QC –MDPC KEM 86	86	-	9857	-
		QC –MDPC KEM 154	154	-	32771	-
11	RLCE-KEM	ID = 0	1	310116	188001	988
		ID = 1	1	179946	118441	785
		ID = 2	3	747393	450761	1545
		ID = 3	3	440008	287371	138
		ID = 4	5	1773271	1232001	2640
		ID = 5	5	1049176	742089	2023
		ID = 6		1059	626	57
12	RQC(Rank Quasi-Cyclic)	RQC-I	1	1491	1491	1055
		RQC-II	3	2741	2741	2805
		RQC-III	5	3510	3510	3574

Так как значения параметров разнятся на несколько порядков, для наглядного представления на рис. 1 – 3 данные длины личных и открытых ключей, а также длины соответствующих шифротекстов представлены в логарифмическом масштабе. Суть использования такого масштабирования заключается в преобразовании длин данных следующим образом: $x = \log_{10} X$, где X – параметр, такой как длина открытого или личного ключа, длина шифротекста, который подлежит масштабированию; x – результат вычисления десятичного логарифма над масштабируемым значением.

Данные на гистограммах приведены для всех вариаций алгоритмов и отсортированы по уменьшению длины. Следует отметить, что в зависимости от предполагаемой сферы применения алгоритма при одном и том же обеспечиваемом уровне криптостойкости будут предпочтительны разные длины ключей. Например, для использования схемы инкапсуляции ключей в системах с ограниченными ресурсами предпочтительнее использовать алгоритмы с более короткими длинами параметров.

Согласно данным, приведенным на рис. 1, можно заключить, что наименьшей длиной личного ключа обладает схема Edon-K, для обеих вариаций – 32 байта. Более длинные ключи имеет схема BIKE, причем закономерно, что вариация BIKE-3, обеспечивающая уровень криптостойкости 5, имеет больший личный ключ, в отличие от BIKE-1 и BIKE-2. Далее следуют примерно сравнимые по длине личных ключей вариации алгоритмов LedaKem, Lepton.CPA и Lepton.CCA, Ouroboros-R. Наибольшие личные ключи у алгоритмов DAGS-5 и RLCE-KEM – 2230272 и 1049176 байт соответственно.

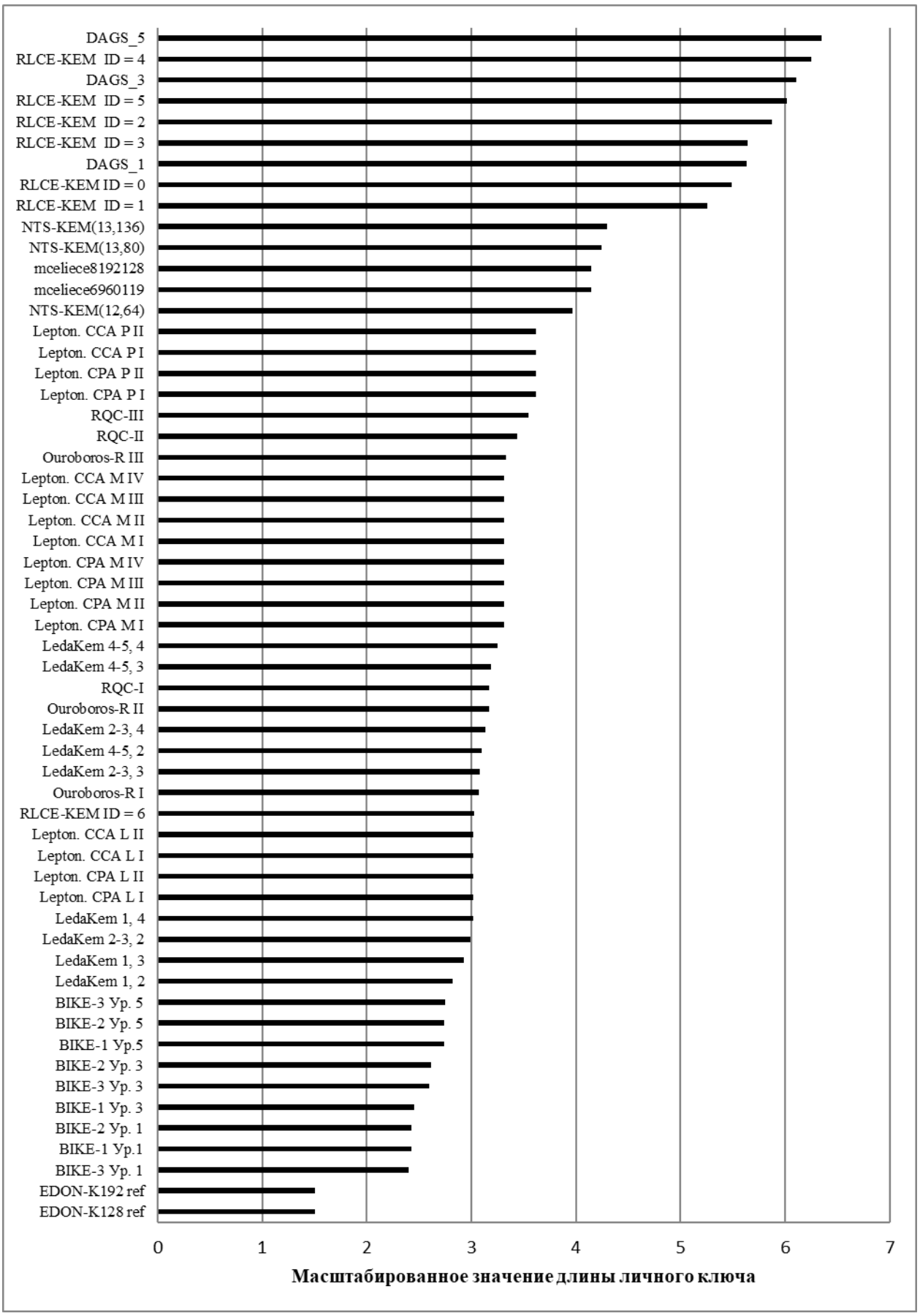


Рис. 1. Гистограмма сравнительного анализа длин личного ключа
(в байтах, логарифмический масштаб) для алгоритмов инкапсуляции ключей

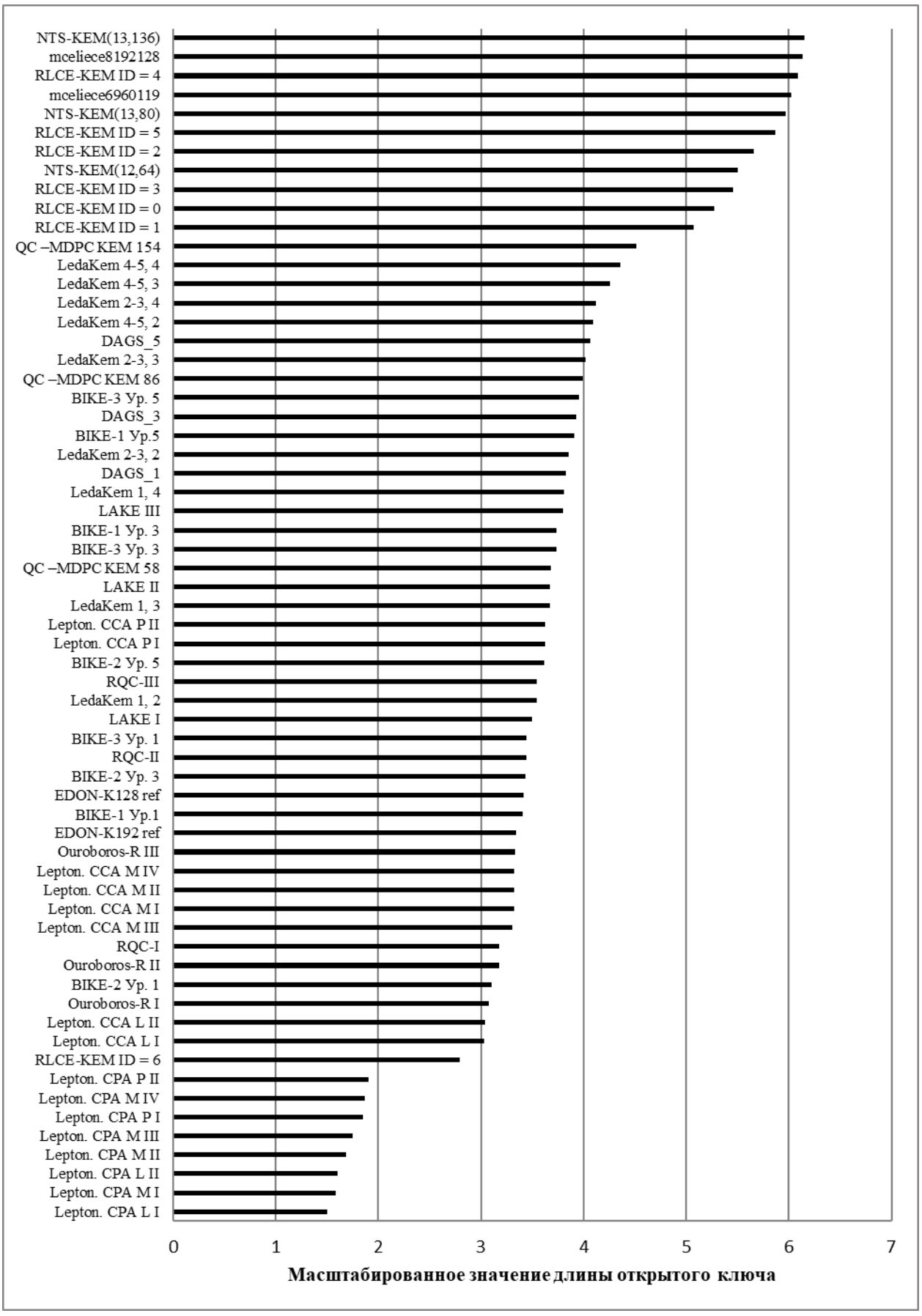


Рис. 2. Гистограмма сравнительного анализа длин открытого ключа
(в байтах, логарифмический масштаб) для алгоритмов инкапсуляции ключей

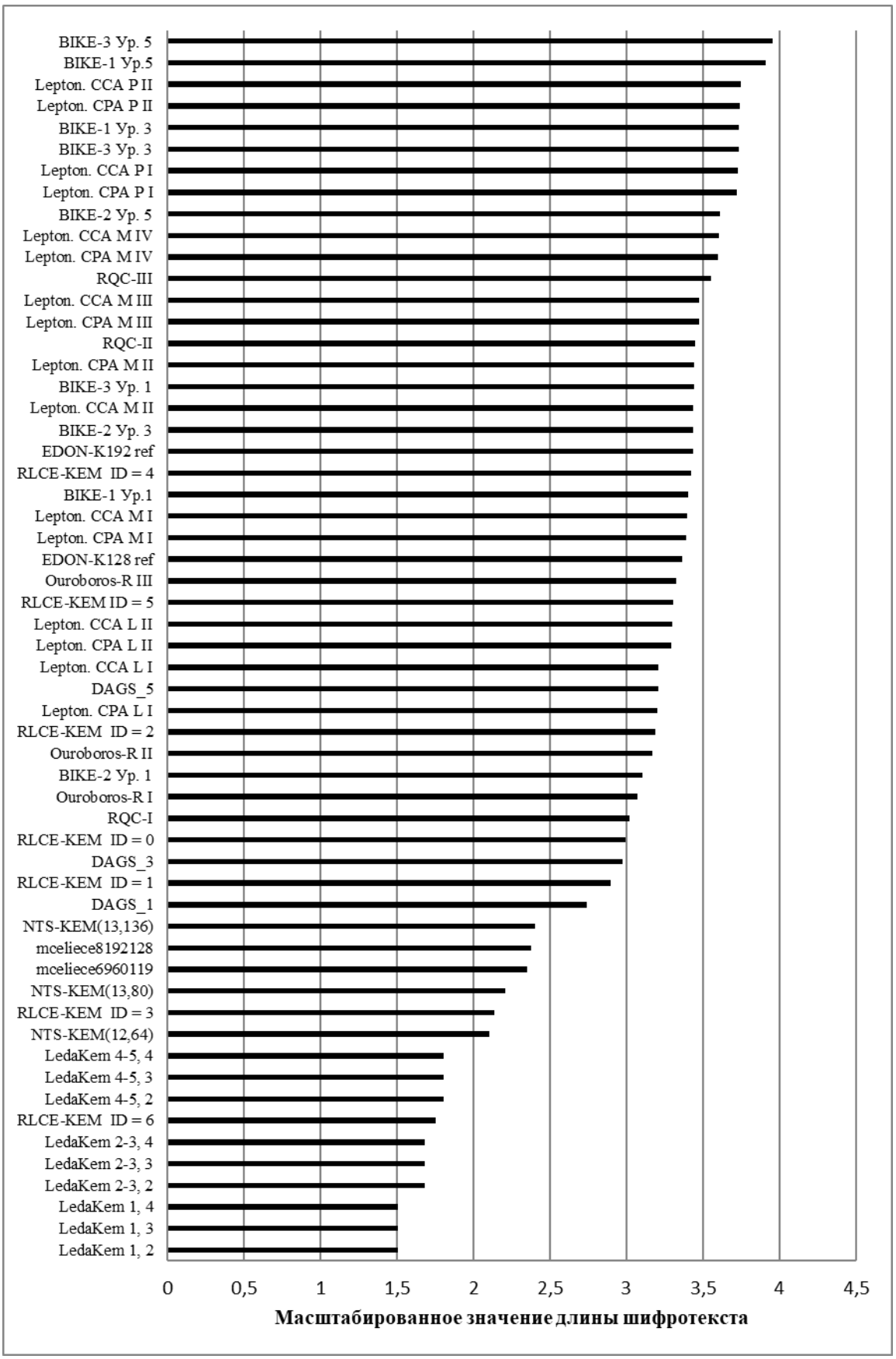


Рис. 3. Гистограмма сравнительного анализа длин шифротекста
(в байтах, логарифмический масштаб) для алгоритмов инкапсуляции ключей

Сравнительный анализ объема памяти, занимаемого открытым ключом в представленных схемах инкапсуляции ключей, дал несколько другой результат. Наименьшая длина открытого ключа у всех вариаций алгоритма Lepton.CPA – от 32 до 80 байт. Затем следуют алгоритмы RLCE-KEM ID = 6, Lepton.CCA, Ouroboros-R I, BIKE-2. Наиболее длинные открытые ключи имеют алгоритмы NTS-KEM (13, 136), Classic McElice 128 и RLCE-KEM ID = 4 – 1419704, 1357824 и 1232001 соответственно.

Для всех алгоритмов длина шифротекста сравнительно невелика и составляет от 32 до 9032,75 байт. По длине получаемого шифротекста алгоритмы инкапсуляции ключей расположились в следующем порядке. Сначала с наименьшей длиной следуют вариации схемы LedaKem и RLCE-KEM ID = 6. Некоторые авторы намеренно старались минимизировать длину шифротекста и при этом сохранить достаточную криптостойкость, чтобы такие алгоритмы инкапсуляции нашли свое применение в системах с ограниченными ресурсами. Логично, что наибольшую длину шифротекста имеют алгоритмы, обеспечивающие, по заявлению их авторов, наивысший уровень криптостойкости. Такими алгоритмами оказались вариации схем BIKE-1 и BIKE-3, Lepton.CPA и Lepton.CCA.

Сравнительный анализ показателей быстродействия

В табл. 2 приведены показатели быстродействия для алгоритмов инкапсуляции ключей. Оценка быстродействия проведена самими авторами на разных вычислительных платформах и представлена в формате количества циклов процессора, затраченных на выполнение операции формирования ключей, инкапсуляции и деинкапсуляции ключей. Все измерения происходили без применения каких-либо технологий оптимизации производительности. Разработчики алгоритма LedaKem эти показателя указали в миллисекундах, затраченных на выполнение операции на конкретной вычислительной платформе. Ввиду того, что все остальные разработчики для измерения быстродействия использовали другую метрику, а именно – циклы процессора, для алгоритма LedaKem эти оценки конвертированы в примерное количество затраченных циклов, определенное из характеристик вычислительной платформы.

Приведенные оценки быстродействия сравнимы при условии игнорирования остальных (помимо приведенных показателей используемого центрального процессора) характеристик используемых вычислительных систем. Такая оценка носит исключительно первичный ознакомительный анализ возможностей производительности перечисленных выше алгоритмов.

Таблица 2

Показатели производительности для алгоритмов инкапсуляции ключей (данные приведены в циклах процессора, которые требуется выполнить для проведения каждой операции)

№	Название	Вычислительная платформа	Версия	Формирование ключевых данных, циклы	Инкапсуляция, циклы	Деинкапсуляция, циклы
1	BIKE	Intel Core i5-6260U, 1.80 ГГц	BIKE-1	730025	689138	2901203
				1709921	1850425	7666855
				2986647	3023816	17483906
			BIKE-2	6383408	281755	2674115
				22205901	710970	7114241
				58806046	1201161	16385956
			BIKE-3	433258	575237	3437956
				1100372	1460866	7732167
				2300332	3257675	18047493
2	Classic McElice	Intel Xeon E3-1220 v3 (Haswell), 3.10 ГГц	mceliece8192128	2000000000	300000	450000
			mceliece6960119	966400	-	17055

№	Название	Вычислительная платформа	Версия	Формирование ключевых данных, циклы	Инкапсуляция, циклы	Деинкапсуляция, циклы
3	DAGS	Intel Core i5-5300U, 2.30 ГГц	DAGS_1	49394032811	20109354	23639371
			DAGS_3	106876216775	26109354	24639371
			DAGS_5	136497712522	49029613	260829051
4	Edon-K	Intel Corei7-7600U, 2.90 Гц	EDON-K128 ref	2500000	576000	28700000
			EDON-K192 ref	2000000	496000	54600000
5	LAKE	Intel Core i7-4700hq, 3.4 ГГц	LAKE I	1580000	300000	1270000
			LAKE II	1740000	310000	2090000
			LAKE III	1790000	350000	2890000
6	Leda Kem	AMD Ryzen 5 1600, 3.2 ГГц,	1, $n_0 = 2$	34,11 мс ≈ 109152000	2,11 мс ≈ 6752000	16,78 мс ≈ 53696000
			1, $n_0 = 3$	16,02 мс ≈ 51264000	2,15 мс ≈ 6880000	21,65 мс ≈ 69280000
			1, $n_0 = 4$	13,41 мс ≈ 42912000	2,42 мс ≈ 7744000	24,31 мс ≈ 77792000
			2-3, $n_0 = 2$	142,71 мс ≈ 456672000	8,11 мс ≈ 25952000	48,23 мс ≈ 154336000
			2-3, $n_0 = 3$	76,74 мс ≈ 245568000	8,79 мс ≈ 28128000	49,15 мс ≈ 157280000
			2-3, $n_0 = 4$	51,93 мс ≈ 166176000	9,46 мс ≈ 30272000	46,16 мс ≈ 147712000
			4-5, $n_0 = 2$	427,38 мс ≈ 1367616000	23,00 мс ≈ 73600000	91,78 мс ≈ 293696000
			4-5, $n_0 = 3$	227,71 мс ≈ 728672000	24,85 мс ≈ 79520000	92,42 мс ≈ 295744000
			4-5, $n_0 = 4$	162,34 мс ≈ 519488000	26,30 мс ≈ 84160000	127,16 мс ≈ 406912000
7	Lepton. CPA	Intel Core-i7 4790, 3.6 ГГц	Light I	33625	78808	33400
			Light II	34912	85347	42462
			Moderate I	48932	117275	45519
			Moderate II	51519	125178	51353
			Moderate III	51508	130057	60289
			Moderate IV	57861	152431	72564
			Paranoid I	96602	237722	97757
			Paranoid II	97884	247932	105200
	Lepton. CCA		Light I	34308	79152	87043
			Light II	34536	86584	100141
			Moderate I	49943	121564	132708
			Moderate II	51658	124426	141988
			Moderate III	52699	130631	151185
			Moderate IV	59450	154473	179520
			Paranoid I	94454	234441	264881
			Paranoid II	94569	244706	282199
8	NTS-KEM	Intel Xeon E5-2667 v2, 3.3 ГГц	NTS-KEM(12,64)	41746373	172463	686087
			NTS-KEM(13,80)	135813837	429301	1300102
			NTS-KEM(13,136)	249939545	574406	2911120
9	Ouroboros-R	Intel Core i7-4770, 3.4 ГГц	Ouroboros-R I	600000	980000	1780000
			Ouroboros-R II	650000	1120000	3260000
			Ouroboros-R III	820000	1390000	4730000

№	Название	Вычислительная платформа	Версия	Формирование ключевых данных, циклы	Инкапсуляция, циклы	Деинкапсуляция, циклы
10	QC –MDPC KEM	Intel Core i7-7500U, 2.7 ГГц	QC –MDPC KEM 154	131038872	20263392	229002269
11	RLCE-KEM	Intel Core i7, 2.9 ГГц	ID = 0	1011071617	1805010	4646941
			ID = 1	465481183	1040629	3589491
			ID = 2	3829675407	3331234	8668186
			ID = 3	1962533052	2361787	7160709
			ID = 4	9612380645	8184051	36705481
			ID = 5	5057459034	5362174	24174369
12	RQC	Intel Core i7-4770, 3.4 ГГц	RQC-I	790000	1970000	5300000
			RQC-II	1760000	5600000	14460000
			RQC-III	2820000	5460000	18000000

Для наглядности на рис. 4 – 7 приведены гистограммы параметров быстродействия для вариаций алгоритмов, обеспечивающих наибольший уровень криптостойкости, а именно: Lepton.CCA P II, Lepton.CPA P II, Ouroboros-R III, LAKE III, EDON-K192, BIKE-3, RQC III, QC –MDPC KEM, NTS-KEM (13, 136), LedaKem 4-5, 4, Classic McElice 128, RLCE-KEM, ID = 5, DAGS_5. Для того чтобы адекватно продемонстрировать эти показатели, все данные на гистограммах приведены с использованием логарифмического масштаба, так как параметры разнятся на несколько порядков.

Следует отметить, что меньшие значения циклов, затрачиваемых на выполнение одной операции, являются предпочтительными. В то же время, бо́льшие значения количества затрачиваемых циклов указывают на низкую скорость выполнения операции.

На рис. 4 приведена сводная гистограмма всех показателей быстродействия, показывающая общее соотношение скорости выполнения трех операций (формирование ключей, инкапсуляция и деинкапсуляция ключей) для каждого из алгоритмов. Все данные указаны в затраченных на выполнение операции циклах.

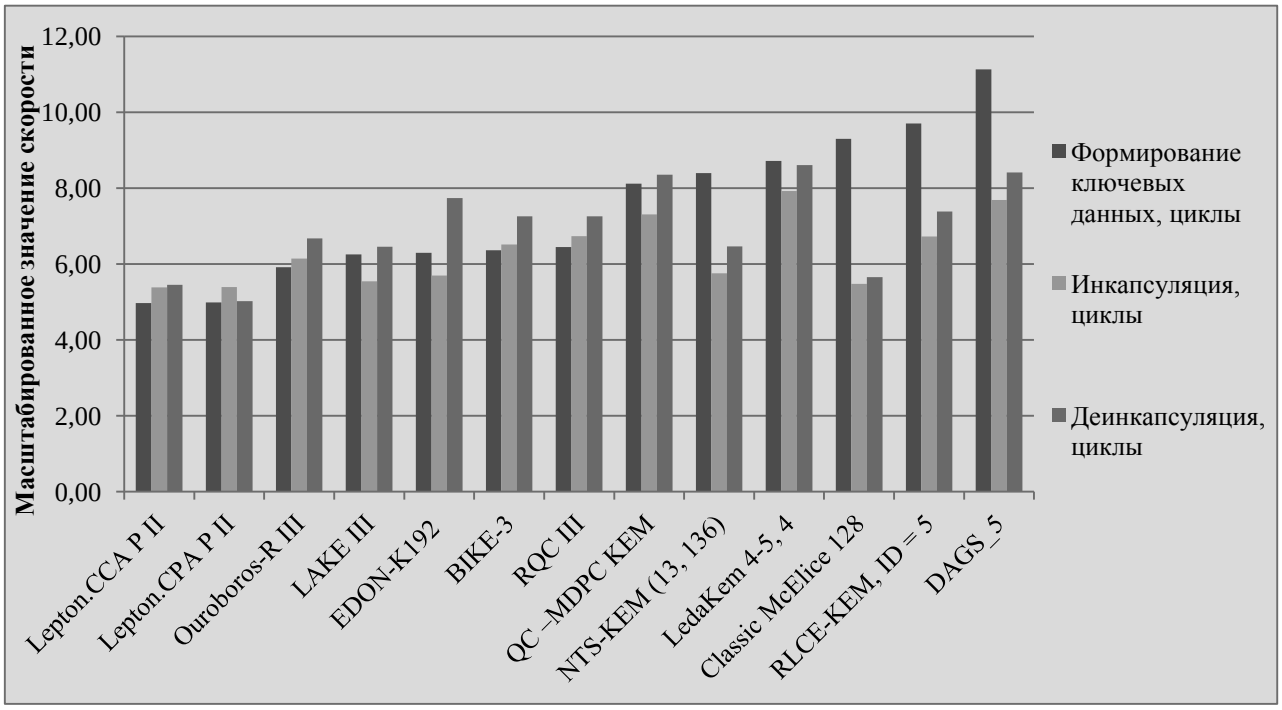


Рис. 4. Гистограмма показателей быстродействия (в логарифмическом масштабе)

Примерно сравнимую скорость выполнения всех операций имеют алгоритмы Lepton.CCA и Lepton.CPA, Ouroboros-R, LAKE, LedaKem. Достаточно большой разрыв в производительности между формированием ключей и инкапсуляцией (деинкапсуляцией) имеют схемы EDON-K, Classic McElic, RLCE-KEM, DAGS_5.

На рис. 5 приведена гистограмма оценки скорости формирования ключевых данных. Как показано выше, алгоритм Lepton имеет достаточно небольшие длины как открытого, так и личного ключей, и благодаря используемому алгоритму скорость формирования ключевых данных для данной схемы наибольшая. Наименьшая скорость формирования у алгоритма DAGS_5.

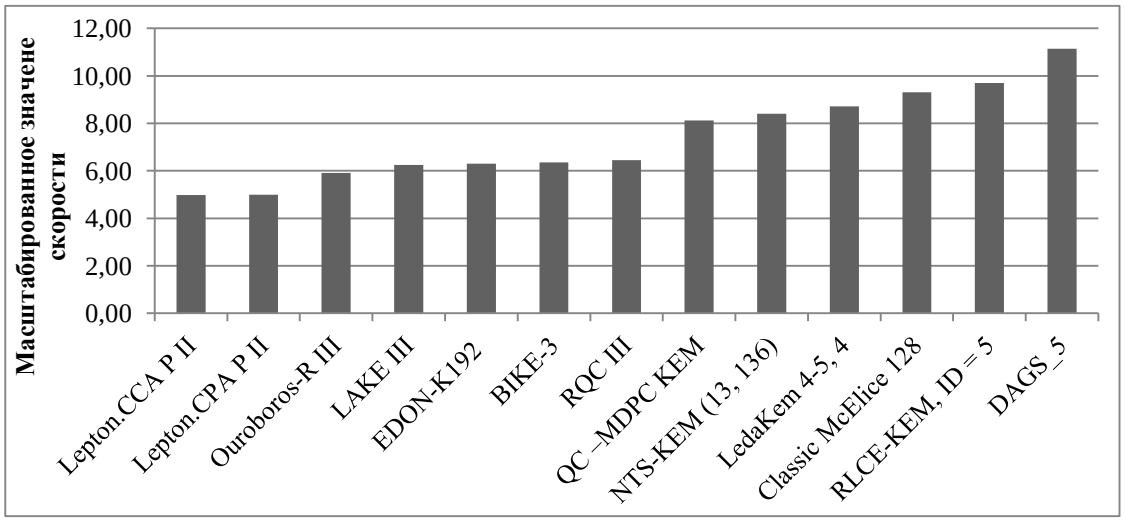


Рис. 5. Гистограмма показателя быстродействия: скорость формирования ключевых данных, в циклах (в логарифмическом масштабе)

Оценка скорости инкапсуляции приведена на рис. 6. Опять наибольшей скоростью обладает алгоритм Lepton. Наименьшая скорость у схемы LedaKem.

Показатель быстродействия – скорость деинкапсуляции напрямую зависит от выбранного метода для деинкапсуляции. Некоторые авторы указывают, что для выполнения этой операции в алгоритме может использоваться несколько различных методик, от выбора которых будет зависеть производительность операции. Так или иначе, согласно результатам оценки наибольшая скорость деинкапсуляции у алгоритма Lepton, наименьшая – у LedaKem.

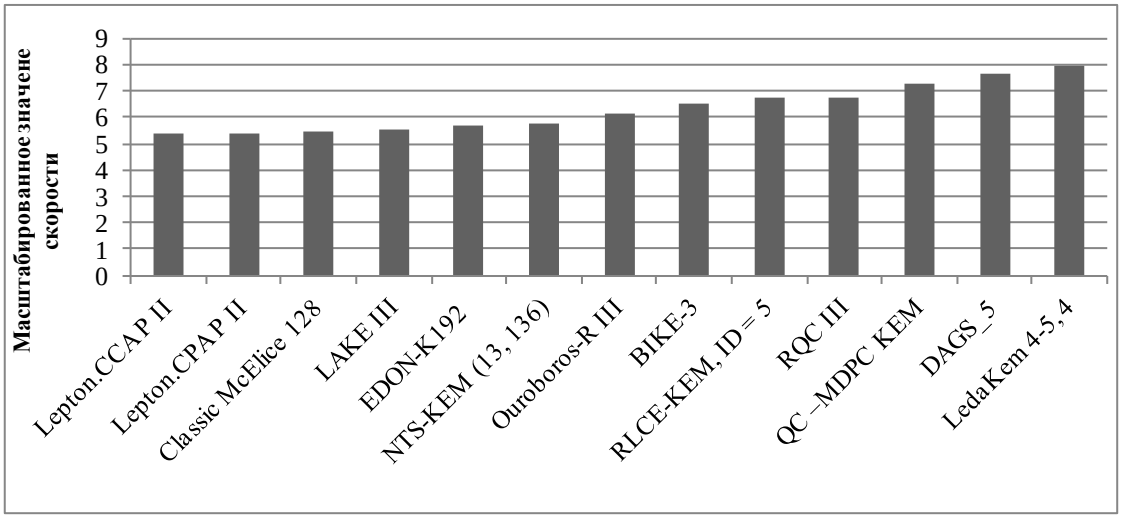


Рис. 6. Гистограмма показателя быстродействия: скорость инкапсуляции, в циклах (в логарифмическом масштабе)

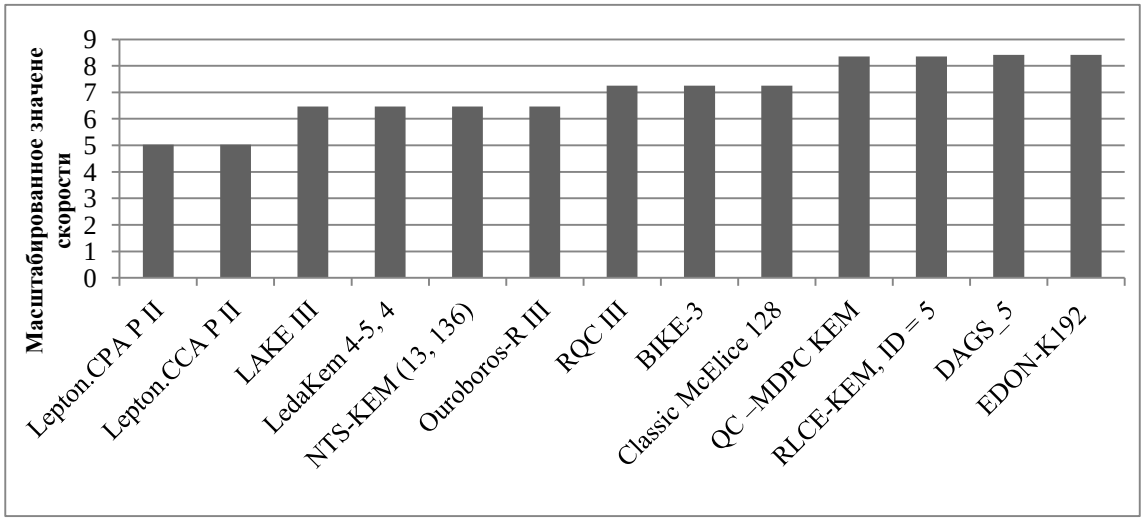


Рис. 7. Гистограмма показателя быстродействия: скорость деинкапсуляции, в циклах (в логарифмическом масштабе)

Выводы

Проведен первичный анализ схем-конкурсантов, представленных на конкурс постквантовой криптографии NIST PQC. Рассмотрены все 12 схем инкапсуляции ключей, проведены сравнения по показателям (указанными разработчиками) входных и выходных параметров, а также по показателям криптографической стойкости и быстродействия. На данный момент оценка криптостойкости и быстродействия взята из данных, указанных разработчиками.

В ходе исследований установлено, что практически все схемы удовлетворяют формальным требованиям к кандидатам на постквантовые схемы инкапсуляции ключей, т.е. имеют различные варианты алгоритмов, которые обеспечивают все три уровня криптостойкости (1-й, 3-й и 5-й). Исключение составляет алгоритм Edon-K (он обеспечивает только 1-й и 3-й уровни стойкости).

Наилучшие показатели быстродействия показал алгоритм Lepton. Однако следует отметить, что оценки быстродействия приводятся авторами для эталонных реализаций алгоритмов. В дальнейшем будут представлены оптимизированные реализации данных схем инкапсуляции ключей, их исследование является перспективным направлением.

Список литературы:

1. Post-Quantum Cryptography, Round 1 Submissions, 2017. [On-line]. Internet: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-1-Submissions.

2. Nicolas Aragon, Paulo Barreto, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Phillipe Gaborit, Shay Gueron, Tim Guneysu, Carlos Aguilar Melchor, Rafael Misoczki, Edoardo Persichetti, Nicolas Sendrier, Jean-Pierre Tillich, Gilles Zemor. BIKE – Bit Flipping Key Encapsulation, NIST Submission, 2017. [On-line]. Internet: http://bikesuite.org/#spec.

3. Daniel J. Bernstein, Tung Chou, Tanja Lange, Ingo von Maurich, Rafael Misoczki, Ruben Niederhagen, Edoardo Persichetti, Christiane Peters, Peter Schwabe, Nicolas Sendrier, Jakub Szefer. Classic McEliece, NIST Submission, 2017. [On-line]. Internet: https://classic.mceliece.org/index.html.

4. Gustavo Banegas, Paolo S.L M. Barreto, Brice Odilon Boidje, Pierre-Louis Cayrel, Gilbert Ndollane Dione, Kris Gaj, Cheikh Thiecoumba Gueye, Richard Haeussler, Jean Belo Klamti, Ousmane N'diaye, Duc Tri Nguyen. DAGS: Key Encapsulation using Dyadic GS Codes. NIST Submission, 2017. [On-line]. Internet: https://www.dags-project.org/#files.

5. Danilo Gligoroski, Kristian Gjøsteen. Post-quantum Key Encapsulation Mechanism EDON-K, NIST Submission, 2017. [On-line]. Internet: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-1-Submissions.

6. Nicolas Aragon, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Olivier Ruatta, Jean-Pierre Tillich, Gilles Zémor. LAKE – Low rAnk parity check codes Key Exchange, NIST Submission, 2017. [On-line]. Internet: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-1-Submissions.

7. Marco Baldi, Alessandro Barenghi, Franco Chiaraluce, Gerardo Pelosi, Paolo Santini. LEDAkem (Low dEnsity coDe-bAsed key encapsulation mechanism), NIST Submission, 2017. [On-line]. Internet: https://www.ledacrypt.org/LEDAkem/.

8. Yu Yu, Jiang Zhang. Lepton: Key Encapsulation Mechanisms from a variant of Learning Parity with Noise, NIST Submission, 2017. [On-line]. Internet: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-1-Submissions.

9. Martin Albrecht, Carlos Cid, Kenneth G. Paterson, Cen Jung Tjhai, Martin Tomlinson. NTS-KEM, NIST Submission, 2017. [On-line]. Internet: https://nts-kem.io/.

10. Carlos Aguilar Melchor, Jean-Christophe Deneuville, Nicolas Aragon, Philippe Gaborit, Slim Bettaieb, Adrien Hauteville, Loïc Bidoux, Gilles Zémor . Ouroboros-R, NIST Submission, 2017. [On-line]. Internet: http://pqc-ouroborosr.org/.

11. Atsushi Yamada, Edward Eaton, Kassem Kalach, Philip Lafrance, Alex Parent. QC-MDPC KEM: A Key Encapsulation Mechanism Based on the QC-MDPC McEliece Encryption Scheme, NIST Submission, 2017. [On-line]. Internet: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-1-Submissions.

12. Yongge Wang. RLCEKeyEncapsulation Mechanism (RLCE-KEM) Specifcation, NIST Submission, 2017. [On-line]. Internet: http://quantumca.org/.

13. Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Phillippe Gaborit, Gilles Zemor. Rank Quasi-Cyclic (RQC) , NIST Submission, 2017. [On-line]. Internet: http://pqc-rqc.org/.

14. Katz, Jonathan; Lindell, Yehuda. Introduction to Modern Cryptography: Principles and Protocols. Chapman & Hall / CRC Press, 2007. 553 p.

15. Bellare, Mihir; Rogaway, Phillip. "Introduction to Modern Cryptography. [On-line]. Internet: http://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf, September 21, 2005.

Харківський національний
університет імені В.Н.Каразіна

Надійшла до редколегії 00.00.2018

УДК 004.056.55

М.О. ПОЛУЯНЕНКО, О.В. ПОТІЙ, д-р техн. наук

ДОСЛІДЖЕННЯ РЕГІСТРІВ ЗСУВУ З НЕЛІНІЙНИМИ ЗВОРОТНИМИ ЗВ'ЯЗКАМИ В ЯКОСТІ КОМБІНУЮЧИХ ТА ФІЛЬТРУЮЧИХ ФУНКЦІЙ

Вступ

Розглянемо загальну структуру схему комбінуючого генератора (рис. 1, *а*) та фільтруючого генератора (рис. 1, *б*) ПВП із застосуванням декількох регістрів зсуву з лінійними зворотними зв'язками (РЗЛЗЗ) або регістрів зсуву з нелінійними зворотними зв'язками (РЗНЗЗ) – $РЗ_i\ (i=1,\ldots,L)$. В даному випадку f розглядається як комбінуюча або фільтруюча функція від L змінних.

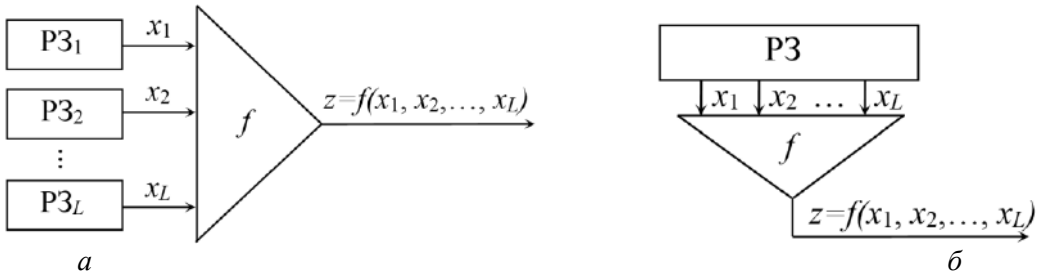


Рис. 1. Структурна схема комбінуючого (*а*) та фільтруючого (*б*) генератора ПВП

Булевою функцією, що відповідає РЗНЗЗ, в загальному виді називається булеве відображення виду $f: GF_2^L \to GF_2$. Булеві функції будемо представляти у вигляді многочленів (поліном Жегалкіна або алгебраїчна нормальна форма – АНФ) над полем $GF(2)$:

$$f(x_1, x_2, \ldots, x_L) = a_0 + \sum_{i=1}^{L} a_i x_i + \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} a_{ij} x_i x_j + \sum_{i=1}^{L-2} \sum_{j=i+1}^{L-1} \sum_{m=j+1}^{L} a_{ijm} x_i x_j x_m + \ldots .$$

Алгебраїчним степенем $def(f)$ функції f називається кількість змінних у самому довгому доданку АНФ, при якому коефіцієнт не дорівнює нулю. Функція степеню не вище 1 є *афінною функцією*. Випадку, коли у афінної функції $a_0 = 0$, відповідає *лінійна функція*. Множина афінних булевих функцій від L змінних позначається як A_L.

В даній роботі розглядаються РЗНЗЗ другого порядку, тобто ті, алгебраїчний степінь яких дорівнює $def(f) = 2$. Крім того, будемо досліджувати лише ті РЗНЗЗ другого порядку, які формують послідовність максимального періоду, тобто М-послідовність. Позначимо такі нелінійні регістри як М-РЗНЗЗ.

Постановка задачі

Розглянемо деякі з основних, у даному випадку, показників оцінки криптографічної стійкості:

– *Збалансованість*.

Булева функція f від L змінних називається збалансованою, якщо функція приймає значення 0 та 1 однаково часто. Це одне з найбільш природних необхідних властивостей, що накладаються на булеві функції, що використовуються в поточних шифрах [0].

Якщо булева функція збалансована, то ймовірність того, що вона прийме значення 0 або 1, однакова та дорівнює $1/2$. Це дозволяє послабити статистичні залежності між входом

функції та її виходом. В іншому випадку у криптоаналитика є можливість, використовуючи розподіл усіх співвідношень, провести криптоаналіз шифру.

– *Наявність заборон.*

В разі аналізу ПВП, що генерується за допомогою фільтруючого генератору, виникає ще одне поняття – заборона булевої функції, тобто наявність комбінацій вихідної послідовності, яка не може мати місце не за яких комбінацій вхідної послідовності.

Інтуїтивно зрозуміло, що наявність заборони у фільтруючої функції генератора робить її «слабкіше», ця заборона ніколи не з'явитися у вихідній послідовності генератора, що погіршує його статистичні властивості.

– *Кореляційна імунність.*

Вимога кореляційної імунної функції пов'язана з протистоянням кореляційній атаці, ідея якої полягає в наступному [10]. Розглянемо комбінуючий генератор ПВП (рис. 1, *а*). Ключем генератора є початковий стан всіх регістрів. Обсяг ключа дорівнює $2^{l_1+\ldots+l_L}$, де l_i – довжина РЗ_i для $i=1,\ldots,L$.

Кожний з РЗ_i генерує послідовність $x_i = x_i^1 x_i^2 \ldots$, як правило, близьку за своїми властивостями до випадкової. Зокрема, при досить великій довжині послідовності для випадково вибраного її біта x_i^j має місце ймовірність випадкової події $x_i^j = 0$: $\text{P}[x_i^j = 0] \approx 1/2$. Отже, якщо $y = y^1 y^2 \ldots$ – довільна послідовність, яка не залежить від x_i, то

$$\text{P}[x_i^j = y^j] = \text{P}[x_i^j = 0]\cdot \text{P}[y^j = 0] + \text{P}[x_i^j = 1]\cdot \text{P}[y^j = 1] \approx \frac{1}{2}\left(\text{P}[y^j = 0] + \text{P}[y^j = 1]\right) = \frac{1}{2}.$$

Припустимо, що $\text{P}[f = x_1] \neq 1/2$ (у цьому випадку говорять, що *функція* f *корелює зі змінною* x_1). За допомогою кореляційної атаки знайдемо початковий стан s_1 РЗ_1. Для цього будемо перебирати всі можливі 2^{l_1} станів РЗ_1, для кожного з них будуємо послідовність $z' = z'_1 z'_2 \ldots$ та підраховуємо кількість збігів з ПВП $z'_i = z_i$. Для всіх послідовностей, крім однієї (що генерується s_1), частка збігів буде $\approx 1/2$. Тим самим визначимо, що частина ключа – стан s_1. Якщо функція f має кореляцію з усіма своїми змінними (або з усіма, крім однієї – тоді стан регістра, відповідного цієї змінної, знайдемо останнім, знаючи стан всіх інших регістрів), то знайдемо ключ генератора за $2^{l_1} + \ldots + 2^{l_L}$ випробувань, що набагато менше складності атаки грубої сили.

– *Нелінійність.*

Практика показує [10], що криптографічні перетворення, які мають властивості, близькі до властивостей лінійних функцій, в багатьох випадках призводять до істотного зниження стійкості шифрів. З цієї причини в криптографії важливе значення мають функції, властивості яких виключають слабкості, властиві функціям, близьким до лінійних. Таким чином, бажаною якістю функції є її нелінійність, що розуміється в широкому сенсі: як заперечення лінійності. У блокових та потокових шифрах застосування функції з високою нелінійністю сприяє підвищенню стійкості шифрів к лінійному та диференціальному методам криптоаналізу.

У літературі мало описується зв'язок між різними криптографічними властивостями. Практика показує [10], що в якості компонент шифру необхідно вибирати «хороші з усіх боків» функції, що є насправді дуже непростим завданням, оскільки багато властивостей суперечать одна одній. Хоча теоретичні результати показують, що у випадкової функції багато криптографічних параметрів, близьких до оптимальних. Питання в тому, як її вибрати, випадкову?

Отримані результати

Введемо деякі визначення, що будемо використовувати у подальшому [0].

Вагою булевої функції або *вагою Хеммінга* називається кількість одиниць у векторі та позначається як $wt(f)$ або $wt(x)$.

Відстанню Хемінга між булевими функціями f та g є відстань Хеммінга між векторами їх значень $dist(f,g)=wt(f\oplus g)$.

Твердження та теореми з метою скорочення обсягу роботи надано без доведення. Доведення є загальнодоступними та наведені, наприклад, у [1 – 8]. Всі значення експериментально перевірені на всій множині РЗНЗЗ (РЗЛЗЗ як окремий випадок) для розмірів з $L=4$ до 9 комірок включно.

Збалансованість

М-РЗНЗЗ, як і М-РЗЛЗЗ, генерують модифіковану послідовність де Брейна і якщо додати до розгляду стан заповнення усіх комірок нульовими значеннями, то отримана функція буде збалансованою. При рівноімовірному і незалежному виборі аргументів булевої функції f, імовірності її значень відповідно $P(1)=wt(f)/2^L$ $P(0)=1-wt(f)/2^L$.

Наявність заборон

М-РЗНЗЗ є функціями, які не мають заборон. Це випливає з того, що РЗНЗЗ формують послідовність де Брейна, яка за визначенням має всі можливі комбінації послідовності.

Однак слід бути обережними, оскільки цілком збалансована фільтруюча функція в тому чи іншому вигляді переносить властивості вхідної послідовності до властивості послідовності, що генерується [10]. Наприклад, в роботі [0] встановлено новий критерій, який ідейно говорить наступне: «фільтруюча функція зберігає заборони (у відповідному сенсі) тоді і тільки тоді, коли вона цілком збалансована». Відповідно, якщо на вхід функції надходить «далека» від випадкової послідовність, то й на виході її статистичні властивості будуть погані.

Кореляційна імунність

Наявність кореляційно імунної функції степеня m означає, що значення функції $Z=f(X)$ статистично незалежні від будь-якого набору з не більше ніж m компонентів довільного вектора-аргументу $X=GF(2)^L$. Це рівнозначно умові, що на вихід перетворення не «просочується» інформація про вектори, що надходять на вхід перетворення і мають вагу Хеммінга не більше m.

Булева функція f називається *кореляційно імунною порядку* m, $1\le m\le L$, якщо для будь-якої сукупності номерів m змінних $1\le i_1<i_2<\ldots<i_m\le L$ випадкові величини $X=(x_{i_1},x_{i_2},\ldots,x_{i_m})$ та $Y=f(x_1,x_2,\ldots,x_L)$ є незалежними.

Можна довести, що кореляційно імунна порядку m функція від L змінних є кореляційно імунною довільного меншого порядку. Таким чином, булевій функції f відповідає деякий максимальний порядок її кореляційної імунності $m_{\max}$, який позначається через $cor(f)$.

Випадок, коли $m=L$, має місце лише коли $f=const$. Максимального кореляційного імунітету степеня $m=L-1$ досягають тільки афінні функції, тобто криптографічно слабкі. Крім того, якщо f збалансована та $cor(f)=L-2$, то функція f також афінна. Таким чином, є сенс розглядати порядок кореляційної імунності m лише у діапазоні $1\le m\le L-3$.

Збалансована кореляційно-імунна функція порядку m називається *m-стійкою функцією*. Формально будь-яку збалансовану булеву функцію можна розглядати як 0-стійку і

довільну булеву функцію як (-1) стійку. За аналогією з $cor(f)$ вводиться позначення для *максимального порядку стійкості*:

$$sut(f)=\begin{cases}-1, & \text{якщо } f \text{ не збалансована},\\ cor(f), & \text{якщо } f \text{ збалансована}.\end{cases}$$

Таблиця 1

Розподіл кількості регістрів в залежності від максимальної стійкості для М-РЗЛЗЗ та М-РЗНЗЗ другого порядку

	Усього	$sut(f)$							
		$m=0$	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$	$m=6$	$m=7$
$L=4$									
М-РЗЛЗЗ	2	0	2	0	–	–	–	–	–
М-РЗНЗЗ 2-го порядку	14	4	10 *m-опт*	–	–	–	–	–	–
$L=5$									
М-РЗЛЗЗ	6	0	2	0	4 *m-опт*	–	–	–	–
М-РЗНЗЗ 2-го порядку	122	64	52	6 *m-опт*	–	–	–	–	–
$L=6$									
М-РЗЛЗЗ	6	0	2	0	4	0	–	–	–
М-РЗНЗЗ 2-го порядку	1 946	788	1 044	76	38 *m-опт*	–	–	–	–
$L=7$									
М-РЗЛЗЗ	18	0	4	0	10	0	4 *m-опт*	–	–
М-РЗНЗЗ 2-го порядку	64 038	33 988	25 578	4 090	378	4 *m-опт*	–	–	–
$L=8$									
М-РЗЛЗЗ	16	0	0	0	12	0	4	0	–
М-РЗНЗЗ 2-го порядку	4 017 982	1 686 218	2 120 124	194 798	16 612	188	42 *m-опт*	–	–
$L=9$									
М-РЗЛЗЗ	48	0	2	0	16	0	28	0	2 *m-опт*
М-РЗНЗЗ 2-го порядку	519 239 746	284 956 836	208 843 948	24 325 344	1091568	21 192	848	10 *m-опт*	–

Нерівність Зігенталера. Якщо f – кореляційно-імунна порядку m функція на $GF(2)^L$, то:

1) $def(f)\le L-m$;

2) якщо f є збалансованою та $sut(f)=m\le L-2$, то $def(f)+sut(f)\le L-1$.

Нерівність Зігенталєра є одним з багатьох протиріч криптографічних властивостей функцій один одному: високий порядок кореляційної імунної функції тягне її низьку алгебраїчну степінь і навпаки.

Якщо функція f є збалансована, $sut(f)=m\le L-2$ та $def(f)=L-m-1$, то f називається m*-оптимальною*.

Звідки маємо m-оптимальні f для РЗЛЗЗ $m=L-1-def(f)=L-2$ та для РЗНЗЗ другого порядку $m=L-1-def(f)=L-3$. Таким чином, ми визначили верхню межу значень для m-стійких функцій. У роботі було досліджено кореляційну імунність усієї множини М-РЗЛЗЗ та М-РЗНЗЗ другого порядку розмірністю $L=4,\ldots,9$. Результати наведені у таблиці 1.

Як бачимо з табл. 1, М-РЗНЗЗ другого порядку досягають значення для m-оптимальних функцій (у таблиці позначені як «m*-опт*») при всіх досліджених L. Однак, є дуже велика частка (приблизно половина усієї множини М-РЗНЗЗ другого порядку), яка не має кореляційної імунності.

Нелінійність

Нелінійністю функції f називається відстань від f до класу афінних функцій. Будемо позначати нелінійність функції f через N_f:

$$N_f = dist(f, A_L) = \min_{g\in A_L} dist(f, g).$$

У випадку парного L максимально можливе значення нелінійності дорівнює $2^{L-1}-2^{(L/2)-1}$, функції, які мають таку нелінійність, виділені в окремий клас – «бент-функції». У разі непарного L точне значення максимальної нелінійності невідомо і представляє складне комбінаторне завдання [10]. Разом з тим, всі бент-функції не збалансовані (на відміну від М-РЗНЗЗ), що робить їх уразливими до статистичного аналізу.

Наступні твердження показують, що чим вище порядок кореляційної імунної функції, тим нижче верхня межа її нелінійності.

Якщо f збалансована і m-стійка, $m\le L-2$. Тоді $N_f \le 2^{L-1}-2^{m+1}$.

За аналогією з поняттям m-оптимальної функції вводиться спеціальна назва для m-стійкої функції максимально можливої нелінійності.

Якщо функція f з $GF(2)^L$ збалансована, $sut(f)=m\le L-2$ і $N_f = 2^{L-1}-2^{m+1}$, то f називається m*-насиченою*.

У табл. 2 наведено розраховані значення за вищенаведеними формулами максимально можливої нелінійності збалансованої функції в залежності від її стійкості.

Таблиця 2

Значення нелінійності m-насичених функцій в залежності від їх максимальної стійкості

		$sut(f)$					
		1	**2**	**3**	**4**	**5**	**6**
N_f	$L=4$	4	0	–	–	–	–
	$L=5$	12	8	0	–	–	–
	$L=6$	28	24	16	0	–	–
	$L=7$	60	56	48	32	0	–
	$L=8$	124	120	112	96	64	0
	$L=9$	252	248	240	224	192	128

Значення нелінійності, наведені у табл. 2, не обов'язково досяжні. Позначимо через $N_{f\max}(L,m)$ *максимально можливу нелінійність* m-стійкої булевої функції, заданої на $GF(2)^L$, та наведемо верхню оцінку для нелінійності m-стійких функцій.

З наведеного випливає, що $N_{f\max}(L,-1)=2^{L-1}-2^{L/2-1}$, це значення може досягатися тільки для парних L. Якщо f є збалансованою функцією та L парне значення, справедливо $N_{f\max}(L,m)=2^{L-1}-2^{L/2-1}-2^{m+1}$ [2].

Таблиця 3

Розподіл кількості регістрів в залежності від нелінійності та максимальної стійкості для М-РЗН33 другого порядку

N_f	$sut(f)$						
	$m=0$	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$	$m=6$
$L=4$							
4	4[1)]	m -нас 10	–	–	–	–	–
0	0	0	0	–	–	–	–
$L=5$							
12	56[1)]	0	–	–	–	–	–
8	8	52	m -нас 6	–	–	–	–
0	0	0	0	0	–	–	–
$L=6$							
24	740	856[1)]	0	–	–	–	–
16	48	188	76	m -нас 38	–	–	–
0	0	0	0	0	0	–	–
$L=7$							
56	26 324[1)]	0	0	–	–	–	–
48	7 624	24 862	3 596	0	–	–	–
32	40	716	494	378	m -нас 4	–	–
0	0	0	0	0	0	0	–
$L=8$							
112	1 620 992	1 737 690	0	0	–	–	–
96	65 078	380 856	192 572	14 270	0	–	–
64	148	1 578	2 226	2 342	188	m -нас 42	–
0	0	0	0	0	0	0	0
$L=9$							
240	216 743 896	0	0	0	–	–	–
224	67 714 544	203 967 024	19 364 756	0	0	–	–
192	498 196	4 872 526	4 953 980	1 079 370	18 642	0	–
128	200	4398	6 608	12 198	2 550	848	m -нас 10
0	0	0	0	0	0	0	0

[1)] – значення N_f є максимальними для даних m, L та відповідають $N_{f\max}(L,m)$, які зазначені у [11].

У [11] вказується, що для непарних L та $L\le 7$, $N_{f\max}(L,-1)=2^{L-1}-2^{(L-1)/2}$, але для непарних L та $L\ge 15$ справедлива нерівність $N_{f\max}(L,-1)>2^{L-1}-2^{(L-1)/2}$.

При $m\ge L-2$, за нерівністю Зігенталєра $def(f)\le 1$, звідки $N_{f\max}(L,m)=0$. Також у [11] є посилання на доведену нерівність $N_{f\max}(L,L-3)=2^{L-2}$ та гіпотезу, що $N_{f\max}(L,L-4)=2^{L-1}-2^{L-3}$. Крім того, наведено деякі точні значення $N_{f\max}(L,m)$ для малих L та m:

$N_{f\max}(4,0)=4$;

$N_{f\max}(5,-1)=N_{f\max}(5,0)=N_{f\max}(5,1)=12$;

$N_{f\max}(6,0) = 26$; $N_{f\max}(6,1) = N_{f\max}(6,2) = 24$;

$N_{f\max}(7,-1) = N_{f\max}(7,0) = N_{f\max}(7,1) = 56$.

Вказані результати не суперечать результатам, отриманим в даній роботі і наведеним нижче.

У табл. 3 зведено отриманий розподіл за нелінійністю усієї множини М-РЗНЗЗ розмірністю $L = 4,\ldots,9$ в залежності від розміру РЗНЗЗ та кореляційної імунності.

Як бачимо з наведених результатів, М-РЗНЗЗ другого порядку одночасно досягають максимально можливої стійкості та максимальної нелінійності. Причому, всі m-оптимальні функції також є і m-насиченими (у табл. 3 позначені як «m-нас»). Крім того, багато М-РЗНЗЗ, які не є m-насиченими функціями за визначенням, досягають максимально можливого результату для $N_{f\max}(L,m)$ наведеного вище.

В якості прикладу наведемо М-РЗНЗЗ другого порядку розмірністю $L = 9$, що відповідають m-насіченим функціям (з нелінійністю $N_f = 128$ та максимальною стійкістю $sut(f) = 6$):

$$f(x_1, x_2, \ldots, x_9) = x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_9 + x_2 \cdot x_5 + x_2 \cdot x_8$$
$$f(x_1, x_2, \ldots, x_9) = x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_1 \cdot x_7 + x_4 \cdot x_7$$
$$f(x_1, x_2, \ldots, x_9) = x_1 + x_2 + x_3 + x_4 + x_5 + x_7 + x_8 + x_9 + x_4 \cdot x_6 + x_4 \cdot x_8$$
$$f(x_1, x_2, \ldots, x_9) = x_1 + x_2 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_1 \cdot x_5 + x_3 \cdot x_5$$
$$f(x_1, x_2, \ldots, x_9) = x_1 + x_2 + x_3 + x_4 + x_6 + x_7 + x_8 + x_9 + x_3 \cdot x_5 + x_3 \cdot x_6$$
$$f(x_1, x_2, \ldots, x_9) = x_1 + x_2 + x_3 + x_5 + x_6 + x_7 + x_8 + x_9 + x_3 \cdot x_6 + x_4 \cdot x_6$$
$$f(x_1, x_2, \ldots, x_9) = x_1 + x_2 + x_3 + x_4 + x_6 + x_7 + x_8 + x_9 + x_1 \cdot x_6 + x_5 \cdot x_6$$
$$f(x_1, x_2, \ldots, x_9) = x_1 + x_2 + x_3 + x_5 + x_6 + x_7 + x_8 + x_9 + x_3 \cdot x_4 + x_3 \cdot x_8$$
$$f(x_1, x_2, \ldots, x_9) = x_1 + x_2 + x_3 + x_4 + x_6 + x_7 + x_8 + x_9 + x_2 \cdot x_7 + x_5 \cdot x_7$$
$$f(x_1, x_2, \ldots, x_9) = x_1 + x_2 + x_3 + x_5 + x_6 + x_7 + x_8 + x_9 + x_2 \cdot x_4 + x_2 \cdot x_7$$

Аналізуючи отримані результати, бачимо, що симетричні М-РЗНЗЗ мають однакові показники $sut(f)$ та N_f. Всі М-РЗНЗЗ, що досліджувались, мають $N_f \geq 2^{L-2}$.

Висновки

Булеві функції, які відповідають М-РЗНЗЗ, є збалансованими, не мають заборон.

Близько половини булевих функцій, що відповідають М-РЗНЗЗ, є кореляційно імунними функціями, деякі з яких досягають максимальної можливої стійкості та є m-оптимальними функціями.

Всі булеві функції, що відповідають М-РЗНЗЗ другого порядку, за виключенням М-РЗЛЗЗ, мають нелінійність $N_f \geq 2^{L-2}$ та деякі досягають максимального значення, тобто є m-насиченими функціями.

Список літератури:

1. Городилова А.А. От криптоанализа шифра к криптографическому свойству булевой функции // Прикладная дискретная математика. 2016. № 3(33). С.16–44.

2. Панкратова И.А. Булевы функции в криптографии : учеб. пособие. Томск : Изд. Дом Томск. Гос. ун-та. 2014. 88 с.

3. Мухачев В.А., Хорошко В.А. Методы практической криптографии. К. : ООО «Полиграф-Консалтинг». 2005. 215 с.

4. Токарева Н.Н. Обобщения бент-функций. Обзор работ // Дискретный анализ и исследование операций. 2010. Т. 17, №1. С.33-62.

5. Токарева Н.Н. Нелинейные булевы функции: бент-функции и их обобщения. Изд-во LAP LAMBERT Academic Publishing (Saarbrucken, Germany). 2011. 180 с.

6. Агафонова И.В. Криптографические свойства нелинейных булевых функций // Семинар по дискрет. гармон. анализу и геометр. моделированию. СПб. : DHA & CAGD, 2007. С. 1–24.

7. Шевелев Ю.П. Дискретная математика. Ч. 1: Теория множеств. Булева алгебра (Автоматизированная технология обучения «Символ») : учеб. пособие. Томск. гос. ун-т систем управления и радиоэлектроники, 2003. 118 с.

8. Молдовян А.А. Криптография. Скоростные шифры. БХВ-Петербург, 2002. 496 с.

9. Логачев О.А., Сальников А.А., Смышляев С.В., Ященко В.В. Булевы функции в теории кодирования и криптологии. 2-е изд. Москва : МЦНМО, 2012. 584с.

10. Смышляев С.В. О криптографических слабостях некоторых классов преобразований двоичных последовательностей // Прикладная дискретная математика. 2010. № 1. С. 5–15.

11. Таранников Ю.В. О корреляционно-иммунных и устойчивых булевых функциях // Математические вопросы кибернетики. Физматлит, 2002. Вып. 11. С. 91–148.

Харківський національний
університет імені В.Н. Каразіна

Надійшла до редколегії 11.02.2018

УДК 681.3.06

Г.З. ХАЛІМОВ, д-р техн. наук, Є.В. КОТУХ, канд. техн.наук,
Ю.О. СЕРГІЙЧУК, О.С. МАРУХНЕНКО

АНАЛІЗ СКЛАДНОСТІ РЕАЛІЗАЦІЙ КРИПТОСИСТЕМИ НА ГРУПІ СУДЗУКІ

Криптографія з відкритим ключем будується на труднощах розв'язання математичних проблем, які дуже часто, але не виключно, виникають з теорії чисел. На початку 80-х років було запропоновано застосування групових теоретичних проблем для криптографії (Wagner і Magyarik [1], Wagner [2], Magliveras [3]). Зокрема в роботах Magliveras та ін. були зроблені пропозиції для криптографічних схем на основі спеціальних розкладених кінцевих груп (так звані логарифмічні сигнатури) [3]. Крім того, відомі інші криптографічні дослідження Gonzarlez Vasco, Steinwandt, Birget, Bohliet і ін. Ці розкладання як математичні об'єкти цікаві самі по собі. Наприклад, работа Hajors про гіпотезу Міньковського показує, що для абелевих груп цей вид розкладання виникає при вивченні багатовимірних покриттів (див. [4]).

Прикладами криптосистем з відкритим ключем є MST1, MST2, MST3. Актуальним завданням їх реалізації є побудова коротких логарифмічних сигнатур. Логарифмічні сигнатури, як особливий тип групових розкладів представляються в якості основних компонентів деяких криптографічних ключів. Науковий інтерес пов'язан з пошуком логарифмічних сигнатур в кінцевих групах (такі розкладання існують для вирішуваних, симетричних і знакозмінних груп), оцінкою їх практичної можливості і секретності.

В роботі розглянуто основні реалізації криптосистем на групах і аналіз оцінки складності обчислень.

Визначення та властивості логарифмічних сигнатур

Основні позначення, визначення та основні факти про логарифмічні сігнатури, накриття для кінцевих груп і їх породжені відображення представимо на основі опису в [4].

Нехай ς – кінцева абстрактна група. Визначимо ширину ς як позитивне ціле число $w = \lceil \log|\varsigma| \rceil$. Позначимо через $\varsigma^{[\mathbf{Z}]}$ сукупність усіх кінцевих послідовностей елементів ς і відобразимо елементи $\varsigma^{[\mathbf{Z}]}$ як однорядкові матриці із записами в ς. Нехай $X = [x_1, x_2 \ldots, x_r]$ і $Y = [y_1, y_2 \ldots, y_s]$ будуть двома елементами в. $\varsigma^{[\mathbf{Z}]}$

Визначимо $X \cdot Y = [x_1y_1, x_1y_2, \ldots, x_1y_s, x_2y_1, x_2y_2, ..., x_2y_s, ..., x_ry_1, x_ry_2, ..., x_ry_s]$.

Замість запису $X \cdot Y$ можемо записати $X \otimes Y$ як звичайний тензорний добуток матриць або для короткого написання залишимо XY. Якщо $X = [x_1, \ldots, x_r] \in \varsigma^{|\mathbf{Z}|}$, позначимо через $\bar{X}$ елемент $\sum_{i=1}^{r} x_i$ у груповому кільці $\mathbf{Z}\varsigma$.

Нехай $\alpha = [A_1, A_2, ...A_s]$ – послідовність $A_i \in \varsigma^{|\mathbf{Z}|}$, така, що $\sum_{i=1}^{s} |A_i|$ обмежена поліномом в $\log|\varsigma|$. Нехай

$$\bar{A}_1 \cdot \bar{A}_2 \cdots \bar{A}_s = \sum_{g \in \varsigma} a_g g \,, a_g \in \mathbf{Z} \,.$$

Нехай S – підмасив ς. Тоді, можемо сказати, що α є:

(i) накриттям для ς (або S), якщо $a_g > 0$ для всіх $g \in \varsigma$ ($g \in S$);

(ii) логарифмічною сігнатурою для $\varsigma(S)$, якщо $a_g = 1$ для кожного $g \in \varsigma$ ($g \in S$).

Нехай α буде накриттям.

Визначимо $\lambda_{\min} := \min\{a_g : g \in \varsigma\}$, $\lambda_{\max} := \max\{a_g : g \in \varsigma\}$ та $\lambda := \lambda_{\max} / \lambda_{\min}$. Співвідношення λ визначає степінь однорідності α. Говоримо, що α – однорідне накриття, якщо $\lambda = 1$. Зокрема, логарифмічна сігнатура є однорідним накриттям.

Зверніть увагу, що якщо $\alpha = [A_1, ... A_s]$ є логарифмічною сігнатурою для ς, тоді кожний елемент $y \in \varsigma$ може бути однозначно виражений як добуток вигляду

$$y = q_1 \cdot q_2 \cdots q_{s-1} \cdot q_s, \; q_i \in A_i. \tag{1}$$

Зазвичай, для загальних накриттів, що факторизуються (1) не унікальне й проблема пошуку розкладення для даного $y \in \varsigma$, у загальному випадку, є обчислювально неможливою.

Нехай $\alpha = [A_1, ... A_s]$ – накриття для ς з $r_i = |A_i|$. Тоді A_i називаються блоками від α і вектором $(r_1, ..., r_s)$ блока довжин r_i і класу α. Визначимо довжину α як ціле число $\ell = \sum_{i=1}^{s} r_i$. Однорідне накриття $\alpha = [A_1, ... A_s]$ класу $(r, r, ..., r)$ називається $[s, r]$-мережею. Говоримо, що α є нетривіальним, якщо $s \geq 2$ й $r_i \geq 2$ для $1 \leq i \leq s$, у протилежному випадку α є тривіальним. Позначимо через $C(\varsigma)$ і $\Lambda(\varsigma)$ відповідні подання накриттів і логарифмічних сігнатур групи ς.

Нехай $\Gamma = \{(\varsigma_\ell, \alpha_\ell)\}_{\ell \in T}$ – сімейство пар, індексоване за допомогою параметра безпеки ℓ, де ς_ℓ – групи загальної презентації, α_ℓ – особливе накриття для ς_ℓ довжини полінома, який дорівнює ℓ. Вважаємо, що Γ – просте, якщо імовірнісний алгоритм поліноміального часу виконання A, за якого для кожного $g \in \varsigma_\ell$, A приймає (α_ℓ, g) вхідні й вихідні дані факторизації $\varphi(g)$ від g відносно α_ℓ з переважною імовірністю успіху. Або вважаємо, що Γ – випадкове, якщо для будь-якого імовірнісного алгоритму поліноміального часу виконання A, імовірність того, що A успішне у факторизації випадкового елемента g від ς, є незначною.

Для кінцевих груп є елементи $\{(\varsigma_\ell, \alpha_\ell)\}_\ell$, для яких факторизація вважається складною. Для прикладу, нехай q – проста степінь числа, для якого проблема дискретного логарифмування в мультиплікативній групі кінцевого поля F_q вважається складною. Нехай $2^{\ell-1} \leq q - 1 < 2^\ell$ і нехай ς_ℓ буде раніше згаданою мультиплікативною групою F_q^*. Нехай f – генератор ς_ℓ. Якщо $\alpha_\ell = [A_1, A_2, ... A_\ell]$, де $A_i = [1, f^{2^{i-1}}]$, тоді α_ℓ – накриття ς_ℓ й факторизація стосовно α_ℓ зводиться до розв'язання проблеми дискретного логарифмування для ς_ℓ.

Нехай $\alpha = [A_1, A_2, ... A_s]$ – накриття групи ς. Нехай $g_0, g_1, ..., g_s \in \varsigma$ і розглянемо $\beta = [B_1, B_2, ... B_s]$ з $B_i = g_{i-1}^{-1} A_i g_i$ для спеціального випадку, де $g_0 = 1$ й $g_s = 1$, тоді β називається багатошаровим накриттям від α. Зверніть увагу, що β також є накриттям для ς.

Нехай $\alpha = [A_1, A_2, ... A_s]$ – накриття класу $(r_1, r_2, ..., r_s)$ для ς з $A_i = [a_{i,1}, a_{i,2}, ..., a_{i,r_i}]$ і нехай $m = \prod_{i-1}^{s} r_i$. Нехай $m_1 = 1$ і $m_i = \prod_{j=1}^{i-1} r_j$ для $i = 2, ..., s$. Позначимо τ як канонічну бієкцію від $\mathbf{Z}_{r_1} \oplus \mathbf{Z}_{r_2} \oplus \cdots \oplus \mathbf{Z}_{r_s}$ на $\mathbf{Z}_m$, тобто

$$\mathbf{Z}_{r_1} \oplus \mathbf{Z}_{r_2} \oplus \cdots \oplus \mathbf{Z}_{r_s} \to \mathbf{Z}_m$$

$$\tau(j_1, j_2, ..., j_s) := \sum_{i=1}^{s} j_i m_i$$

Використовуючи τ, можемо визначити сюр'єктивне відображення $\breve{\alpha}$, породжене α:

$$\breve{\alpha}: \mathbf{Z}_m \to \varsigma$$

$$\breve{\alpha}(x) := a_{1,j_1} \cdot a_{2,j_2} \cdots a_{s,j_s}$$

де $(j_1, j_2, ..., j_s) = \tau^{-1}(x)$. Оскільки τ й τ^{-1} ефективно обчислювані, то відображення $\breve{\alpha}(x)$ також ефективно обчислюване.

З іншого боку, з даним накриттям α і елементом $y \in \varsigma$, щоб визначити будь-який елемент $x \in \breve{\alpha}^{-1}(y)$, необхідно отримати кожне з можливих розкладень класу для y і визначити показники $j_1, j_2, ..., j_s$ такі, що $y = a_{1,j_1} \cdot a_{2,j_2} \cdots a_{s,j_s}$. Це можливо тільки якщо α – просте. Оскільки вектор $(j_1, j_2, ..., j_s)$ визначено, то $\breve{\alpha}^{-1}(y) = \tau(j_1, j_2, ..., j_s)$ може бути ефективно обчислено.

Два накриття α й β вважатимуться еквівалентними, якщо $\breve{\alpha} = \breve{\beta}$.

Приклад

Наведемо приклад за участю α й β для змінної групи A_5. Класи α й β дорівнюють (5,2,6) і (3,4,5) і $|A_5| = 5 \cdot 2 \cdot 6 = 3 \cdot 4 \cdot 5 = 60$ відповідно. У табл. 1 блоки α й β подані вертикально. Щоб ефективно обчислити τ^{-1} й τ, прикладемо канонічні логарифмічні сігнатури τ_α й τ_β адитивної групи $\mathbf{Z}_{60}$ вліво від α і вправо від β. Відповідні класи τ_α й τ_β дорівнюють (5,2,6) і (3,4,5) лише для α й β.

Таблиця 1

Дві логарифмічні сігнатури від A_5

	τ_α	α
	$\mathbf{Z}_{60}$	A_5
$x_1 \to$	0	(1)(2)(3)(4)(5)
	1	(1 2 5 3 4)
	2	(1 5 4 2 3)
	3	(1 3 2 4 5)
	4	(1 4 3 5 2)
$x_2 \to$	0	(1 2 5 3 4)
	5	(2 4) (3 5)
$x_3 \to$	0	(1 3 5 4 2)
	10	(13) (24) (5)
	20	(1)(2)(3)(4)(5)
	30	(15)(23)(4)
	40	(132)(4)(5)
	50	(123)(4)(5)

A_5

β	τ_β	
A_5	$\mathbf{Z}_{60}$	
(1)(2)(345)	0	$\leftarrow y_1$
(1)(2)(354)	1	
(1)(2)(3)(4)(5)	2	
(1)(23)(45)	0	$\leftarrow y_2$
(1)(253)(4)	3	
(1)(243)(5)	6	
(1)(2)(3)(4)(5)	9	
(124)(3)(5)	0	$\leftarrow y_3$
(1)(235)(4)	12	
(13)(2)(45)	24	
(1 5 3 4 2)	36	
(1 4 3 2 5)	48	

Тепер можемо продемонструвати, як на практиці обчислити $\breve{\alpha}: \mathbf{Z}_{60} \to A_5$. Будь-який елемент $x \in \mathbf{Z}_{60}$ можна однозначно записати як суму елементів τ_α, використовуючи тільки один елемент з кожного блоку. Визначення цієї декомпозиції x містить у собі «жадібний» вибір компонентів, одного з кожного блоку, послідовно з нижнього блоку до верхнього й, по суті, визначає $\tau^{-1}(x) = (j_1, j_2, j_3)$. Якщо x_i є елементами A_5 відповідними j_i, то обчислюємо $\breve{\alpha}(x) = x_1 x_2 x_3$. Зокрема, якщо $x = 47$, маємо: 47=40+5+2 і компоненти $j_1 = 2$, $j_2 = 5$ і

$j_3 = 40$, елементи, що вказують $x_1 = (15423)$ $x_2 = (24)(35)$ і $x_3 = (132)$ від A_5. Тоді можемо обчислити: $\breve{\alpha}(47) = x_1 x_2 x_3 = (15423) \cdot (24)(35) \cdot (132) = (125)$.

Якщо розкладемо $y = \breve{\alpha}(x)$ відносно другої логарифмічної сігнатури β, то отримаємо $y = y_1 y_2 y_3$. З елементів y_i отримуємо відповідні елементи адитивної τ_β і формуємо суму. Для окремого випадку, $y = (125) = y_1 y_2 y_3 = (354) \cdot (253) \cdot (124)$ відповідними компонентами τ_β є 1,3,0.

Таким чином, $\breve{\beta}^{-1}((125)) = 1 + 3 + 0 = 4$. Необхідно зазначити у даному прикладі, що α й β належать до класу простих логарифмічних сігнатур, але β, насправді, суперпроста. Ми не пояснюватимемо, як ефективно отримати розкладення $y = y_1 y_2 y_3$, для цього дивіться [6].

Коли група, яка лежить в основі, обрана правильно, бієкція $\breve{\alpha}\breve{\beta}^{-1}$ може використовуватися як криптографічне перетворення з ключем (α, β) у симетричній криптосистемі PGM [8, 9] або як криптографічні примітиви в інших системах.

Опис криптосистеми *MST₃*

Розглянемо структуру криптосистеми MST_3 [4]. Нехай ς – кінцева неабелева група з нетривіальним центром $\mathbf{Z}$, таким, що ς не розкладаються над $\mathbf{Z}$. Також припустимо, що $\mathbf{Z}$ є досить великим, таким, що пошук перебором у $\mathbf{Z}$ є обчислювально нездійсненним.

Криптографічна гіпотеза, що є основою для криптосистеми, полягає в тому, що якщо $\alpha = [\mathrm{A}_1, \mathrm{A}_2, ..., \mathrm{A}_s] := (a_{i,j})$ – випадкове накриття для «великого» підмасива S в ς, то пошук розкладення $g = a_{1j_1} a_{2j_2} \cdots a_{sj_s}$ для будь-якого елемента $g \in S$ відносно α є невирішуваною проблемою.

Генерація ключових даних

Аліса обирає велику групу ς, описану раніше й генерує:

(1) проста логарифмічна сігнатура $\beta = [B_1, B_2, ..., B_s] := (\mathrm{b}_{ij})$ класу $(\mathrm{r}_1, \mathrm{r}_2, ..., \mathrm{r}_s)$ для $\mathbf{Z}$;

(2) випадкове накриття $\alpha = [\mathrm{A}_1, \mathrm{A}_2, ..., \mathrm{A}_s] := (a_{i,j})$ такого самого класу, як і β для деякої підмножини J від ς, такого, що $\mathrm{A}_1, ..., \mathrm{A}_s \subseteq \varsigma \setminus \mathbf{Z}$.

Потім вона обирає $t_0, t_1 ..., t_s \in \varsigma \setminus \mathbf{Z}$ й обчислює:

(3) $\breve{\alpha} = [\tilde{\mathrm{A}}_1, \tilde{\mathrm{A}}_2, ..., \tilde{\mathrm{A}}_s]$, де $\tilde{A}_i = t_{i-1}^{-1} A_i t_i$ для $i = 1, ..., s$;

(4) $\gamma := (h_{ij}) = (b_{ij} \tilde{a}_{ij})$.

Аліса публікує свій відкритий ключ $(\alpha = (a_{ij}), \gamma = (h_{ij}))$, а $(\beta = (\mathrm{b}_{ij}), (\mathrm{t}_0, ..., \mathrm{t}_s))$ – зберігає як свій закритий ключ.

Шифрування

Якщо Боб хоче відіслати повідомлення $x \in \mathbf{Z}_{|Z|}$ для Аліси, то він:

(1) обчислює $y_1 = \breve{\alpha}(x)$ і $y_2 = \breve{\gamma}(x)$;

(2) посилає Алісі $y = (y_1 y_2)$.

Дешифрування

Тепер, коли Аліса знає y_2:

$$y_2 = \breve{\gamma}(x) = b_{1j_1}\tilde{a}_{1j_1} \cdot b_{2j_2}\tilde{a}_{2j_2} \cdots b_{sj_s}\tilde{a}_{sj_s} = b_{1j_1}t_0^{-1}a_{1j_1}t_1 \cdots b_{sj_s}t_{s-1}^{-1}a_{sj_s}t_s =$$
$$= b_{1j_1}b_{2j_2} \cdots b_{sj_s}t_0^{-1}a_{1j_1}a_{2j_2} \cdots a_{sj_s}t_s = \breve{\beta}(x).t_0^{-1}\breve{\alpha}(x)t_s = \breve{\beta}(x).t_0^{-1}y_1t_s,$$

вона може обчислити $\breve{\beta}(x) = \mathrm{y}_2 t_s^{-1}y_1^{-1}t_0$.

Аліса відновлює x з $\breve{\beta}(x)$, використовуючи $\breve{\beta}^{-1}$, який ефективно обчислюємо, оскільки β – проста.

Особливості реалізації криптосистеми MST_3 на Судзукі 2-групах

В реалізації MST_3 запропонована Судзукі 2-група з порядком q^2. Використовуючи позначення Хігмана [5], Судзукі 2-група з порядком q^2 буде позначена як $A(\mathrm{m},\theta)$. Нехай $q = 2^m$ з $3 \le m \in N$ є таким, що поле F_q має нетривіальний автоморфізм θ непарного порядку. Тут мається на увазі, що m не є степенем 2. Тоді групи $A(\mathrm{m},\theta)$ існують.

Насправді, якщо ми визначаємо $\varsigma := \{S(a,b) \mid a,b \in \mathrm{F}_q\}$, де $S(a,b) = \begin{pmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ b & a^\theta & 1 \end{pmatrix}$ є матрицею 3×3 над полем F_q, це показує, що група ς ізоморфна $A(\mathrm{m},\theta)$. Отже, ς має порядок q^2 і маємо

$$Z := \mathbf{Z}(\varsigma) = \Phi(\varsigma) = \varsigma' = \Omega_1(\varsigma) = \{S(0,b) \mid b \in \mathrm{F}_q\}$$

Оскільки центр $\mathbf{Z}(\varsigma)$ є елементарною абелевою групою порядку q, він може бути ідентифікований з адитивною групою поля F_q. Крім того, фактор-група $\varsigma / \Phi(\varsigma)$ є елементарна абелева група порядку q. Тоді легко перевірити, що множення двох елементів у ς здійснюється відповідно до правила

$$S(a_1,b_1)S(a_2,b_2) = S(a_1+a_2, b_1+b_2+a_1^\theta a_2).$$

Обернений елемент знаходиться за формулою

$$S(a,b)^{-1} = S(a, b+a^{\theta+1}).$$

Алгоритм роботи системи для шифрування має такі етапи [6].

Генерація ключових даних:

1. Обрати велику групу $G = A(m,\theta),\ q = 2^m$.

2. Згенерувати логарифмічну сігнатуру, що факторизується:
$\beta = [B_1,...,B_s] = (b_{i,j}) = (S(0, b_{i,j}.b))$ типа $(r_1,...,r_s)$, где $b_{i,j}.b \in F_q$.

3. Згенерувати випадкове накриття $\alpha = [A_1,...,A_s] = (a_{i,j}) = (S(a_{i,j}.a, a_{i,j}.b))$ того ж типу, що й β, де $a_{i,j}.a \in F_q / \{0\}, a_{i,j}.b \in F_q$.

4. Згенерувати випадкові значення $t_0, t_1, ..., t_s \in G$, матрицю випадкових бітів $\sigma = [q \times q]$.

5. Побудувати гомоморфізм f:$G \to Z$, визначений як $f(S(a,b))=S(0,g(a))$ (в даній реалізації було використано множення на випадкову бітову матрицю $f(a) = a^\sigma$).

6. Обчислити $\gamma = [H_1,...,H_s] = (h_{i,j}) = (S(h_{i,j}.a, h_{i,j}.b))$, де $h_{i,j} = t_{i-1}^{-1} * a_{i,j} * t_i * b_{i,j} * f(a_{i,j})$.

7. Відкритий ключ – $[\alpha, \gamma]$, приватний ключ – $[\beta, (t_0, t_1, ..., t_s), f]$ та додаткові дані, необхідні для факторизації β.

Шифрування повідомлення m:

1. Створити елемент $\chi = S(0, m) \in G$
2. Згенерувати випадкове число $R \in Z$
3. Обчислити криптограму $y_1 = \alpha'(R) * \chi$, $y_2 = \gamma'(R) * \chi$.

Зауваження: для зменшення розмірів шифротексту достатньо зберігати $(y_{1.a}, y_{1.b}, y_{2.b})$, при розшифруванні складова $y_{2.a}$ може бути відновлена за формулою $y_{2.a} = y_{1.a} \oplus t_{0.a} \oplus t_{S.a}$

Розшифрування:

1. Обчислити $\beta'(R) = f(y_1)^{-1} * y_1^{-1} * t_0 * y_2 * t_S^{-1}$.
2. Виконати факторізацію $R = \beta'^{-1}(R)$.
3. Обчислити $\alpha'(R)$.
4. Відновити $m = y_{1.b} \oplus \alpha'(R)_{.b}$.

Тестування шифрування виконано на комп'ютері з ОС Ubuntu 16.04, процесором Intel® Core™ i7-4702MQ CPU @2,20 GHz, 12 ГБ ОЗП, результати представлено в табл. 2, 3.

Таблиця 2

Витрати на шифрування/розшифрування в кінцевому полі 128 біт

Класи розбиття	Час генерації ключових даних, мс	Розмір приватного ключа, байт	Розмір публічного ключа, байт	Час зашифрування 100 КБайт, мс	Час розшифрування 100 КБайт, мс
128[2] →64[4]	56	78830	39761	4749	2711
64[4] →32[16]	59	111726	75217	2388	1487
32[16] →16[256]	169	671918	590609	1205	888

Таблиця 3

Витрати на шифрування/розшифрування в кінцевому полі 256 біт

Класи розбиття	Час генерації ключових даних, мс	Розмір приватного ключа, байт	Розмір публічного ключа, байт	Час зашифрування 100 КБайт, мс	Час розшифрування 100 КБайт, мс
256[2] →128[4]	57	249630	128593	14811	7911
128[4]→ 64[16]	106	361502	248657	7540	4196
64[16] →32[256]	798	2193054	1967569	3782	2318

Порівняння з направленим шифруванням з RSA алгоритмом представлено в табл. 4.

Таблиця 4

Витрати на шифрування/розшифрування за RSA алгоритмом

Розрядність ключових параметрів, біт	Час генерації ключових даних, мс	Розмір приватного ключа, байт	Розмір публічного ключа, байт	Час зашифрування 100 КБайт, мс	Час розшифрування 100 КБайт, мс
512	3,368	342	92	66,987	641,277
1024	8,685	632	160	117,947	2116,400
2048	63,658	1214	292	243,887	9853,580
4096	707,645	2373	548	591,868	64250,400

Висновки

1. Для оптимізації витрат щодо розміру приватного та публічного ключів, часу зашифрування та розшифрування необхідно здійснити підбір класу розбиття логарифмічної сігнатури на блоки. Часові витрати можна зменшити в декілька разів. Використання кінцевого поля 128, 256 бітів достатньо для забезпечення найвищого класу захисту по класифікації криптосистем.

2. При обчисленні в кінцевому полі 2048 та 4096 бітів час зашифрування та розшифрування RSA алгоритмом в десятки разів більше в порівнянні з криптосистемою MST_3, але забезпечує суттєву економію витрат щодо розміру приватного та публічного ключів.

Список літератури:

1. N.R. Wagner and M. R. Magyarik. A Public Key Cryptosystem Based on the Word Problem // Advances in Cryptology. Proceedings of CRYPTO 1984, pp. 19-36, edited by G. R. Blakley and D. Chaum, Lecture Notes in Computer Science 196. Berlin: Springer, 1985.

2. N.R. Wagner. Searching for Public-Key Cryptosystems // In Proceedings of the 1984 Symposium on Security and Privacy (SSP ’84), pp. 91-98. Los Alamitos, CA: IEEE Computer Society Press, 1990.

3. S.S. Magliveras. A Cryptosystem from Logarithmic Signatures of Finite Groups // Proceedings of the 29th Midwest Symposium on Circuits and Systems, pp. 972-975. Amsterdam: Elsevier Publishing Company, 1986.

4. W. Lempken, S.S. Magliveras, Tran van Trung and W. Wei. A public key cryptosystem based on non-abelian finite groups // J. of Cryptology, 22(2009), 62-74.

5. G. Higman, Suzuki 2-groups.Ill. J. Mathematic. 1963. V.7. P.79-96.

6. Pavol Svaba. Covers and logarithmic signatures of finite groups in cryptography : Dissertation. Bratislava : Slowakische Republik, 2011.

Харківський національний
університет радіоелектроніки

Надійшла до редколегії 15.03.2018

УДК 004.056

Е.В. ИСИРОВА, А.В. ПОТИЙ, д-р техн. наук, В.В. СЕМЕНЕЦ, д-р техн. наук

ПРИНЦИПЫ ПОСТРОЕНИЯ ДЕЦЕНТРАЛИЗОВАННОЙ ИНФРАСТРУКТУРЫ ОТКРЫТЫХ КЛЮЧЕЙ

Введение

Успешное внедрение современных технологий электронного управления, электронных доверительных услуг невозможно без создания соответствующей инфраструктуры. Технологичной инфраструктурой реализации упомянутых технологий выступает инфраструктура открытых ключей (ИОК). Использование электронных доверительных услуг с применением цифровой подписи опирается на доверие между субъектами взаимодействия, инфраструктуру открытых ключей и направлено на реализацию модели доверия.

В 2017 году в Украине принят Закон Украины «Об электронных доверительных услугах», который определяет правовые и организационные основы предоставления электронных доверительных услуг, в том числе трансграничных, права и обязанности субъектов правовых отношений в сфере электронных доверительных услуг, порядок осуществления государственного надзора (контроля) за соблюдением требований законодательства в сфере электронных доверительных услуг, а также правовые и организационные основы осуществления электронной идентификации. Для эффективного использования и качественного предоставления таких услуг необходимо решать много технологически сложных задач и технических проблем. В 2004 году в Украине была реализована архитектура ИОК, которая стала базой для использования технологии с открытыми ключами и предоставления услуг по управлению криптографическими ключами. Данная архитектура представляет собой иерархическую систему. Кроме иерархической архитектуры существует еще ряд возможных для использования, которые не были использованы из-за невозможности надежной реализации модели доверия. Цель статьи – предложение новой концепции построения инфраструктуры открытых ключей с использованием технологии blockchain.

Описание существующей инфраструктуры открытых ключей, проблемные вопросы при построении

Инфраструктура открытых ключей (ИОК, англ. PKI – Public Key Infrastructure) – набор средств (технических, материальных, людских и т. д.), распределенных служб и компонентов, в совокупности используемых для поддержки криптозадач на основе закрытого и открытого ключей [1].

В основе PKI лежат использование криптографической системы с открытым ключом и несколько основных принципов [2]:

- закрытый ключ (private key) известен только его владельцу;
- удостоверяющий центр создает электронный документ – сертификат открытого ключа, тем самым удостоверяя факт того, что закрытый (секретный) ключ известен эксклюзивно владельцу этого сертификата, открытый ключ (public key) свободно передается в сертификате;
- никто не доверяет друг другу, но все доверяют удостоверяющему центру;
- удостоверяющий центр подтверждает или опровергает принадлежность открытого ключа заданному лицу, которое владеет соответствующим закрытым ключом.

Основным нормативным документов является стандарт ITU-T X.509 для инфраструктуры открытого ключа и инфраструктуры управления привилегиями (англ. Privilege Management Infrastructure). Он определяет стандартные форматы данных и процедуры распределения открытых ключей с помощью соответствующих сертификатов с цифровыми подписями. Эти сертификаты предоставляются центрами сертификации (англ. Certificate Authority). Кроме того, X.509 определяет формат списка отозванных сертификатов (англ. Certificate revocation lists, CRL), формат сертификатов атрибутов (англ. Attribute certificates) и алгоритм проверки подписи построением пути сертификации (англ. Certification path validation algorithm).

ИОК состоит из ряда подсистем:

- организационно-техническая (включает в себя политику сертификации, регламент);
- подсистема управления списками отозванных сертификатов (уполномоченный на сертификацию, центр регистрации, репозиторий, конечные пользователи);
- подсистема применений ИОК (web-защита, защищенный email, защищенный документооборот, VPN)

Для технологии открытых ключей необходимо, чтобы пользователь открытого ключа был уверен, что этот ключ принадлежит именно тому удаленному субъекту (пользователю или системе), который будет использовать средства шифрования или цифровой подписи. Такую уверенность дают сертификаты открытых ключей. Сертификат имеет ограниченный срок действия. Поскольку пользователь сертификата может самостоятельно проверить его подпись и срок действия, сертификаты могут распространяться через незащищенные каналы связи и серверные системы, а также храниться в кеш-памяти незащищенных пользовательских систем.

При построении ИОК необходимо решать проблемные вопросы на нескольких уровнях:

– правовом (регулирование взаимоотношений между участниками процессов сертификации);
– системном (обоснование выбора архитектуры с учетом решаемых задач);
– процедурно-функциональном (определение основных функциональных требований к системе сертификации, установление перечня услуг центров сертификации);
– функционально-техническом (определение функциональной структуры, физической топологии, обоснование требований безопасности);
– техническом (обоснование выбора аппаратных средств для центров сертификации, в том числе средств криптографической защиты).

Основные угрозы для систем такого типа:

– отказ от выполнения действий;
– подделка сертификата.

Для обеспечения доверия необходимо обеспечить функционирование системы в рамках актуальной модели доверия. Стандарт X.509 [1] предполагает возможное использование следующих моделей доверия:

– строгая иерархия уполномоченных на сертификацию;
– нестрогая иерархия уполномоченных на сертификацию;
– иерархия на базе политик;
– модель распределенного доверия;
– четырехсторонняя модель доверия;
– модель доверия вокруг пользователя;
– web-модель доверия.

На сегодняшний день подавляющее ИОК построены на основе строгой иерархии уполномоченных на сертификацию (рис. 1)

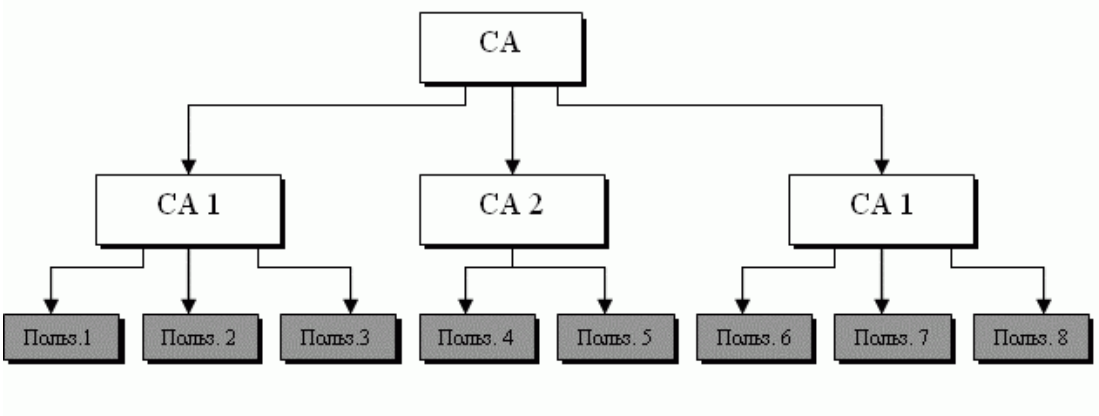


Рис. 1. Иерархическая структура ИОК [2]

Однако такая структура имеет недостатки:

– вся безопасность системы зависит от корневого сертификата центрального уполномоченного на сертификацию. В случае его компрометации все сертификаты в системе являются скомпрометированными;

– пользователи фактически не распоряжаются своими идентификационными данными, при необходимости внести какие-либо корректировки необходимо обращаться к уполномоченному на сертификацию;

– отсутствие интероперабельности системы. Сертификаты, выпущенные разными уполномоченными, не могут быть использованы параллельно;

– отсутствие однозначного соответствия между пользователем и сертификатом, поскольку для одного пользователя может быть выпущено несколько сертификатов.

Другие модели доверия либо слабо распространены, либо не используются вовсе. Однако анализ показал, что с помощью новой технологии blockchain могут быть надежно реализованы и другие модели доверия, в частности модель доверия вокруг пользователя. Рассмотрим ее подробнее.

Модель доверия вокруг пользователя [1, 2]. В такой модели доверия пользователь самостоятельно отвечает за решения каким сертификатам доверять, а какие считать ненадежными. Эти решения зависят от ряда факторов. Первичным источником доверия являются сертификаты родственников, друзей, знакомых, т.е. тех, кого пользователь знает лично (т.е. первичная идентификация проводится самостоятельно пользователем).

Доверие, сконцентрированное вокруг пользователя, иллюстрирует известная система Pretty Good Privacy (PGP) (рис. 2). Пользователь А может решить: доверять сертификату В (на основе доверия к цепочке сертификатов от пользователя D к пользователю С и пользователю В) или отвергнуть сертификат В, аргументируя это тем, что к "неизвестному" пользователю В ведет слишком много связей от "знакомого" пользователя D.

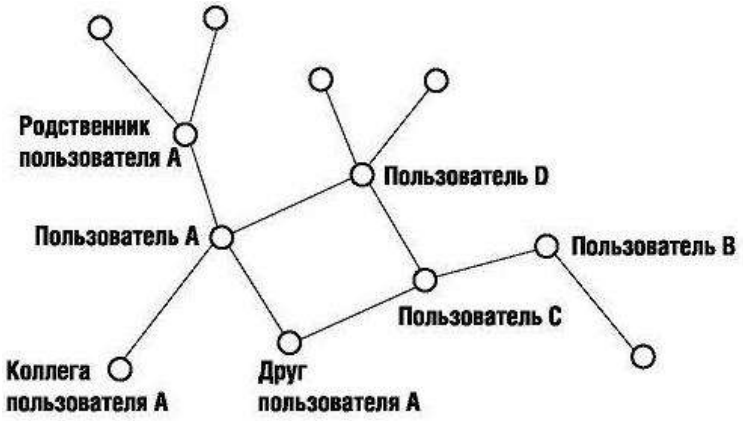


Рис. 2. Модель доверия вокруг пользователя [2]

В силу своей зависимости от действий и решений пользователей модель доверия, сконцентрированного вокруг пользователя, может использоваться только в узком и высокотехнологичном сообществе, но она нежизнеспособна в обычном сообществе, в котором многие пользователи не имеют достаточных знаний о безопасности и технологии PKI. Более того, эта модель не подходит для тех сфер (корпоративной, финансовой, правительственной), где необходим контроль за тем, с кем взаимодействуют и кому доверяют пользователи.

Далее предложим вариант, как можно избежать указанных недостатков с применением технологии blockchain. Кратко опишем саму технологию blockchain.

Технология blockchain

Блокче́йн (англ. blockchain или block chain) – выстроенная по определенным правилам непрерывная последовательная цепочка блоков, содержащих информацию [4].

Блок транзакций – специальная структура для записи группы транзакций в системе Биткойн и аналогичных ей. Транзакция считается завершенной и достоверной («подтвержденной»), когда проверены ее формат и подписи, и когда сама транзакция объединена в группу с несколькими другими и записана в специальную структуру – блок.

Содержимое блоков может быть проверено, так как каждый блок содержит информацию о предыдущем блоке. Все блоки выстроены в одну цепочку, которая содержит информацию обо всех совершенных когда-либо операциях в базе. Самый первый блок в цепочке – первичный блок (англ. genesis block) – рассматривается как отдельный случай, так как у него отсутствует родительский блок.

Блок состоит из заголовка и списка транзакций (рис. 3). Заголовок блока включает в себя свой хеш, хеш предыдущего блока, хеши транзакций и дополнительную служебную информацию. Для транзакций в блоке используется древовидное хеширование, аналогичное формированию хеш-суммы для файла в протоколе BitTorrent. Транзакции, кроме начисления комиссии за создание блока, содержат внутри параметра input ссылку на транзакцию с предыдущим состоянием данных [4].

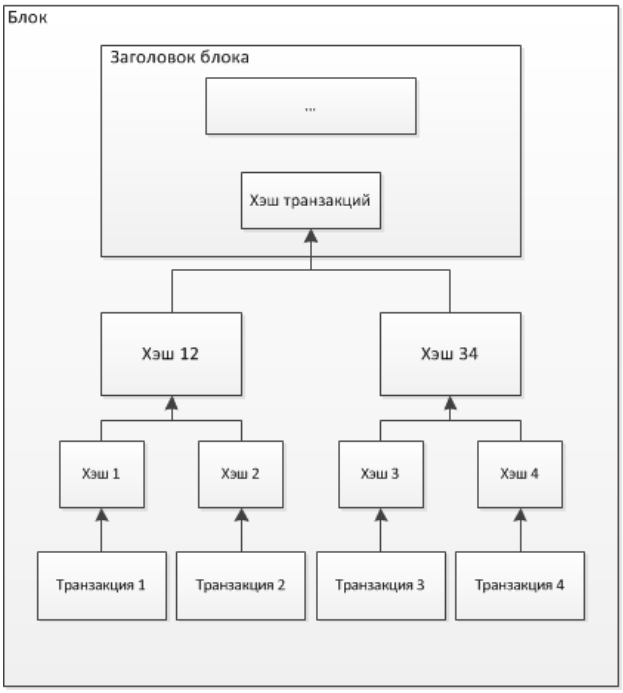


Рис. 3. Структура блока

Созданный блок будет принят остальными пользователями, если числовое значение хеша заголовка равно или ниже определенного числа, величина которого периодически корректируется. Так как результат хеширования функции SHA-256 считается необратимым, на данный момент нет алгоритма получения желаемого результата, кроме случайного перебора. Если хеш не удовлетворяет условию, то в заголовке изменяется параметр *nonce* и хеш пересчитывается. Обычно требуется большое количество пересчетов. Когда вариант найден, узел рассылает полученный блок другим подключенным узлам, которые проверяют блок. Если ошибок нет, то блок считается добавленным в цепочку, и следующий блок должен включить в себя его хеш.

Блоки одновременно формируются множеством «участников». Удовлетворяющие критериям блоки отправляются в сеть, включаясь в распределенную базу блоков. Регулярно возникают ситуации, когда несколько новых блоков в разных частях распределенной сети называют предыдущим один и тот же блок, то есть цепочка блоков может ветвиться. Специально или случайно можно ограничить ретрансляцию информации о новых блоках (например, одна

из цепочек может развиваться в рамках локальной сети). В этом случае возможно параллельное наращивание различных ветвей. В каждом из новых блоков могут встречаться как одинаковые транзакции, так и разные, вошедшие только в один из них. Когда ретрансляция блоков возобновляется, участники начинают считать главной цепочку с учетом уровня сложности хеша и длины цепочки. При равенстве сложности и длины предпочтение отдается той цепочке, конечный блок которой появился раньше (рис. 4). Транзакции, вошедшие только в отвергнутую ветку, теряют статус подтвержденных.

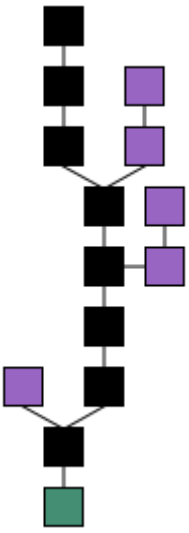


Рис. 4. Цепочка блоков

Таким образом, цепочка блоков содержит историю владения, с которой можно ознакомиться, например, на специализированных сайтах.

Распределенная база данных blockchain формируется как непрерывно растущая цепочка блоков с записями обо всех транзакциях. Копии базы или ее части одновременно хранятся на множестве компьютеров и синхронизируются согласно формальным правилам построения цепочки блоков. Информация в блоках не шифрована и доступна в открытом виде, но отсутствие изменений удостоверяется криптографически через хэш-цепочки (элемент цифровой подписи) [4].

База публично хранит в незашифрованном виде информацию обо всех транзакциях, подписываемых с помощью асимметричного шифрования. Для предотвращения многократной траты используются метки времени, реализованные путем разбиения БД на цепочку специальных блоков, каждый из которых, в числе прочего, содержит в себе хеш предыдущего блока и свой порядковый номер. Каждый новый блок осуществляет подтверждение транзакций, информацию о которых содержит и дополнительное подтверждение транзакций во всех предыдущих блоках цепочки. Изменять информацию в блоке, который уже находится в цепи, непрактично, так как в таком случае пришлось бы редактировать информацию во всех последующих блоках. Таким образом, успешная double-spending атака на практике крайне маловероятна.

Попадание транзакции в блок является подтверждением ее достоверности вне зависимости от наличия других транзакций. Каждый новый блок считается дополнительным «подтверждением» транзакций из предыдущих блоков. Если в цепочке три блока, то транзакции из последнего блока будут подтверждены один раз, а помещенные в первый блок будут иметь три подтверждения. Достаточно дождаться нескольких подтверждений, чтобы свести вероятность отмены транзакции к минимуму.

Если контролировать более 50 % суммарной вычислительной мощности сети, то существует теоретическая возможность при любом пороге подтверждений одни и те же биткойны

передать два раза разным получателям – одна из транзакций будет публичной и подтверждаться в общем порядке, а вторая не будет афишироваться, ее подтверждения будут происходить блоками скрытой параллельной ветви. Лишь через некоторое время сеть получит сведения о второй транзакции, она станет подтвержденной, а первая утратит подтверждения и будет игнорироваться.

Открытость цепочки блоков позволяет внести в произвольный блок изменения. Но тогда потребуется пересчет хеша не только измененного блока, но и всех последующих. Фактически, для такой операции потребуется мощность не меньше той, которая была использована для создания измененного и последующих блоков (то есть всей текущей мощности), что делает такую возможность крайне маловероятной.

Концепция построения PKI на базе технологии blockchain

Частично данная идея нашла воплощение в протоколе безопасности CSMP на основе технологии blockchain, предложенном авторами приложения Crypviser [3].

Модель аутентификации, основанная на технологии blockchain, позволяет пользователям действительно идентифицировать и подтверждать открытые ключи друг друга. Это исключает угрозу «человек посередине» и попытки манипуляции любого рода как на стороне сервера, так и со стороны третьих лиц.

Идея основана на способности технологии blockchain к децентрализованному распространению и управлению открытыми ключами. Поскольку blockchain представляет собой децентрализованную базу данных, она содержит информацию о соответствии уникального идентификатора каждого пользователя и первую половину его открытого ключа *(id: first_half (PK))*. Сервер Crypviser (CV-сервер) содержит информацию о соответствии между уникальным ID пользователя и значением второй половины его открытого ключа *(id: second_half (PK))*.

Первоначальная аутентификация. В ходе регистрации учетной записи приложение Crypviser генерирует уникальное значение ID и первоначальный секретный SK (Shared Key – общий ключ) на локальном устройстве пользователя. Открытый ключ является производным из SK. Эти ключи постоянно используются лишь в целях первоначальной идентификации. CV-сервер, действующий в качестве узла blockchain, имеет собственную пару ключей.

Регистрация учетной записи. По завершении процесса регистрации новая пара ключей генерируется на устройстве пользователя, связанном с учетной записью. В то же время уникальный хеш *CrypID* генерируется на различных источниках энтропии, например частичные хеши симметричного ключа, используемого для защиты локальной базы данных и парольной фразы пользователя. Вторая часть открытого ключа, *CrypID* и идентификатор пользователя *(id: second_half (PK):CrypID)* передаются на CV-сервер с помощью установленного безопасного соединения. Идентификатор пользователя, который был сгенерирован на этапе первоначальной аутентификации, предназначен для обеспечения анонимности на стороне CV-сервера.

Интеграция blockchain. Для записи первой половины первоначального открытого ключа в blockchain пользователь проводит транзакцию (отправляет токены для аутентификации) в пользу CV-сервера. Транзакция содержит метаданные со значением первой половины *first_half* (PK – открытый ключ), которые прописываются в реестре blockchain. После этого записанная часть открытого ключа пользователя может быть проверена на CV-сервере и на сторонах пользователей для исключения попыток атаки «человек посередине» при передаче половины открытого ключа через сеть.

Лишь владелец секретного ключа может «потратить» токен, решив специальную криптографическую «задачу», связанную со сложными расчетами с помощью CrypID. Это означает, что CV-сервер обеспечивает действие первой части открытого ключа пользователя, прописанного в blockchain.

Для подтверждения и проверки первой части открытого ключа, прописанного в blockchain на стороне пользователя, CV-сервер аналогичным образом отправляет пользователю токены для аутентификации. Приложение Crypviser выполняет аналогичные алгоритмы для проверки аутентичности записанной части своего открытого ключа.

Таким образом, CV-сервер и пользователь одновременно проверяют подлинность половины первоначального открытого ключа пользователя. Безопасность частичного значения открытого ключа, записанного в реестре, обеспечивается другими узлами с помощью функции распределения данных.

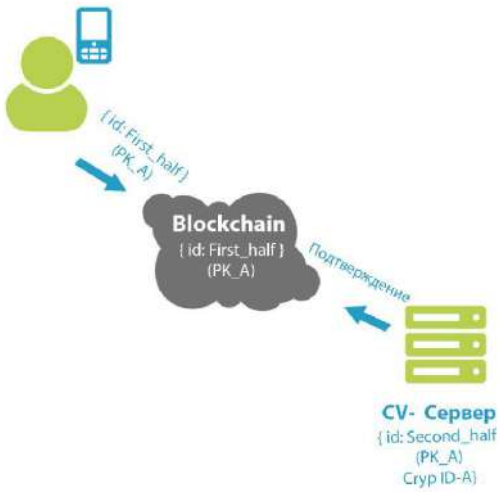


Рис. 5. Распространение открытого ключа [3]

Аутентификация открытого ключа. Алгоритм распространения и проверки достоверности открытого ключа между сторонами описан ниже:

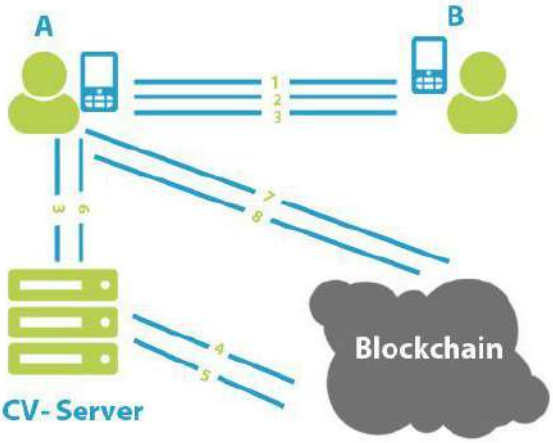


Рис. 6. Аутентификация открытого ключа

1. Сторона А хочет инициировать новый сеанс связи с криптографической защитой и отправляет сообщение следующего содержания:

(Nonce_A, timestamp_A)

2. Сторона В отправляет следующий ответ:

(Nonce_B, timestamp_B,E[(timestamp_A, Nonce_A, id_B, hash(id:PK_B))SK_B]CrypID_B)SK_B,

где *E[(timestamp_A, Nonce_A,id_B, hash(id:PK_B))SK_B]CrypID_B* – вывод зашифрованных данных с CrypID, принадлежащим Стороне В.

Важно отметить, что данные сначала подписываются ключом Стороны В, а затем шифруются.

3. Пользователь А получает сообщение и пересылает его как запрос CV-серверу через защищенный TLS-канал:

$$E[(timestamp\_A, Nonce\_A, id\_B, hash(id: PK\_B))SK\_B]CrypID\_B),$$

4. CV-сервер с CryptID Стороны В расшифровывает зашифрованный текст и проверяет соответствие идентификатора Стороны В, сохраненного локально, его ID, полученному из дешифрованных данных;

5. Затем он получает первую часть открытого ключа из реестра blockchain по идентификатору Стороны В и объединяет ее со второй частью открытого ключа, хранящейся в локальной базе данных, а также проверяет цифровую подпись полученных данных.

6. CV-сервер подписывает сообщение своим секретным ключом, шифрует его с помощью CrypID Стороны А и отправляет следующие данные Стороне А:

$$E[(\text{‘}OK\text{’}, timestamp\_A, Nonce\_A, id\_B, hash(id: PK\_B),$$

$$second\_half(PK\_B))SK\_S]CrypID\_A.$$

7. Получив сообщение, Сторона А выполняет следующие действия:

- дешифрует данные, полученные с CV-сервера, и проверяет цифровую подпись;
- сравнивает значения timestamp_A и Nonce_A с ранее отправленными Стороне В. Значение timestamp_A должно находиться в допустимом временном интервале, а значение Nonce_A должно совпадать.

8. Сторона А запрашивает blockchain и получает первую часть открытого ключа, которая принадлежит Стороне В.

- получает весь открытый ключ Стороны В, объединив полученную часть открытого ключа от blockchain и CV-сервера
- проверяет цифровую подпись всего пакета, полученного от пользователя В ранее;
- рассчитывает хеш *(id:PK_B)*) и сравнивает его со значением хеша данных, полученных от CV-сервера.

9. В случае успешного завершения всех проверок Сторона А считает, что PK_B изначально принадлежит Стороне В. Затем Сторона А отправляет Стороне В следующее сообщение:

$$E[(timestamp\_B, Nonce\_B, id\_A, hash(id:PK\_A))SK\_A]CrypID\_A$$

Сторона В использует тот же алгоритм для получения и аутентификации открытого ключа стороны А.

Таким образом, мы видим, что CV-сервер по сути выступает в роли третьей доверенной стороны, однако его функции резко сокращены по сравнению с функциями третьей доверенной стороны в системе, основанной на модели доверия строгой иерархии.

Кроме того, с использованием такого решения исключается возможность атаки типа «man in the middle».

«Отказ» от третьей доверенной стороны

Технология blockchain, на наш взгляд, способна самостоятельно обеспечивать доверие в системе.

По сути, blockchain – это журнал с фактами (реестр фактов), реплицируемый на несколько компьютеров, объединенных в сеть равноправных узлов (P2P). Фактами может быть что угодно, от денежных операций и до подписания контента. Члены сети – анонимные лица, называемые узлами. Все коммуникации внутри сети используют криптографию, чтобы надежно идентифицировать отправителя и получателя. Когда узел хочет добавить факт в журнал, в сети формируется консенсус, чтобы определить, где этот факт должен появиться в журнале; этот консенсус называется блоком.

Эта идея может быть воплощена для реализации инфраструктуры открытых ключей без построения строгой иерархии уполномоченных на сертификацию. Что, в свою очередь, резко снизит расходы на содержание громоздкой системы иерархической структуры.

Основные принципы децентрализованной ИОК.

1. Каждый пользователь (пользователь выступает узлом) хранит свою ключевую пару самостоятельно. Сертификат открытого ключа передается вместе с подписанным сообщением.
2. Запись о транзакции по законам blockchain хранится в распределенной базе.
3. Блок транзакций содержит реестр состояний сертификата.
4. При проверке правильности транзакции (фактически действительности сертификата открытого ключа) проверяющему необходимо проследить реестр состояния сертификата отправителя вплоть до его первой публикации (аналогичные действия проходят в системе Bit-Coin для проверки наличия «средств» на «счету» клиента, т.е. исключение «двойного расходования»).
5. Первичная идентификация нового пользователя является обязательной и должна быть надежно подтверждена. Для этой и только для этой цели необходим доверенный узел (аналог уполномоченного на сертификацию в иерархической структуре). Его роль будет состоять в первичном выпуске сертификата нового пользователя, а также в случаях, необходимых для изменения статуса сертификата. После первой транзакции, совершенной этим пользователем, обращение к доверенному узлу больше не возникает, кроме ситуаций, требующих изменения статуса сертификата данного пользователя. Т. е. данный узел будет обеспечивать новых пользователей «родительским» блоком («genesis block»), для того, чтобы уже существующие узлы могли проверять статусы сертификата нового пользователя. Целесообразным представляется возложить эту роль на государственную структуру.

Введем следующие условные обозначения:

M – сообщение,

$Sign$ – цифровая подпись отправителя,

H- криптографическая хеш-функция,

$Sert$ – сертификат открытого ключа отправителя,

ID – уникальный идентификатор отправителя, выданный ему при первичной идентификации,

$Status$ – статус сертификата открытого ключа отправителя,

$Sign$ – цифровая подпись.

Первичная идентификация пользователя. Как уже упоминалось, первичная идентификация должна проводиться государственной структурой (доверенным узлом). При обращении к которой пользователю будет выдан его уникальный идентификатор ID и соответствующий ему сертификат открытого ключа $Sert$. Следует отметить, что доверенный узел не хранит у себя ID пользователя, более того, он его не знает.

Первая транзакция нового пользователя должна быть обращена к доверенному узлу для того, чтобы последующие могли ссылаться на нее по законам blockchain. Так как для надежного подтверждения транзакции необходимо вычисление 3 – 5 блоков, следующих за блоком с данной транзакцией, рекомендуется оправлять транзакцию не только к одному представителю доверенного узла, а к нескольким (оператор регистрации, оператор сертификации, администратор безопасности).

После прохождения первичной идентификации данные распространяются в распределенную базу данных, в которой они хранятся в следующем виде:

Данные, хранящиеся в виде таблицы в распределенной БД (blockchain)

$H(Sert,ID)$	$H(Sert,Status)$	$Status$

Алгоритмы формирования и проверки подписи. Алгоритм формирования подписи не отличается от существующего и зависит только от типа используемой подписи.

Алгоритм проверки подписи в ИОК на основе тенологии blockchain

Отправитель генерирует транзакцию:

$$M;\ Sign;\ H(Sert, ID);\ Sert;\ Status \quad (1)$$

Ниже приведен вариант адаптированного протокола:

1. {«hash»:« »,
2. «ver»:1,
3. «vin_sz»:1,
4. «vout_sz»:1,
5. «lock_time»:0,
6. «size»: ,
7. «in»:[
8. {«prev_out»:
9. {«hash»:« …»,
10. «n»:0},
11. «scriptSig»:«… …»}],
12. «out»:[
13. {«value»:« »,
14. «scriptStatus»:« »}]}

Здесь строка 1 содержит хэш оставшейся части транзакции, выраженной в шестнадцатеричном виде. Это используется в качестве идентификатора транзакции.

Строка 2 указывает на версию протокола.

Строки 3 и 4 указывают на то, что транзакция имеет один ввод и один вывод соответственно.

Строка 5 содержит значение lock_time, которое может быть использовано для контроля, когда транзакция будет завершена. Если lock_time установлен в 0, это означает, что транзакция немедленно завершена.

Строка 6 содержит размер (в байтах) транзакции.

Строки с 7 по 11 определяют входные данные к операции. В частности, строки с 8 по 10 говорят нам, что ввод должен быть взят с вывода из предыдущей сделки с соответственной хэш-суммой, выраженной в шестнадцатеричном формате, n = 0 указывает на то, что это будет первый вывод из той транзакции. Строка 11 содержит подпись отправителя, затем пробел, а затем соответствующий открытый ключ в шестнадцатеричном формате.

Строки с 12 по 14 определяют выходные данные. В частности, строка 13 говорит нам о значении вывода: статус сертификата. Строка14 это отображение языка сценариев и адреса строки.

Алгоритм проверки состоит из двух этапов:

I этап заключается в проверке цифровой подписи $Sign$ на основании сертификата открытого ключа отправителя $Sert$. Если эта проверка выполнена успешно (т.е. цифровая подпись наложена именно при помощи личного ключа, которому соответствует предоставленный сертификат открытого ключа отправителя), необходимо переходить к этапу 2 для удостоверения, что данный сертификат открытого ключа отправителя действительно принадлежит отправителю.

II этап состоит из таких шагов:

1. Получаем значение и адрес поля $Status$ из таблицы на основании полученного от отправителя $H(Sert, ID)$;
2. Вычисляем $H'(Sert,\ Status)$;

3. Получаем значение и адрес поля $Status^{/}$ из таблицы на основании $H^{/}(Sert,\ Status)$;

4. Если значение и адрес $Status = Status^{/}$, проверка считается успешной.

Предложенная выше система обладает рядом преимуществ:

– значительное снижение затрат на содержание громоздкой иерархической структуры уполномоченных на сертификацию;

– пользователь самостоятельно контролирует свои идентификационные данные и способен немедленно сообщить о необходимости их корректировки (компрометации);

– нивелирование угрозы «man in the middle»;

– «исчезновение цели» для направленной атаки. В отличие от иерархической структуры, когда главными мишенями для злоумышленников были центры сертификации ключей, в данном случае отсутствует явная цель для атаки, т.к. записи хранятся распределенно и, по сути, злоумышленник вынужден атаковать всю сеть целиком, а не конкретный узел;

– система может быть использована не только непосредственно для услуги электронной подписи, но и для обеспечения электронной идентификации граждан;

– выход из строя одного или нескольких узлов не приводит к остановке системы;

– отсутствие необходимости делать и хранить резервные копии;

– интероперабельность системы заключается в том, что сертификаты, выпущенные различными уполномоченными на сертификацию, могут легко использоваться в единой системе;

– легкая масштабируемость, т.к. добавление нового пользователя (нового узла) происходит без изменений основных принципов функционирования архитектуры.

Выводы

1. Анализ показал, что стойкость ИОК на базе технологии blockchain будет превышать стойкость централизованной системы. Следует понимать, что речь идет не о криптостойкости, а именно о резильентности системы в целом.

2. Энергетические затраты, необходимые для реализации атаки на систему, будут составлять 50 % от вычислительной мощности такой системы. Нарушителю необходимо будет атаковать всю систему целиком. Соответственно, для того, чтобы иметь 50 %-й шанс на успех в решении одного блока, ему необходимо будет располагать вычислительной мощностью равной вычислительной мощности всей остальной системе. Кроме того, рекомендация 3 – 5 ступенчатого подтверждения резко и значительно снижает его шансы, т.к. для этого ему будет необходима вычислительная мощность, которая существенно превышаюшает вычислительную мощность всей системы. Таким образом, стойкость системы повышается с ростом числа узлов (пользователей).

3. В существующей ИОК данная концепция позволит ликвидировать проблему совместимости сертификатов, выпущенных различными уполномоченными на сертификацию.

4. Применение изложенного подхода позволит облегчить переход на новые алгоритмы подписей, в частности на постквантовые, в которых стойкость зависит не от криптопериода ключа (3 года, 5 лет), а от количества наложенных подписей (например в hash based подписях). Исходя из этого, blockchain технология позволит более рационально управлять сертификатами открытых ключей.

5. Более того, для некоторых алгоритмов hash based подписей (таких как подписи Merkle, XMSS) взможны упрощения алгоритмов (исключения из протокола сертификата открытого ключа и использование вместо него пути аутентификации), что позволит уменьшить объем передаваемых данных, т.к. путь аутентификации должен быть передан в любом случае согласно алгоритмов подписи Merkle, XMSS.

Список литературы:

1. ISO/IEC 9594-8 ITU-T Rec. X.509 «Основные положення сертификации ключем и сертификации атрибутов».

2. Инфраструктура открытых ключей: технологии, архитектура, построение и внедрение : учеб. пособие / А.В. Потий, А.В. Леншин, Л.С. Сорока, В.И. Есин, Б.И. Мороз. Днепропетровск : Академия пограничной службы Украины, 2011. 202 с.

3. CrypViser GmbH Whitepaper

4. Michael Nielsen How the Bitcoin protocol actually works http://www.michaelnielsen.org/ddi/how-the-bitcoin-protocol-actually-works/

Харьковский национальный
университет имени В.Н.Каразина;
Харьковский национальный
университет радиоэлектроники

Поступила в редколлегию 05.03.2018

УДК 004.056.55

О.О. КУЗНЕЦОВ, д-р техн. наук, В.О. ФРОЛЕНКО, Є.С. ЄРЬОМІН,
Д.В. ІВАНЕНКО, канд. техн. наук

ДОСЛІДЖЕННЯ КРОСПЛАТФОРМНИХ РЕАЛІЗАЦІЙ ПОТОКОВИХ СИМЕТРИЧНИХ ШИФРІВ

Вступ

На сьогоднішній день інформаційні технології використовуються майже у всіх сферах нашого життя. Дуже широке розповсюдження отримав Інтернет речей (Internet of Things, IoT), майже у кожної людини є персональній мобільний пристрій. У зв'язку з цим з'явилось декілька проблем. По-перше, потрібно забезпечити роботу програмних засобів на всіх пристроях, тобто забезпечити кросплатформність реалізації відповідних сервісів та інформаційних служб. По-друге, необхідно забезпечити безпеку мобільних пристроїв та вирішити питання їх недостатньої потужності. Тому, актуальним завданням є дослідження різних криптографічних механізмів захисту інформації з можливістю кросплатформної реалізації, оцінка швидкодії в умовах обмежених обчислювальних ресурсів при портуванні на різні мобільні пристрої та операційні системи.

Мета роботи – аналіз та порівняльні дослідження сучасних симетричних потокових криптоперетворень, зокрема тестування їх швидкодії при кросплатформовій реалізації мовою програмування Java. Тестування проводилися на операційних системах Windows 10 (x64), Debian (Kali), Android (x64) на різних обчислювальних системах.

Досліджувані симетричні шифри та умови тестування

При проведенні експериментальних досліджень були розглянуті сучасні потокові симетричні шифри Enocoro [1 – 3], Decim [1, 4, 5], Grain [6], HC [6], MUGI [1], Mickey [6], Rabbit [1, 6], RC-4 [7, 8], Salsa20 [6], SNOW2.0 [1], Sosemanuk [6], Strumok [9, 10], Trivium [2, 6], а також алгоритм блокового шифрування AES [11, 12], який може застосовуватися у потокових режимах шифрування. Метою досліджень є визначення швидкості шифрування на різних обчислювальних платформах при кросплатформовій реалізації мовою програмування Java.

Потоковий симетричний шифр Enocoro – апаратно-орієнтований криптоалгорим, який описано у [1 – 3]. Це байт-орієнтований шифр із довжиною ключа 128 біти та вектору ініціалізації 64 біти. Незважаючи на те, що Enocoro є апаратно орієнтованим шифром, він також має і ефективну програмну реалізацію. Для досягнення різних вимог, використовуються байтові операції.

Потоковий симетричний шифр Decim – спеціалізований для апаратного застосування алгоритм був розроблений Комом Бербаїном, Олівером Біллетом, Анн Канту, Николя Куртуа, Бландіном Дебре, Генрі Гильбертом, Луї Губином, Аліном Гуже, Луї Гранбуланом, Седериком Ларду, Марином Мінье, Томасом Порніним та Ервом Сибе [4]. Це апаратно орієнтований потоковий шифр з 80-бітним ключем та 64-бітним вектором ініціалізації (IV), який було подано до проекту потокового шифрування eSTREAM (не пройшов далі третього етапу конкурсу). Конструкція Decim заснована на нелінійному фільтрі регістру зсуву з лінійними зворотними зв'язками (РЗЛЗЗ, англ. LFSR) та нерегулярному механізмі розрідження псевдовипадкових послідовностей, який називають ABSG. Як наслідок, Decim має низьку апаратну складність. Після виявлення певних недоліків в [5] цей алгоритм було вдосконалено в нова версія Decim, що названо Decim^{v2}, виглядає більш безпечною, крім того, має меншу складність апаратної реалізації, ніж попередня версія Decim [1].

Потоковий симетричний шифр Grain, який було представлено Мартіном Хеллом, Томасом Юханссоном та Віллі Мейєром у 2004 на міжнародному конкурсі eSTREAM за другим профілем (апаратно орієнтовані шифри) [6]. Симетричний алгоритм синхронного поточного шифрування, який орієнтований на використання на обчислювальних машинах з обмеженою

кількістю вентилів (gate), невеликими потужністю та обсягом пам'яті. В залежності від апаратної реалізації шифр Grain може бути біт-орієнтованим або слово-орієнтованим. В Grain v1 на вхід подається ключ довжиною 80 біт та вектор ініціалізації довжиною 64 біти. В основі конструкції алгоритму лежать два регістри зсуву – з лінійним та нелінійним зворотним зв'язком та вихідна функція. Рекомендована довжина ключового потоку, який може бути вироблено на одній парі ключ/вектор, – 2^{44} біт.

Потоковий симетричний шифр HC-256, який було розроблено у 2004 році [6]. HC-256 простий, безпечний, програмно-орієнтований шифр з ефективною реалізацією і може вільно використовуватися. Спрощену версію HC-128 було представлено на eSTREAM у першому профілі. Для ініціалізації використовується 256-бітний ключ та вектор ініціалізації довжиною 256 біт. Рекомендована максимальна довжина ключової послідовності – 2^{128}.

Потоковий симетричний шифр MUGI є генератором ключових потоків, який було рекомендовано проектом CRYPTREC для використання у 2003 році урядом Японії. Алгоритм стандартизовано у ISO/IEC 18033-4 [1]. У якості початкових даних MUGI використовує 128-бітовий секретний ключ, 128-бітовий вектор ініціалізації. MUGI використовує нелінійні блоки підстановки та лінійні трансформації з використанням MDS матриці алгоритму AES. Основні конструкції шифру подібні до конструкцій шифру Panama. Шифр MUGI є слово-орієнтованим.

Потоковий симетричний шифр Mickey, вдосконалену версію 2.0 якого було представлено у 2005 році Стивом Беббіджем та Метью Доддом [6] (розшифровується як Mutual Irregular Clocking KEYstream generator – генератор ключового потоку із взаємно нерівномірним рухом). Його призначено для апаратних платформ з обмеженими ресурсами, тобто потоковий шифр MICKEY був розроблений за другим профілем, як апаратно-орієнтований шифр. Для ініціалізації початкового стану використовуються ключ довжиною 80 біт та вектор ініціалізації довжиною до 80 біт. Максимально можлива довжина ключового потоку дорівнює 2^{40} біт на одному ключі, але з використанням різних векторів ініціалізації одної довжини. Алгоритм шифрування MICKEY має просту апаратну реалізацію, але при цьому забезпечує високий рівень безпеки. Завдяки використанню нерегулярного руху регістрів зсуву, а також нових методів забезпечується висока стійкість до певних криптоаналітичних атак.

Потоковий симетричний шифр Rabbit, розробниками алгоритму є Мартін Боесгаард, Метте Вестерагер, Томас Педерсен, Йеспер Крістіансен та Ове Скавиньюс [6]. У травні 2005 року, цей шифр був представлений на конкурсі eStream у першому профілі – програмно-орієнтовані алгоритми. Алгоритм використовує 128-бітний ключ і 64-бітний вектор ініціалізації. На одній парі ключ/вектор може бути вироблено до 2^{67} бітів ключового потоку. Стандартизовано у ISO/IEC 18033-4 [1].

Потоковий симетричний шифр RC-4 був створений Рональдом Рівестом, співробітником компанії «RSA Security», в 1987 році. Скорочення «RC4» офіційно позначає «Rivest cipher 4» або «шифр Рівеста» («4» – номер версії) [7]. Протягом семи років шифр був комерційною таємницею, і точний опис алгоритму надавався тільки після підписання угоди про нерозголошення, але у вересні 1994 року його опис було анонімно відправлено в список розсилки «Cypherpunks» [8]. Володарі легальних копій вихідного коду RC4 підтвердили ідентичність алгоритмів при розбіжностях в позначеннях і структури програми.

Потоковий симетричний шифр Salsa 20, який було розроблено Даніелем Бернштейном [6]. Алгоритм став переможцем конкурсу eSTREAM в першому профілі (програмно-орієнтовані алгоритми). Для ініціалізації внутрішнього стану використовується ключ довжиною 256 біт, 64-бітний nonce та 64-бітна позиція блоку ключового потоку. Максимальна довжина псевдовипадкової ключової послідовності дорівнює 2^{70} біт.

Потоковий симетричний шифр SNOW 2.0 є генератором ключових потоків [1], який використовує як вхідні дані 128 або 256-бітовий секретний ключ *K* і 128-бітовий вектор ініціалізації *IV*. Шифр є слово-орієнтованим. Автори алгоритму – Томас Йохансон та Патрік Екдаль. Алгоритм було стандартизовано у ISO/IEC 18033-4. Для SNOW 2.0 максимально реко-

мендовану кількість біт ключового потоку, виробленого на одній парі *(K, IV)* дорівнює $23*2^{50}$ біт. Це обмеження виправдане з точки зору забезпечення стійкості алгоритму проти криптоаналітичних атак.

Потоковий симетричний шифр Sosemanuk – це синхронний програмно-орієнтований потоковий шифр, який відповідає першому профілю конкурсу eECRYPT [6]. Його довжина ключа може бути обрана між 128 і 256 бітами. Шифр працює з 128 бітовим початковим значенням, при цьому, як стверджується розробниками алгоритму, будь-яка довжина ключа досягає 128-бітного захисту. Алгоритм Sosemanuk використовує деякі основні принципи потокового шифру SNOW 2.0 і деякі перетворення, отримані з блокового шифру SERPENT.

Потоковий симетричний шифр Strumok вперше представлений в [9, 10]. В основі алгоритму лежить класична схема підсумовуючого генератора [1, 6], подібна генератору SNOW-2.0, який визначено в ISO/IEC 18033-4:2011 [1]. В останній редакції алгоритм Струмок використовує 256-бітний вектор ініціалізації IV та 256-бітний або 512-бітний секретний ключ K і забезпечує високий та надвисокий рівень стійкості із врахуванням можливого застосування квантового криптографічного аналізу. Криптоалгоритм орієнтований на 64-розрядні обчислювальні системи, отже розмір слова визначено рівним 64 бітам.

Потоковий симетричний шифр Trivium – це симетричний апаратно-орієнтований паралельний потоковий шифр. Авторами шифру є Крістоф Де Канн'єр и Барт Пренел [6]. Trivium найбільш простий шифр проекту eSTREAM (другий профіль), який демонструє відмінні результати криптостійкості. За специфікацією алгоритм Trivium – це паралельний потоковий шифр, призначений для генерації 2^{64} біт ключового потоку з 80 біт секретного ключа і 80 біт вектору ініціалізації. Шифр є біт-орієнтованим.

Блоковий симетричний шифр AES, який стандартизовано в США як FIPS-197 [11]. На міжнародному рівні стандартизовано у ISO/IEC 18033-3 [12]. Використовує ключ довжиною 128, 192 або 256 біт. В залежності від довжини ключа відбувається 10, 12 або 14 раундів шифрування. AES базується на принципі, відомому як мережа замін-перестановок та, завдяки цьому, має швидку апаратну та програмну реалізацію. У режимі зворотного зв'язку за виходом цей шифр можна використовувати як потоковий.

Перелік досліджуваних алгоритмів наведено у табл. 1, де вказано короткі відомості про шифри та належність до відповідних стандартів чи дослідницьких проектів.

Таблиця 1

Криптоалгоритми, обрані для порівняння

Назва шифру	Джерело специфікації	Розмір стану, біт	Розмір ключа, біт	Розмір IV, біт
AES	FIPS-197, CRYPTREC, ISO/IEC 18033-4	128	128, 256	256
Enocoro	ISO/IEC 29192-3	272	80, 128	64
DECIMv2	ISO/IEC 18033-4, eSTREAM	288	128	128
GRAIN	eSTREAM	128	128	96
HC	eSTREAM	128, 256	128, 256	128, 256
MUGI	ISO/IEC 18033-4	128	128	128
MICKEY	eSTREAM	160	128	128
Rabbit	ISO/IEC 18033-4, eSTREAM	513	128	64
RC4	Список розсилки Cypherpunks	256	256	–
SALSA-20	eSTREAM	512	128	64
SNOW2.0	ISO/IEC 18033-4	512	128, 256	128, 56
SOSEMANUK	eSTREAM	512	128	128
Струмок-256	Цей документ	1024	256	256
Струмок-512		1024	512	512
TRIVIUM	eSTREAM, ISO/IEC 29192-3	288	80	80

В потокових алгоритмах інформація подається та обробляється у вигляді байт послідовності, де шифрується кожен символ відкритого тексту незалежно від інших символів [2]. Отже, важливими показниками потокових симетричних шифрів, які вимірюються за обраною методикою, є наступні:

– швидкість шифрування довгих потоків;
– швидкість шифрування коротких пакетів;
– швидкість ініціалізації ключових параметрів.

Для тестування було використано різні обчислювальні системи:

– переносний персональний комп'ютер з процесором Intel Pentium 3550m 2.3ГГц, операційна система Windows 10 (x64) та Debian (kali), оперативна пам'ять 4Гб (1600МГц);

– смартфон Samsung Galaxy S7 із функціональністю кишенькового персонального комп'ютера з процесором Samsung Exynos 8890 2.4ГГц, операційна система Android (x64), оперативна пам'ять 4Гб.

За першими двома показниками вимірюється час шифрування та швидкість шифрування за співвідношенням байтів/мікросекунду. При вимірюванні швидкості шифрування для довгих потоків ми повинні вважати на те, що саме цей показник має найбільшу потенційну перевагу над блоковими шифрами, тому, ймовірно, ця цифра буде найважливішим критерієм у більшості додатків [6]. Цікаво визначити, на якій довжині пакетів потоковий шифр почне програвати за швидкістю блоковим шифрам. Отже в дослідженні ми застосовували декілька довжин блоків.

Результати досліджень кросплатформних реалізацій потокових шифрів

*Переносний персональний комп'ютер***.** У табл. 2 - 6 наводяться показники швидкодії різних шифрів при проведенні досліджень на переносному персональному комп'ютері.

За першим критерієм вимірювався час, витрачений на шифрування довгих потоків (1Гбайт). Як видно з табл. 2, найбільшу швидкість при шифруванні довгих пакетів (1Гбайт даних) показують алгоритми Sosemanuk, Strumok та SNOW2.0, найгірші показники отримали шифри Mickey, Decim та Grain.

За другим критерієм вимірюється швидкість шифрування коротких пакетів різної довжини. Для забезпечення репрезентативності досліджень було обрано різні довжини пакетів телекомунікаційного трафіку. Дослідження проводилися шляхом шифрування: 50 пакетів по 1500 байтів; 350 пакетів по 40 байтів; 120 пакетів по 576 байтів. Отримані результати зведено у табл. 3 – 5.

Таблиця 2

Шифрування довгих потоків, Windows 10 (x64)

Назва алгоритму	Час, msec	Швидкість,	
		Bytes/μsec	Мбіт/с
AES-128	93398	11.4965	87.7053
AES-256	87819	12.2266	93.2765
Enocoro	57102	18.8033	143.4653
Decim	3276232	0.3277	2.4576
Grain	1638008	0.6555	4.9152
HC-128	14341	74.8725	571.1825
HC-256	33624	31.9333	243.6037
MUGI	35816	29.9796	228.3547
Mickey	54237364	0.1979	0.0486
Rabbit	19664	54.6036	416.5857
RC-4	320659	3.34855	25.4974
Salsa20	45737	23.4768	179.0923
SNOW2.0-128	23379	45.9274	350.3137
SNOW2.0-256	22071	48.6497	371.0926
Sosemanuk	12096	88.7684	677.1766
Strumok-256	20845	51.5108	392.9043
Strumok-512	21469	50.0134	381.5454
Trivium	1048576	1.0241	7.8028

Таблиця 3

Шифрування коротких пакетів (50 пакетів по 1500 байтів), Windows 10 (x64)

Назва алгоритму	Час, μsec	Швидкість,		
		Bytes/μsec	Мбіт/с	Packets/μsec
AES-128	3952	0.3795	2.8672	0.0126
AES-256	5476	0.2739	2.0480	0.0091
Enocoro	3676	0.4080	3.0720	0.0136
Decim	23534	0.0063	0.0409	0.0002
Grain	14775	0.0101	0.0716	0.0033
HC-128	3087	0.4850	3.6864	0.0160
HC-256	9453	0.1580	1.1264	0.0052
MUGI	29673	0.0505	0.3788	0.0016
Mickey	4474571	0.0003	0.0204	10^{-5}
Rabbit	354	4.2372	32.2564	0.1412
RC-4	6661	0.2251	1.7100	0.0075
Salsa20	33767	0.0444	0.3379	0.0014
SNOW2.0-128	4061	0.3693	2.8164	0.0123
SNOW2.0-256	3894	0.3852	2.9286	0.0128
Sosemanuk	7573	0.1980	1.4336	0.0066
Strumok-256	1434	1.0457	7.8848	0.0348
Strumok-512	1450	1.0344	7.5848	0.0344
Trivium	80356	0.0186	0.1331	0.0006

Таблиця 4

Шифрування коротких пакетів (120 пакетів по 576 байтів), Windows 10 (x64)

Назва алгоритму	Час, μsec	Швидкість,		
		Bytes/μsec	Мбіт/с	Packets/μsec
AES-128	1245	0.4626	3.4816	0.0960
AES-256	7432	0.0775	0.5120	0.0160
Enocoro	6102	0.0943	0.7168	0.0190
Decim	224623	0.0025	0.0204	0.0005
Grain	15835	0.0363	0.2764	0.0075
HC-128	9345	0.0616	0.4608	0.0128
HC-256	25721	0.0223	0.1638	0.0046
MUGI	27349	0.0210	0.1536	0.0043
Mickey	4511735	0.0001	0.0009	$26 \cdot 10^{-5}$
Rabbit	338	1.7041	12.9024	0.3550
RC-4	7238	0.0795	0.6041	0.0016
Salsa20	36527	0.0157	0.1126	0.0032
SNOW2.0-128	5534	0.1040	0.7884	0.0220
SNOW2.0-256	5462	0.1052	0.7987	0.0219
Sosemanuk	3846	0.1497	1.1366	0.0312
Strumok-256	1318	0.4368	3.2768	0.0910
Strumok-512	1335	0.4312	3.2870	0.0898
Trivium	73037	0.0078	0.0512	0.0016

Таблиця 5

Шифрування коротких пакетів (350 пакетів по 40 байтів), Windows 10 (x64)

Назва алгоритму	Час, μsec	Швидкість,		
		Bytes/μsec	Мбіт/с	Packets/μsec
AES-128	765	0.4575	3.4816	0.0522
AES-256	768	0.4557	3.3792	0.0523
Enocoro	1954	0.0020	0.0102	0.0179
Decim	48257	0.0008	0.0061	0.0072
Grain	4753	0.0084	0.0614	0.0735
HC-128	7256	0.0055	0.0409	0.0482
HC-256	15324	0.0026	0.0204	0.0228
MUGI	9571	0.0041	0.0307	0.0365
Mickey	1303412	$3\cdot10^{-5}$	0.0003	0.0002
Rabbit	131	0.3053	2.3244	2.6717
RC-4	1037	0.0385	0.2048	0.0241
Salsa20	97234	0.0004	0.0030	0.0035
SNOW2.0-128	1256	0.0318	0.2355	0.2786
SNOW2.0-256	1187	0.0336	0.2560	0.2948
Sosemanuk	2564	0.0156	0.1126	0.1365
Strumok-256	274	0.1456	1.1059	1.2742
Strumok-512	320	0.1250	0.9523	1.0937
Trivium	15176	0.0026	0.0204	0.0164

Таблиця 6

Ініціалізація ключових параметрів, Windows 10 (x64)

Назва алгоритму	Встановлення ключів		Встановлення векторів ініціалізації	
	Час, sec	Кільк. / µs	Час, sec	Кільк. / µs
AES-128	0.0072	0.9715	$1.16\cdot10^{-4}$	4.3105
AES-256	0.0086	0.8108	$1.25\cdot10^{-4}$	4
Enocoro	0.0075	0.9270	$7.1\cdot10^{-5}$	7.0423
Decim	0.0001	53.435	$5.9\cdot10^{-5}$	8.4745
Grain	0.0010	6.9582	$7.6\cdot10^{-5}$	6.5784
HC-128	1.1679	0.0059	$6.3\cdot10^{-5}$	7.9362
HC-256	4.2364	0.0016	$7.8\cdot10^{-5}$	6.4105
MUGI	0.0246	0.2839	$5.4\cdot10^{-5}$	9.2594
Mickey	10.695	0.0006	0.3347	0.0013
Rabbit	0.0017	3.9230	0.0001	4.1332
RC-4	0.0175	0.3997	-	-
Salsa20	-	-	-	-
SNOW2.0-128	0.0056	1.2289	$4.2\cdot10^{-5}$	11.9041
SNOW2.0-256	0.0018	3.7981	$6.3\cdot10^{-5}$	7.9365
Sosemanuk	0.0053	1.2991	$1.77\cdot10^{-4}$	2.8248
Strumok-256	0.0045	1.5527	$1.1\cdot10^{-5}$	45.4546
Strumok-512	0.0020	3.4163	$4\cdot10^{-6}$	125
Trivium	1.4926	0.0046	$1.52\cdot10^{-4}$	3

Як видно із табл. 3 – 5, при великій довжині пакетів виграш мають потокові шифри Rabbit, Strumok, SNOW2.0 та Enocoro. Але при зменшенні довжини пакетів (до декілька десятків байт) перевагу має, як і очікувалося, блоковий шифр AES.

Швидкість ініціалізації ключових параметрів є найменш критичним параметром для відображення швидкості шифрування. Ці часові витрати зневажливо малі в порівнянні з процесом генерації ключового потоку. При дослідженнях проводилося 7000 ключових установок та 500 установок векторів ініціалізації. Отримані результати тестування показано в табл. 6, вони свідчать про перевагу алгоритмів Decim та Grain. Далі йдуть шифри Rabbit, SNOW2.0, Sosemanuk та Strumok.

У табл. 7 – 11 наводяться показники швидкості шифрів на ОС Debian (kali) за тією ж методикою, що і для ОС Windows 10 (x64).

Таблиця 7

Шифрування довгих потоків, Debian (kali)

Назва алгоритму	Час, msec	Швидкість,	
		Bytes/μsec	Мбіт/с
AES-128	92919	11.7471	11.1616
AES-256	85848	12.9264	12.3187
Enocoro	54865	19.5728	18.6654
Decim	3465528	0.3098	0.2949
Grain	1905016	0.5365	0.51097
HC-128	13438	79.9036	76.1958
HC-256	31326	34.2764	32.6860
MUGI	31257	34.3526	32.7577
Mickey	4861480	0.2200	0.02099
Rabbit	7888	136.1284	1038.43
RC-4	117824	9.1130	8.6906
Salsa20	43850	24.6373	23.4496
SNOW2.0-128	29442	36.4464	34.7545
SNOW2.0-256	29170	36.8092	35.1027
Sosemanuk	18443	58.2197	55.5212
Strumok-256	25414	42.2504	40.2841
Strumok-512	27554	38.9682	37.1609
Trivium	1076536	1.0762	1.0260

Таблиця 8

Шифрування коротких пакетів (50 пакетів по 1500 байтів), Debian (kali)

Назва алгоритму	Час, μsec	Швидкість,		
		Bytes/μsec	Мбіт/с	Packets/μsec
AES-128	2752	0.4995	3.7888	0.0319
AES-256	6275	0.2939	2.1504	0.0097
Enocoro	3176	0.4270	0.3072	0.0147
Decim	440000	0.0034	3.1744	0.0001
Grain	164306	0.0090	0.0061	0.0003
HC-128	2147	0.5340	3.9936	0.0210
HC-256	5253	0.2990	2.2528	0.0112
MUGI	21673	0.1125	0.8192	0.0023
Mickey	3367000	0.0004	0.0030	10^{-4}
Rabbit	347	0.1440	32.972	4.3227
RC-4	74616	0.0201	0.1024	0.0006
Salsa20	34567	0.0432	0.3072	0.0012
SNOW2.0-128	3242	0.4623	3.4816	0.0154
SNOW2.0-256	6212	0.2762	2.0480	0.0098
Sosemanuk	4367	0.3430	2.5600	0.0114
Strumok-256	1869	0.8024	6.0416	0.0267
Strumok-512	1801	0.8325	6.3488	0.0277
Trivium	93863	0.0159	0.1024	0.0005

Таблиця 9

Шифрування коротких пакетів (120 пакетів по 576 байтів), Debian (kali)

Назва алгоритму	Час, μsec	Швидкість,		
		Bytes/μsec	Мбіт/с	Packets/μsec
AES-128	2105	0.4726	3.6044	0.1695
AES-256	8272	0.0795	0.6041	0.0435
Enocoro	5722	0.1143	0.8704	0.0233
Decim	466436	0.0012	0.0092	0.0002
Grain	144000	0.0034	0.0204	0.0008
HC-128	7215	0.0726	0.5529	0.0173
HC-256	17721	0.0632	0.4812	0.0116
MUGI	25349	0.0240	0.1740	0.0049
Mickey	3522000	0.0001	0.0010	$3 \cdot 10^{-5}$
Rabbit	334	1.7245	13.1077	0.3592
RC-4	66432	0.0086	0.0614	0.0018
Salsa20	32527	0.0187	0.1331	0.0072
SNOW2.0-128	1234	0.4540	3.4611	0.0970
SNOW2.0-256	3262	0.1762	1.3414	0.0335
Sosemanuk	7264	0.0756	0.5734	0.0162
Strumok-256	1802	0.3195	2.4371	0.0665
Strumok-512	1680	0.3427	2.6112	0.0714
Trivium	85924	0.0067	0.0409	0.0013

Таблиця 10

Шифрування коротких пакетів (350 пакетів по 40 байтів), Debian (kali)

Назва алгоритму	Час, μsec	Швидкість,		
		Bytes/μsec	Мбіт/с	Packets/μsec
AES-128	785	0.4565	3.4816	0.0532
AES-256	798	0.4587	3.4806	0.0510
Enocoro	1134	0.0060	0.0409	0.0259
Decim	121023	0.0003	0.0020	0.0028
Grain	48210	0.0008	0.0061	0.0072
HC-128	9256	0.0047	0.0307	0.0323
HC-256	13324	0.0031	0.0204	0.0288
MUGI	7531	0.0091	0.0614	0.0645
Mickey	1069000	$3 \cdot 10^{-5}$	0.0002	0.0003
Rabbit	138	0.2898	2.1504	2.5362
RC-4	9466	0.0042	0.0307	0.0264
Salsa20	93164	0.0004	0.0020	0.0032
SNOW2.0-128	843	0.0468	0.3481	0.4786
SNOW2.0-256	2237	0.0176	0.1331	0.1578
Sosemanuk	2763	0.0144	0.1024	0.1265
Strumok-256	404	0.0989	0.7475	0.8656
Strumok-512	361	0.1108	0.8396	0.9695
Trivium	17577	0.0022	0.0102	0.0199

Таблиця 11

Ініціалізація ключових параметрів, Debian (kali)

Назва алгоритму	Встановлення ключів		Встановлення векторів ініціалізації	
	Час, sec	Кількість / µs	Час, sec	Кількість / µs
AES-128	0.0070	0.9940	$1.33 \cdot 10^{-4}$	3.759
AES-256	0.0102	0.6830	$1.52 \cdot 10^{-4}$	3.289
Enocoro	0.0093	0.7470	$8.6 \cdot 10^{-5}$	5.813
Decim	0.0008	8.0275	$1.37 \cdot 10^{-4}$	3.649
Grain	0.0011	5.8873	$9.2 \cdot 10^{-5}$	5.434
HC-128	1.2589	0.0055	$1.28 \cdot 10^{-4}$	3.906
HC-256	4.4324	0.0015	$1.62 \cdot 10^{-4}$	3.086
MUGI	0.0291	0.2405	$7.9 \cdot 10^{-5}$	6.329
Mickey	11.069	0.0006	0.3377	0.001
Rabbit	0.0015	4.5425	0.0001	4.901
RC-4	0.0162	0.4297	-	-
Salsa20	-	-	-	-
SNOW2.0-128	0.0069	1.0060	$6.1 \cdot 10^{-5}$	8.196
SNOW2.0-256	0.0016	4.2296	$5.6 \cdot 10^{-5}$	8.928
Sosemanuk	0.0036	1.9283	$1.65 \cdot 10^{-4}$	3.030
Strumok-256	0.0047	1.4858	$2.5 \cdot 10^{-5}$	20.00
Strumok-512	0.0025	2.7821	$2 \cdot 10^{-5}$	25.00
Trivium	1.0863	0.0064	$1.73 \cdot 10^{-5}$	2.890

Як можна побачити з отриманих результатів, при тестуванні на ОС Debian (kali) більшість шифрів погіршили свої результати. Однак співвідношення швидкодії між окремими шифрами за різними показниками майже такі самі. Слід відмітити, що алгоритм Струмок, який забезпечує високі показники криптографічного захисту, достатні для застосування у постквантовий період (довжина ключа 256 та 512 бітів, довжина вектору ініціалізації 256 бітів, довжина внутрішнього стану 1024 біти), за більшістю показників швидкодії також має певну перевагу, зокрема в більшості випадків він не поступається найкращим світовим аналогам.

Смартфон із функціональністю персонального комп'ютера. Результати досліджень швидкодії потокових шифрів при застосуванні їх на смартфоні із функціональністю персонального комп'ютера наведено у табл. 12 – 16.

Як бачимо із даних табл. 12, за критерієм шифрування довгих пакетів найбільшу швидкість показують алгоритми HC, Rabbit, Salsa20 і Strumok, найгірші показники отримали шифри MUGI, Mickey та Trivium.

За критерієм шифрування коротких пакетів перевагу має шифр Rabbit, далі йдуть шифри AES, Strumok, SNOW2.0 та інші. Найгірші показники мають шифри Decim, Grain, Mickey та Trivium. Але слід відмітити, що загальної тенденції не спостерігається, бо окремі шифри дають нестабільні результати. Можливо, на швидкість шифрування дуже впливає рівень завантаженості обчислювальної системи іншими процесами.

За останнім показником (час ініціалізації ключових даних) перевагу мають шифри Grain і Trivium. Далі йдуть шифри Strumok, Rabbit, SNOW2.0 та інші.

Таблиця 12

Шифрування довгих потоків, Android (x64)

Назва алгоритму	Час, msec	Швидкість,	
		Bytes/μsec	Мбіт/с
AES-128	2185096	0.4913	3.7483
AES-256	920464	1.1665	8.8996
Enocoro	887846	1.2094	9.2269
Decim	$2 \cdot 10^{-7}$	0.0361	0.2755
Grain	$1.6 \cdot 10^{-7}$	0.0644	0.4914
HC-128	23146	46.4825	354.64
HC-256	27157	39.6214	302.28
MUGI	138013696	0.0077	0.0592
Mickey	$3.5 \cdot 10^{-8}$	0.0030	0.0229
Rabbit	38496	27.8924	212.81
RC-4	1081352	0.9929	7.5752
Salsa20	110243	9.7397	74.309
SNOW2.0-128	397753	2.7040	20.629
SNOW2.0-256	457926	2.3447	17.888
Sosemanuk	342231	3.1374	23.937
Strumok-256	186653	5.7526	43.889
Strumok-512	184243	5.8278	44.463
Trivium	$4.9 \cdot 10^{-8}$	0.0021	0.0165

Таблиця 13

Шифрування коротких пакетів (50 пакетів по 1500 байтів), Android (x64)

Назва алгоритму	Час, μsec	Швидкість,		
		Bytes/μsec	Мбіт/с	Packets/μsec
AES-128	10376	0.0385	0.2941	0.3468
AES-256	11467	0.0034	0.0259	0.0925
Enocoro	12199	0.0032	0.0244	0.0286
Decim	725739	$5 \cdot 10^{-5}$	0.0004	0.0004
Grain	2562911	10^{-5}	0.0001	0.0001
HC-128	10341	0.0038	0.0289	0.0340
HC-256	25749	0.0015	0.0114	0.0135
MUGI	22673	0.0017	0.0129	0.0154
Mickey	3528571	10^{-5}	0.0001	$9 \cdot 10^{-5}$
Rabbit	841	0.0475	0.3623	0.4161
RC-4	72537	$5 \cdot 10^{-4}$	0.0042	0.0048
Salsa20	12354	0.0032	0.0244	0.0283
SNOW2.0-128	8573	0.0046	0.0350	0/0408
SNOW2.0-256	5690	0.0070	0.0535	0.0615
Sosemanuk	18573	0.0021	0.0160	0.0188
Strumok-256	6134	0.0065	0.0495	0.0570
Strumok-512	3496	0.0114	0.0869	0.1001
Trivium	5256627	$7 \cdot 10^{-6}$	$5 \cdot 10^{-5}$	$6 \cdot 10^{-5}$

Таблиця 14

Шифрування коротких пакетів (120 пакетів по 576 байтів), Android (x64)

Назва алгоритму	Час, μsec	Швидкість,		
		Bytes/μsec	Мбіт/с	Packets/μsec
AES-128	28453	0.0527	0.4020	0.0017
AES-256	28579	0.0524	0.3990	0.0017
Enocoro	8473	0.1770	1.3504	0.0059
Decim	1728461	$8 \cdot 10^{-4}$	0.0066	$2 \cdot 10^{-5}$
Grain	1264469	0.0011	0.0083	$3 \cdot 10^{-5}$
HC-128	17387	0.0862	0.6573	0.0028
HC-256	20457	0.0733	0.5592	0.0024
MUGI	88463	0.0169	0.1289	$5 \cdot 10^{-4}$
Mickey	$2.2 \cdot 10^{-5}$	$5 \cdot 10^{-5}$	0.0004	10^{-6}
Rabbit	3351	0.4476	3.4142	0.0149
RC-4	2016361	$7 \cdot 10^{-4}$	0.0056	$2 \cdot 10^{-5}$
Salsa20	70610	0.0212	0.1617	$7 \cdot 10^{-4}$
SNOW2.0-128	37261	0.0402	0.3067	0.00134
SNOW2.0-256	38377	0.0390	0.2975	0.0013
Sosemanuk	272471	0.0055	0.0419	10^{-4}
Strumok-256	11012	0.1362	1.0395	0.0045
Strumok-512	14936	0.1004	0.7659	0.0033
Trivium	$1.7 \cdot 10^{-8}$	$8 \cdot 10^{-6}$	$6 \cdot 10^{-5}$	$2 \cdot 10^{-7}$

Таблиця 15

Шифрування коротких пакетів (350 пакетів по 40 байтів), Android (x64)

Назва алгоритму	Час, μsec	Швидкість,		
		Bytes/μsec	Мбіт/с	Packets/μsec
AES-128	30434	0.0189	0.1443	0.0039
AES-256	31644	0.0182	0.1388	0.0037
Enocoro	58575	0.0098	0.0747	0.0020
Decim	1872478	$3 \cdot 10^{-4}$	0.0023	$6 \cdot 10^{-5}$
Grain	1104634	$5 \cdot 10^{-4}$	0.0039	10^{-4}
HC-128	20543	0.0280	0.2136	0.0058
HC-256	25721	0.0223	0.1701	0.0046
MUGI	88286	0.0065	0.0495	0.0013
Mickey	$2 \cdot 10^{-7}$	$2 \cdot 10^{-5}$	0.0001	$4 \cdot 10^{-6}$
Rabbit	2441	0.2359	1.7997	0.0491
RC-4	1224535	$4 \cdot 10^{-4}$	0.0035	$9 \cdot 10^{-5}$
Salsa20	45201	0.0127	0.0968	0.0026
SNOW2.0-128	26982	0.0213	0.1625	0.0044
SNOW2.0-256	27361	0.0210	0.1602	0.0043
Sosemanuk	18699	0.0308	0.2349	0.0064
Strumok-256	17189	0.0335	0.2555	0.0069
Strumok-512	18472	0.0311	0.2372	0.0064
Trivium	$2 \cdot 10^{-7}$	$2 \cdot 10^{-5}$	10^{-5}	$4 \cdot 10^{-5}$

Таблиця 16

Ініціалізація ключових параметрів, Android (x64)

Назва алгоритму	Встановлення ключів		Встановлення векторів ініціалізації	
	Час, sec	Кількість / µs	Час, sec	Кількість / µs
AES-128	0.1554	0.0450	0.0042	0.1180
AES-256	0.2764	0.0253	0.0047	0.1040
Enocoro	0.5824	0.0120	0.0025	0.1970
Decim	0.0257	0.2720	0.0021	0.2367
Grain	0.0005	12.433	0.0001	4.6728
HC-128	0.5714	0.0122	0.0008	0.5590
HC-256	2.8776	0.0024	0.0010	0.4940
MUGI	0.1638	0.0427	0.0 067	0.0740
Mickey	85.916	$8 \cdot 10^{-5}$	0.0027	0.1794
Rabbit	0.3347	0.0209	0.0004	1.1467
RC-4	0.0541	0.1292	-	-
Salsa20	-	-	-	-
SNOW2.0-128	0.1145	0.0611	0.0005	0.8880
SNOW2.0-256	0.1145	0.0610	0.0006	0.7418
Sosemanuk	0.2394	0.0292	0.0197	0.0253
Strumok-256	0.0679	0.1030	0.0004	1.0869
Strumok-512	0.0621	0.1125	0.0004	1.0822
Trivium	0.0066	1.0523	0.0008	0.6090

Висновки

Отримані результати експериментальних досліджень свідчать, що розроблені кросплатформні реалізації потокових шифрів дозволяють застосувати відповідні криптоперетворення на різних обчислювальних платформах. Для цього достатня наявність відповідного інтерпретатору, за допомогою якого вдається забезпечити роботу програмних засобів на різних пристроях, тобто забезпечити кросплатформність відповідних сервісів та інформаційних служб, в тому числі із врахуванням вимог недостатньої потужності та малоресурсності апаратної складової.

Слід відмітити, що кросплатформна реалізація криптоалгоритмів значно зменшує показники швидкодії як при шифруванні довгих потоків, так і при обробці окремих пакетів та ініціалізації ключових даних. Однак співвідношення по швидкодії між окремими шифрами часто зберігаються. Певні зміни цих співвідношень пояснюються особливостями апаратної складової та відповідних обчислювальних алгоритмів, зокрема, окремі шифри орієнтовано на апаратну реалізацію, деякі – на програму із певною розрядністю операційної системи.

Отримані результати будуть корисними в подальшому обґрунтуванні пропозицій для практичного застосування алгоритмів потокового шифрування.

Список литературы:

1. ISO/IEC 18033-4:2011. Information technology – Security techniques – Encryption algorithms – Part 4: Stream ciphers. [Електронний ресурс] URL: http://www.iso.org/iso/home/store/catalogue_ics/ catalogue_detail_ics.htm?csnumber=54532 [Dec., 2012].

2. ISO/IEC 29192-3:2012. Information technology – Security techniques – Lightweight cryptography – Part 3: Stream ciphers. [Електронний ресурс] URL: https://www.iso.org/standard/56426.html 3. Pseudorandom Number Generator Enocoro. [Електронний ресурс] URL: http://www.cryptrec.go.jp/english/cryptrec_13_spec_cypherlist_files/PDF/23_00espec.pdf

4. C. Berbain, O. Billet, A. Canteaut, N. Courtois, B. Debraize, H. Gilbert, L. Goubin, A. Gouget, L. Granboulan, C. Lauradoux, M. Minier, T. Pornin, and H. Sibert. Decim – A new Stream Cipher for Hardware applications. In ECRYPT Stream Cipher Project Report 2005/004. [Електронний ресурс] URL: http://www.ecrypt.eu.org/

5. W.Hongjun and B.Preneel. Cryptanalysis of Stream Cipher Decim. [Електронний ресурс] URL: http://www.ecrypt.eu.org/stream/

6. The eSTREAM Project. [Електронний ресурс] URL: http://www.ecrypt.eu.org/

7. Frequently Asked Questions. [Електронний ресурс] URL: http://people.csail.mit.edu/rivest/faq.html#Ron

8. Thank you Bob Anderson. [Електронний ресурс] URL: http://cypherpunks.venona.com/date/1994/09/msg00304.html

9. Kuznetsov O., Lutsenko M. and Ivanenko D. Strumok stream cipher: Specification and basic properties // 2016 Third International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC *S&T)*, Kharkiv, 2016, pp. 59-62.

10. Gorbenko I., Kuznetsov A., Lutsenko M. and Ivanenko D. The research of modern stream ciphers // 2017 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T), Kharkov, 2017, pp. 207-210.

11. FIPS-197: Advanced Encryption Standard (AES) / National Institute of Standards and Technology, 2001. [Електронний ресурс] URL: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

12. ISO/IEC 18033-3:2010. Information technology – Security techniques – Encryption algorithms. Part 3: Block ciphers. [Електронний ресурс] URL: https://www.iso.org/standard/54531.html

Харківський національний
університет імені В.Н. Каразіна;
Харківський національний
університет радіоелектроніки

Надійшла до редколегії 03.04.2018

УДК 621.37:621.391

В.Н. ШЛОКИН, С.Г. РАССОМАХИН, д-р техн. наук

ВЕРОЯТНОСТНАЯ МОДЕЛЬ ДАКТИЛОСКОПИЧЕСКИХ ОБРАЗОВ КОМПЬЮТЕРНОЙ БИОМЕТРИЧЕСКОЙ АУТЕНТИФИКАЦИИ

Введение

В настоящее время проводятся интенсивные исследования возможностей разработки методов, предназначенных для полноценного использования биометрических характеристик персоналий в реализации защищенных алгоритмов паролирования и дистанционной аутентификации [1]. Стандартизованными для целей аутентификации являются следующие биометрические характеристики персоналий [2, 3]: черты лица (Face features); отпечатки пальцев (Finger Print); радужная оболочка глаза (Iris). В большинстве случаев анализ биометрических образов при их распознавании производится по совокупности значимых характерных точек изображений. При анализе отпечатков пальцев эти характерные точки получили название «минуции», или «точки Гальтона» и являются участками папиллярного рисунка кожи, где отдельные линии сливаются или раздваиваются (бифуркации) и обрываются (окончания). Другими словами, это уникальные для каждого отпечатка пальца точки, в которых изменяется структура папиллярных линий.

Среди биометрических способов идентификации персоналий дактилоскопия занимает особое место. Распределения минуций отдельных реализаций отпечатков могут быть описаны достаточно сложными зависимостями ввиду существенного отличия анализируемых образцов как по возможному числу характерных точек, так и по произвольности их размещения. Виды и функции распределения искажений из-за множественного характера источников причин также достаточно неоднозначны. Выбор между простотой и адекватностью моделей размещения и искажения минуций является компромиссной задачей. При этом, ввиду закрытости существующих алгоритмов распознавания отпечатков, в большинстве случаев отсутствует возможность набора объема статистики размещения и искажения минуций, достаточного для строгого решения задачи моделирования. Поэтому актуальным является эвристический анализ процессов сканирования биометрических образов с учетом их природы и особенностей возникновения возможных ошибок. Это необходимо для создания эффективных методов «нечеткой экстракции» данных для криптографических приложений [5].

Анализ существующих результатов

Большинство существующих систем обработки информации дактилоскопических баз данных используют оригинальные защищенные алгоритмы и программы с закрытым исходным кодом, доступ к которым строго ограничен. Для анализа распределений характеристик и ошибок при обработке данных отпечатков пальцев можно воспользоваться открытой программой SourceAFIS.FingerprintAnalysis.exe [4]. На рис. 1 представлены выбранные исходные изображения отпечатков, использованные для анализа обработки изображений. Данные образцы получены в различных условиях работы сканера и вариациях угла сканирования.

Обработка изображений по алгоритму SourceAFIS [4] дает картину извлеченных минуций, показанную на рис. 2. В зависимости от ориентации образца, смещения сканированного участка, а также контрастности и яркости исходного изображения наблюдаются существенные различия результатов обработки, проявляющиеся в изменяющемся числе обнаруженных минуций и вариациях их взаимного расположения. На представленных портретах окружностями отмечены участки одной и той же области отпечатков, смещенные по кадрам в зависимости от различных углов сканирования и смещения сканируемого оригинала. Как видно, степень похожести изображений чрезвычайно мала. Визуальное сходство наблюдается только в случае достаточно совпадающих условий получения отпечатков (на рис. 2 – это пары портретов 2 и 4 или 3 и 7).

Рис. 1. Примеры реализаций отпечатков, полученных для одного и того же объекта

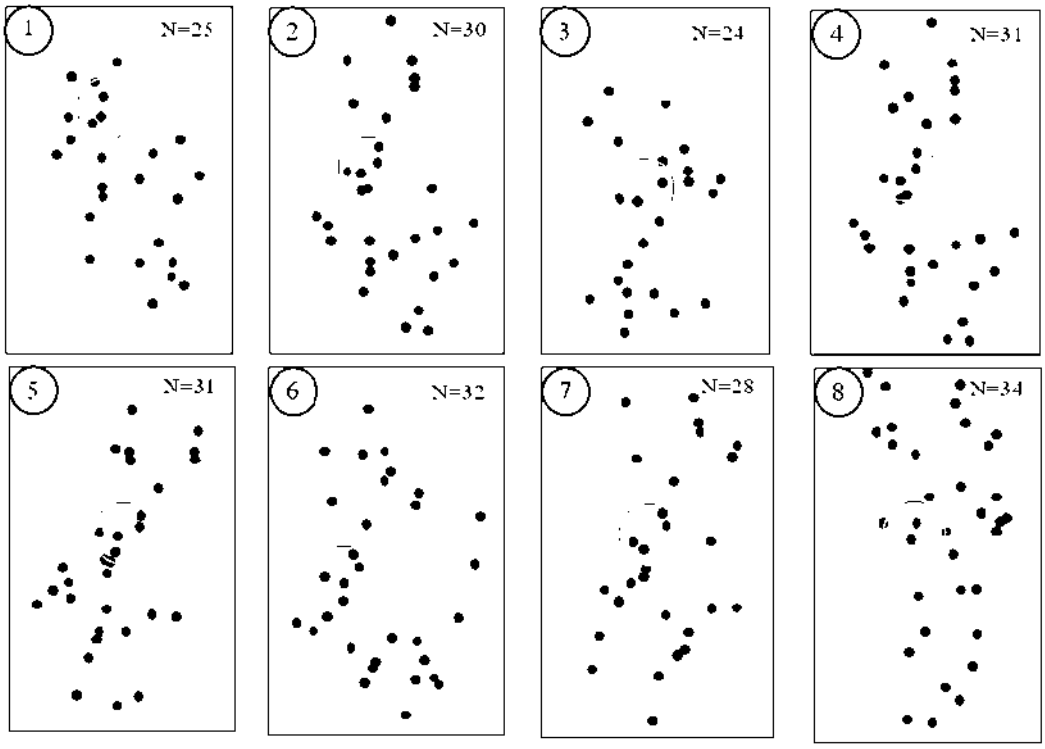


Рис. 2. Портреты плоскостного распределения извлеченных точечных минуций

Как следует из рассмотренного примера, процесс получения и анализа отпечатков на основе различных реализаций дактилоскопических изображений может быть описан достаточно сложной моделью случайного распределения особенностей портретов, а также моделью случайного распределения ошибок отдельных реализаций.

Цель статьи

Цель статьи – разработка вероятностных моделей сканирования и обработки дактилоскопических портретов, а также процессов возникновения ошибок и различий в отдельных реализациях сканирования одних и тех же объектов. Полученные в работе результаты составляют инструментарий для разработки и статистического исследования показателей эффективности новых методов компьютерной биометрической аутентификации.

1. Математическая модель вероятностного распределения характерных точек на портретах отпечатков

Для разработки робастного метода анализа и обработки данных на основе различных реализаций дактилоскопических изображений необходима модель описания случайного распределения минуций и углов прихода папиллярных линий в точки минуций, а также модель случайного распределения ошибок отдельных реализаций. Анализ плоскостных портретов распределений минуций позволяет отметить следующие *особенности*, которые могут быть основой для эмпирического выбора вида распределений точек по координатам кадра при имитации портретов отпечатков пальцев:

– плотность распределения точек вдоль горизонтальной X и вертикальной Y оси имеет примерно равномерный характер в центральной части кадра и незначительно спадает к его краям;

– линейные смещения центра отпечатка по горизонтали и вертикали не означают появления на краях кадра зон, свободных от минуций (в поле сканирования могут попасть новые точки);

– число характерных точек может варьировать для различных персоналий (определено международным стандартом) в пределах 16÷60, а при получении различных реализаций для конкретного субъекта в пределах ±20-25%. Например, на рис. 2 число выявленных минуций одного и того же объекта изменяется от 24-х до 34-х;

– распределение «углов прихода» в характерные точки имеет примерно равномерный характер в диапазоне $[0,\ 2\pi]$.

Для построения модели распределения минуций на основе отмеченных особенностей процесса, будем использовать следующие исходные *предположения*:

– координаты портрета отпечатка X, Y, а также значения углов прихода нормируются в диапазоне $[-0.5; +0.5]$, при этом геометрический центр изображения имеет нулевые координаты на плоскости $[0; 0]$, а сам портрет размещается в единичной квадратной области, охватывающей все четыре квадранта плоскости изображения;

– для первичной генерации случайных чисел, необходимых для получения распределений координат минуций на реализациях портретов отпечатков, используется датчик равномерно распределенных (непрерывно) чисел в диапазоне $[0;1]$: $f(x_i, y_i) \square unif[0,1],\ i \in 1\ldots N$, где N – число минуций на портрете – случайная величина, не выходящая из диапазона $[15; 60]$ с математическим ожиданием $m_N = 25 \div 35$ и унимодальным распределением.

Анализ перечисленных особенностей, а также учет сделанных предположений позволяют использовать для формального описания плотности распределения вероятностей (ПРВ) $f(x)$ и $f(y)$ декартовых координат x, y характерных точек плоскостных портретов зависимость, представленную на рис. 3. Данный вид ПРВ обеспечивает равномерное распределение точек в центральной части единичного квадрата и спадающие вероятности появления точек к границам квадратной области портрета. Область ненулевых значений ПРВ $[-0.75; +0.75]$ выходит на величину 0,25 в обе стороны за пределы единичного квадрата, что обеспечит ненулевую вероятность появления точек в крайних областях портрета при введении ошибок в виде возможного дрейфа геометрического центра. Выбор ПРВ представленного вида не является единственно возможным, однако, на наш взгляд, является вполне приемлемым по сочетанию простоты и отмеченных выше особенностей и предположений**.**

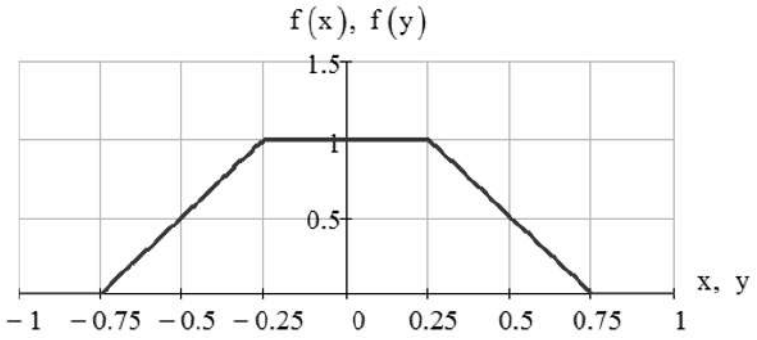


Рис. 3. Функции плотности распределения вероятностей координат минуций

Для получения испытательных образцов портретов размещения минуций необходим источник случайных чисел, распределенных в соответствии с ПРВ $f(x)$, $f(y)$ (рис. 3). Ввиду идентичности распределений вдоль координат плоскости при использовании нормированного единичного квадрата портрета, в дальнейшем будем рассматривать только функцию $f(x)$:

$$f(x)=\begin{cases}2x+1.5 & if \ -0.75\le x<-0.25;\\ 1 & if \ -0.25\le x\le 0.25;\\ -2x+1.5 & if \ 0.25<x\le 0.75;\\ 0 & if \ |x|>0.75.\end{cases} \tag{1}$$

Для генерации случайной величины, подчиненной распределению (1), можно воспользоваться функциональным преобразованием результата стандартного для большинства систем программирования датчика случайных чисел, расположенных непрерывно равномерно в диапазоне $[0,1]$. Воспользуемся методом обратных функций [6]: если $z \square unif[0,1]$, то случайная величина x, получаемая функциональным преобразованием z вида

$$x=\begin{cases}\sqrt{z}-0.75 & if \ 0\le z<0.25;\\ z-0.5 & if \ 0.25\le z\le 0.75;\\ -\sqrt{1-z}+0.75 & if \ 0.75<z\le 1;\end{cases} \tag{2}$$

будет иметь ПРВ вида (1).

На рис. 4 представлена гистограмма статистических испытаний функционального преобразования (2) от при числе опытов, равном 30000 и разбиении интервала на 100 равных подинтервалов. Пунктирной линией на рис. 4 показана огибающая (1). Полученный алгоритм генерации случайных чисел будет использован в дальнейшем для получения координат характерных точек нормированных квадратных портретов отпечатков.

Для генерации случайной величины – числа характерных точек реализации портрета отпечатка выберем использование модели дискретной (целочисленной) случайной величины N из диапазона целых чисел $[15, 45]$ с дискретным нормальным усеченным распределением и числовыми характеристиками:

– математическим ожиданием $m_N \approx 30$;

– среднеквадратическим отклонением $\sigma \approx 2 \div 5$.

Для получения случайного числа минуций на реализации портрета N снова воспользуемся функциональным преобразованием данных датчика $unif[0,1]$. Используем моделирование реализаций случайной величины N на основании центральной предельной теоремы [6].

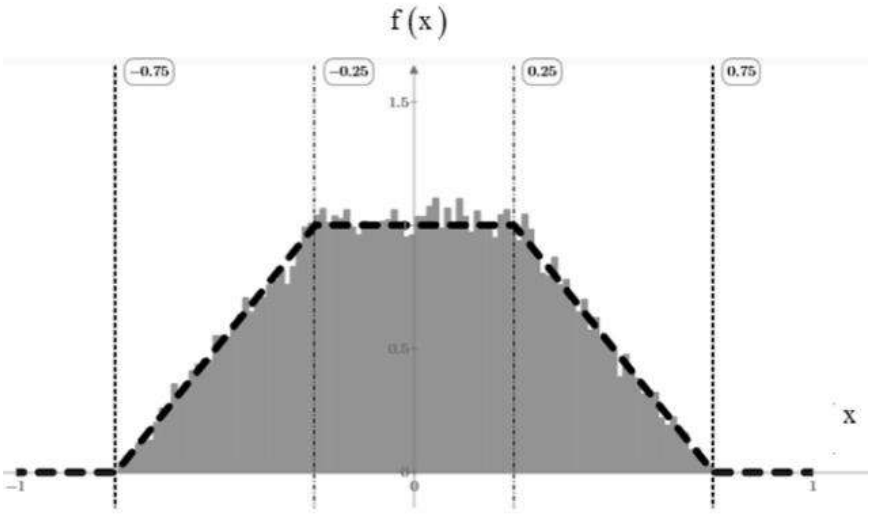


Рис. 4. Результат статистических испытаний датчика случайных координат

Перейдем к дискретному виду равномерно распределенных чисел, используя операцию целочисленного округления и центрирования:

$$z' = round(z) - 0.5, \text{ где } z \square unif[0,1]. \tag{3}$$

Тогда, ограничившись числом слагаемых, равным $m_N = 30$, случайное число минуций на портрете можно определить, как сумму

$$N = \sum_{i=1}^{30} z' + 30. \tag{4}$$

Дискретная случайная величина N может принимать целые значения из диапазона $[15,45]$. Усеченная нормальная функция ПРВ этой случайной величины аппроксимируется взвешенными биномиальными коэффициентами:

$$Q(N_i) = \binom{i}{30} \cdot \left(\frac{1}{2}\right)^{30}, \quad i \in [0,30], \; N_i \in [15,45], \tag{5}$$

где $Q(N_i)$ – вероятность того, что число минуций на портрете (с учетом точек, замаскированных за пределами единичного квадрата) будет составлять величину N_i.

Вид и числовые характеристики распределения (5) показаны на рис. 5.

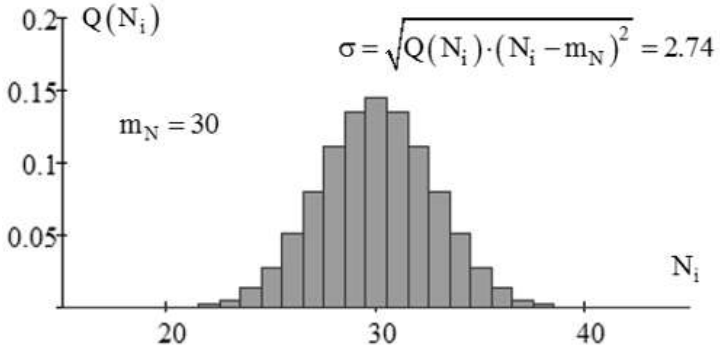


Рис. 5. ПРВ числа характерных точек на нормированном портрете отпечатка

Для моделирования нормированных в единичном квадрате случайных значений углов прихода в характерные точки целесообразно использовать равномерно распределенную на единичном отрезке случайную величину $z \square unif[0,1]$:

$$\varphi = z. \qquad (6)$$

№	X	Y	φ
1	0.21	-0.07	0.6
2	0.28	0.18	0.58
3	0.12	-0	0.49
4	0.19	0.31	0.74
5	**0.07**	**-0.68**	**0.62**
6	0.05	-0.12	0.8
7	-0.25	0.33	0.58
8	-0.01	0.42	0.91
9	0.17	0.15	0.73
10	0.12	0.23	0.67
11	**0.3**	**0.54**	**0.32**
12	**0.64**	**-0.14**	**0.31**
13	0.13	0.44	0.11
14	0.09	-0.05	0.85
15	0.01	0.39	0.15
16	**-0.05**	**-0.55**	**0.08**
17	**0.51**	**0.5**	**0.64**
18	-0.25	-0.34	0.55
19	0.23	-0.41	0.41
20	0.13	-0.41	0.47
21	0.14	0.08	0.15
22	0.06	0.23	0.74
23	-0.05	0.17	0.83
24	**0.69**	**0.31**	**0.87**
25	**0.1**	**0.7**	**0.3**
26	-0.33	-0.3	0.13
27	-0.17	-0.11	0.78
28	0.15	0.08	0.61

Истинный угол прихода определяется на основе нормированного значения из (6): $\Phi = 2\pi \cdot \varphi$. В таблице представлены результаты моделирования нормированного портрета на основании распределений (1), (4) и (6). Заштрихованные строки в таблице соответствуют точкам, не попавшим в единичный квадрат. Поэтому, несмотря на то, что в эксперименте получено $N = 28$, в единичном квадрате (рис. 6) оказалась только 21 точка. «Замаскированные» точки могут проявиться, если при возникновении искажений портрета возникнут смещения по осям X и Y или соответствующие повороты изображения.

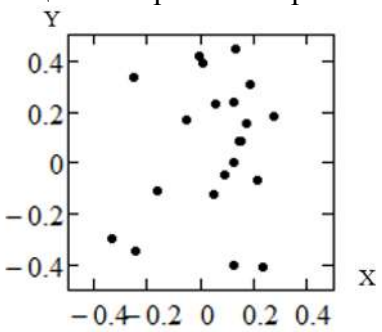


Рис. 6. Пример реализации случайного распределения характерных точек

При необходимости в соответствующем алгоритме обработки можно рассматривать трехмерное пространство размещения точек, если добавить третью координату для нормированного угла φ.

2. Математическая модель искажений распределения характерных точек

Анализ возможных вариаций сканирования отпечатков на основе существующих баз данных позволяет выявить следующие основные факторы, приводящие к появлению отличий в различных реализациях портретов одного и того же объекта: смещения геометрического центра, вызванные изменением положения объекта в поле сканирования; вращения изображений, возникающие по тем же причинам; «стирания» или появление «ложных» точек из-за неправильных настроек алгоритма сканера или попадания в поле сканирования посторонних объектов; дрейф взаимного расположения точек из-за ошибок алгоритма распознавания.

2.1. Ошибки смещения геометрического центра

Данные ошибки вызваны неточностью расположения объекта сканирования относительно центра поля сканируемого окна. Для описания распределения таких ошибок целесообразно использовать унимодальную трапецеидальную центрированную ПРВ вида (1) с измененным (компрессированным) по горизонтали масштабом. Это является следствием предположения о сравнительно небольших отклонениях объекта сканирования в нормированном квадрате окна.

$$\left.\begin{matrix}\Phi(d_x)\\ \Phi(d_y)\end{matrix}\right\} = \begin{cases} 12.5\cdot d_x + 3.75 & if \ \ -0.3 \le d_x < -0.1; \\ 2.5 & if \ \ -0.1 \le d_x \le 0.1; \\ -12.5 d_x + 3.75 & if \ \ 0.1 < d_x \le 0.3; \\ 0 & if \ \ |d_x| > 0.3. \end{cases} \quad (7)$$

Вид ПРВ (7) представлен на рис. 7. Для получения случайной величины, распределенной по данному закону, вновь воспользуемся методом обратных функций [6].

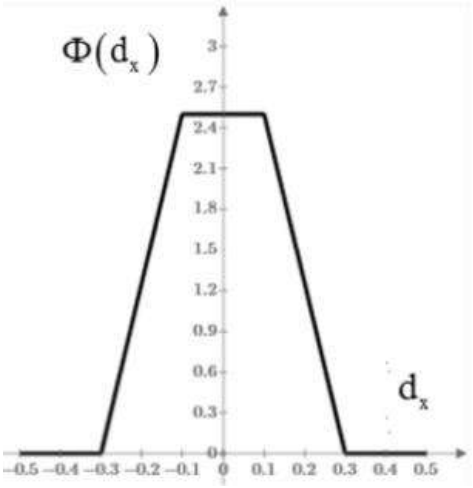


Рис. 7. ПРВ ошибок координатного смещения

Если $z \sim unif[0,1]$, то для получения случайной величины d_x с распределением (7) необходимо использовать преобразование:

$$d_x(z) = \begin{cases} \sqrt{0.16\cdot z} - 0.3 & if \ \ 0 \le z < 0.25; \\ 0.4\cdot z - 0.2 & if \ \ 0.25 \le z \le 0.75; \\ -\sqrt{0.16\cdot(1-z)} + 0.3 & if \ \ 0.75 < z \le 1. \end{cases} \quad (8)$$

Справедливо полагать, что ошибки смещения, обладающие функцией ПРВ (8), действуют по координатам X, Y единичного квадрата портрета отпечатка независимо.

2.2. Ошибки из-за «стирания» и добавления «ложных» точек

Распределение ошибок стирания может параметризовано величиной p_E – вероятностью стирания (Erasure) одной, отдельно взятой характерной точки на портрете отпечатка. В предположении независимости ошибок стирания, их распределение описывается обычным дискретным биномиальным распределением:

$$P_E(k) = \sum_{i=0}^{k} \binom{i}{N} \cdot {p_E}^i \cdot (1-p_E)^{N-i}, \quad (9)$$

где $P_E(k)$ – вероятность стирания не более, чем k точек на портрете; N – число выявленных точек, определяемое распределением (5). Величина параметра p_E для различных способов обработки отпечатков может располагаться в пределах $0 < p_E \le 0.1$. Для моделирования процесса стирания после получения модели портрета (таблица), с использованием датчика равномерно распределенных в единичном интервале чисел производится генерация вектора $Z = \{z_1, z_2, \cdots, z_N\}$, элементы которого $z_i \sim unif[0,1],\ i = 1 \ldots N$. На основе вектора Z вычисляется вектор стираний $E = \{e_1, e_2, \cdots, e_N\}$, элементы которого имеют бинарное значение и получаются функциональным преобразованием координат вектора Z:

$$e_i = \left\lfloor \frac{z_i}{1-p_E} \right\rfloor, \quad e_i = [0,1],\ i = 1 \ldots N. \quad (10)$$

Далее строки в таблице реализации портрета, обладающие номерами, которые соответствуют порядковым номерам единичных элементов вектора E, удаляются из таблицы и, соответственно, из портрета минуций.

Вероятностное описание ошибок, связанных с появлением ложных (несуществующих в действительности) точек может быть сделано на основе распределения Пуассона:

$$\Pr(K) = \frac{\lambda_A{}^K}{K!}\exp\{\lambda_A\}, \tag{11}$$

где $\Pr(K)$ – вероятность появления K ложных точек на реализации портрета; λ_A – (Appearance) эмпирически определенное математическое ожидание числа ложных точек на одном портрете.

Аппроксимация случайной величины – количества ложных точек на портрете, подчиненной распределению Пуассона (11), достигается с использованием датчика $unif[0,1]$ и обычного биномиального распределения следующим образом. На основании статистической обработки достаточного количества реализаций портретов эмпирически определяется математическое ожидание количества ложных точек на одном портрете λ_A. Как правило, $0.1 \le \lambda_A \le 0.5$ (при этом добавления сохраняют свойство быть менее вероятными, чем стирания). Генерируется вектор $Z = \{z_1, z_2, \cdots, z_M\}$ с равномерно распределенными в единичном интервале элементами $z_i \sim unif[0,1]$, $i = 1 \ldots M$. Затем, на основании преобразования, аналогичного (10), получается вектор добавления $A = \{a_1, a_2, \cdots, a_M\}$ с бинарными элементами, получаемыми по правилу

$$a_i = \left\lfloor \frac{z_i}{1 - \frac{\lambda_A}{M}} \right\rfloor, \quad a_i = [0,1], \; i = 1 \ldots M. \tag{12}$$

Количество единичных элементов в векторе A подчинено биномиальному закону (9) с параметром $P_E = \frac{\lambda_A}{M}$. Тогда количество добавляемых ложных точек на портрете определяется квадратом длины вектора A:

$$K = |A|^2 = \sum_{i=1}^{M} a_i. \tag{13}$$

Аппроксимация распределения Пуассона (11) будет тем более точной, чем больше выбранное значение M. Для приемлемой аппроксимации распределения Пуассона при $\lambda_A \sim 1$ достаточно потребовать выполнения неравенства

$$M \ge \lambda^{-1}. \tag{14}$$

Моделирование появления ложных точек производится на основании полученной в результате вычислительного эксперимента величины K, определяемой (13): в таблицу добавляется K строк (при $K = 0$ строки не добавляются). Генерация значений X, Y и φ для дописываемых строк производится так же, как для существующих в таблице точек – с использованием $z \sim unif[0,1]$ и функциональных преобразований (2) и (6). При этом не исключается случай, когда дополнительные точки окажутся за пределами единичного квадрата и окажутся замаскированными на исходном портрете.

2.3. Ошибки вращения изображений

Для моделирования ошибок вращения примем следующие соглашения для декартовой системы координат нормированного единичного квадрата портрета отпечатка. Вращение изображения портрета отпечатка вокруг геометрического центра единичного квадрата с координатами $[0,0]$ удобно моделировать поворотом осей координат на плоскости на заданный угол α (рис. 8) .

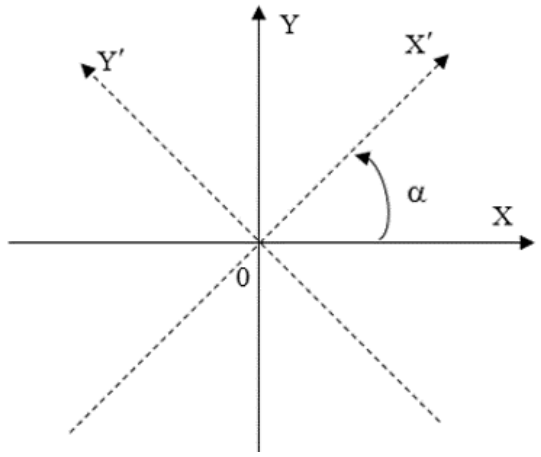


Рис. 8. Иллюстрация вращения координат

Нулевому значению угла поворота соответствует ось $0X$, положительным направлением угла является движение против часовой стрелки. При этом следует иметь в виду, что поворот осей на угол α эквивалентен повороту исходного изображения (в координатах $X0Y$) в противоположную сторону на угол $-\alpha$ (в координатах $X'0Y'$). Известно, что связь между координатами произвольной точки $\begin{pmatrix} X \\ Y \end{pmatrix}$ в исходной системе координат $X0Y$ и координатами новой точки $\begin{pmatrix} X' \\ Y' \end{pmatrix}$ в развернутой на угол α системе $X'0Y'$ задается в матричной форме выражением

$$\begin{pmatrix} X' \\ Y' \end{pmatrix} = U(\alpha) \cdot \begin{pmatrix} X \\ Y \end{pmatrix}, \text{ где } U(\alpha) = \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix}. \quad (15)$$

Использование преобразования (15) позволяет определить алгоритм моделирования поворота изображения. Пусть G – матрица портрета (см. таблицу выше) размером $(N \times 3)$. Разобьем ее на подматрицы $G = (XY \| \varphi)$, где XY – подматрица размером $(N \times 2)$, содержащая N строк с парой координат X и Y для каждой из N точек исходного портрета; φ – вектор столбец, содержащий нормированные в интервале $[0,1]$ значения углов прихода для соответствующих N точек.

Поворот расположения точек портрета на угол α (заданный в радианах) достигается преобразованием подматрицы XY:

$$XY' = U(-\alpha) \cdot (XY)^T, \quad (16)$$

где $(XY)^T$ – транспонированная подматрица XY.

Изменение вектора φ, связанное с поворотом портрета, определяется формулой

$$\varphi' = \left(\varphi - \frac{\alpha}{2 \cdot \pi}\right) \bmod 1 - \left\lfloor \left(\varphi - \frac{\alpha}{2 \cdot \pi}\right) \bmod 1 \right\rfloor, \quad (17)$$

где операция $(\text{"arg"}) \bmod 1$ извлекает дробную часть из "arg" с учетом знака.

Матрица повернутого на угол α портрета определяется объединением полученных подматрицы и вектора:

$$G' = \left((XY')^T \| \varphi' \right). \quad (18)$$

Ошибки вращения изображений задаются распределением случайного угла поворота α. Эмпирические соображения позволяют ограничить диапазон возможных значений в пределах прямого угла

$$\alpha \in \left[-\frac{\pi}{4}, +\frac{\pi}{4} \right], \quad (19)$$

а к функции ПРВ предъявить требования унимодальности и центрированности. Для

моделирования случайной величины α используем метод, основанный на центральной предельной теореме и позволяющий аппроксимировать усеченный нормальный закон суммированием ограниченного числа равномерно распределенных в единичном диапазоне центрированных случайных чисел. Используем аппроксимацию нормального закона суммированием четырех преобразованным линейным способом случайных величин $z_i \square unif[0,1],\ i \in 1,\ldots,4$. Ограничению (19) соответствуют нормированные на 2π значения $\alpha_H \in \left[-\frac{1}{8}, +\frac{1}{8}\right]$, тогда реализация случайного нормированного угла поворота получается функциональным преобразованием z_i вида

$$\alpha_H = \sum_{i=1}^{4}\left(\frac{z_i - 0.5}{16}\right). \tag{20}$$

Аппроксимируемая нормальная ПРВ имеет вид

$$f(\alpha_H) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{\alpha_H{}^2}{2\sigma^2}\right), \quad \sigma = \sqrt{\frac{1}{3\cdot 256}}. \tag{21}$$

На рис. 9 представлен вид функции (21) (штриховая линия) и гистограмма $H(\alpha_H)$ распределения вероятностей аппроксимации (20), полученная при числе испытаний, равном 10^5.

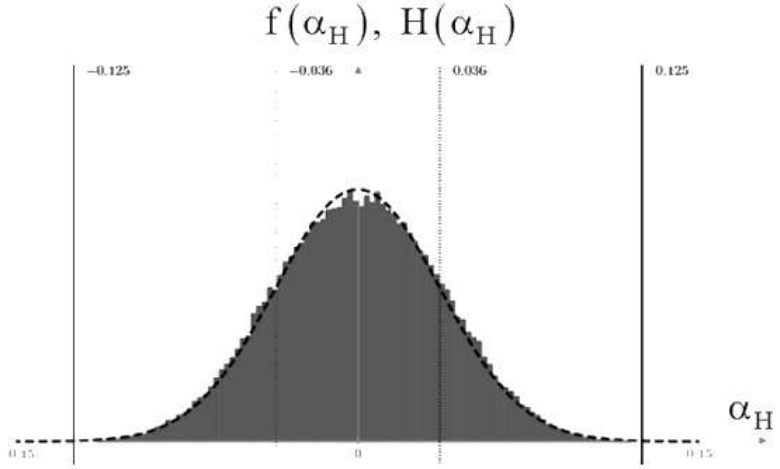


Рис. 9. ПРВ и гистограмма аппроксимации ПРВ
нормированного случайного угла поворота

Как видно, использование только четырех слагаемых в сумме выражения (20) обеспечивает хорошее приближение усеченной нормальной ПРВ. Абсолютное значение случайного угла поворота, моделируемого преобразованием (20), не выходит за оговоренные выше пределы $\pm 45^o$, а среднеквадратичное значение составляет

$$\sigma = \sqrt{\frac{1}{3\cdot 256}}\cdot 360^o = 0.036\cdot 360^o \approx 13^o.$$

Таким образом, можно констатировать, что проведенные обоснования и полученные в разделах 1 и 2 математические преобразования представляют собой законченную функциональную модель для имитации получения реализаций дактилоскопических портретов в условиях реальных искажений.

Выводы

Основной результат статьи заключается в получении инструментария для вероятностного моделирования реализаций отпечатков пальцев в условиях неточного сканирования и помеховых воздействий. При этом, на основе анализа реальных реализаций, предложены адекватные модели генерации случайных параметров дактилоскопических портретов, а также модели возможных искажений и ошибок распознавания биометрических образцов. Для обоснования видов и числовых характеристик использованных функций плотности распределения вероятностей применен эвристический анализ дактилоскопических данных и результатов их обработки существующими открытыми программными средствами. Универсальность разработанному комплексу моделей придает использование (в качестве первичного источника энтропии) датчика случайных вещественных чисел, равномерно распределенных в диапазоне от нуля до единицы. Подобные датчики встроены, практически, в библиотеки всех известных систем программирования.

Дальнейшее использование полученных результатов для статистического моделирования процессов первичной биометрической аутентификации персоналий дает возможности разработки методов оптимальной предобработки результатов сканирования для криптографических задач, связанных с нечеткой экстракцией данных, и экспериментальной проверки их эффективности.

Список литературы:

1. Proposal for a regulation of the European Parliament and of the council on electronic identification and trust services for electronic transactions in the internal market, Brussels. [Электронный ресурс]. http://ec.europa.eu/information_society/policy/esignature/eu_legislation/revision .

2. ISO/IEC 19794-1 – 5. Information technology – Biometric data interchange formats. Part 1 – 5.

3. Daugman J. Information Theory and the IrisCode // IEEE transactions on information forensics and security. 2016. Vol. 11, No. 2. Pp. 400-409.

4. SourceAFIS for Java and .NET [Электронный ресурс]. https://sourceafis.machinezoo.com/ .

5. Takahashi1 K. Signature Schemes with a Fuzzy Private Key / Kenta Takahashi1, Takahiro Matsuda, Takao Murakami, Goichiro Hanaoka, Masakatsu Nishigaki // Proceedings of ACNS, 2017. Pp. 1-51.

6. Вентцель Е. С., Овчаров Л. А. Теория вероятностей. Москва : Наука, 1969. 366 с.

Харьковский национальный
университет имени В.Н. Каразина

Поступила в редколлегию 05.04.2018

МЕТОДЫ ПОСТРОЕНИЯ ЗАЩИЩЕННЫХ ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

УДК 681.3.07 (3.06)

О.П. НАРЄЖНІЙ, канд. техн. наук, В.В. СЕМЕНЕЦЬ, д-р техн. наук,
Т.О. ГРІНЕНКО, канд. техн. наук

МЕТОД ВИМІРЮВАННЯ КВАНТОВОГО ФАЗОВОГО ШУМУ ТА ШИРИНИ ЛІНІЇ РОБОЧОГО ПЕРЕХОДУ РАДІООПТИЧНОЇ СИСТЕМИ ГЕНЕРАТОРА ВИПАДКОВИХ ЧИСЕЛ

Вступ

Такі важливі питання квантової інформації, як квантові шуми, квантова корекція помилок, числові параметри квантової інформації (ентропія фон Неймана, пропускна здатність квантового каналу зв'язку та інші) знаходяться в стадії інтенсивних експериментальних та теоретичних досліджень [1, 2]. Методологія і теоретична база вивчення проблем вимірювання квантових шумів, що виникають при проектуванні і створенні апаратно-програмних реалізацій квантових генераторів випадкових чисел (КГВЧ), в даний час тільки формуються.

Важливим напрямком досліджень є питання реалізації та використання методів метрологічних досліджень за допомогою сучасної еталонної бази на різних етапах генерування випадкових послідовностей та верифікації квантових генераторів. Останні дослідження дозволили припустити, що одними із найбільш переважних є методи генерування випадкових послідовностей, що ґрунтуються на фізичному джерелі випадковості з використанням елементарних квантово-механічних рішень. В загальному розумінні поява кожного окремого результату такого квантово-механічного рішення є об'єктивно випадковою (невизначуваною, невідомою). Існує ряд елементарних методів (рішень) генерування випадкових послідовностей, які можна використовувати в якості джерел випадковості. Серед основних механізмів генерування ключів слід виділити такі [2 – 20]: метод розщеплення одиничного фотона на два шляхи та поляризації одиничного фотона; метод виявлення заплутаності шляху числа фотонів; методи підрахунку часу генерації або кількості фотонів; використання гомодинного виявлення флуктуації вакуумного стану; інтерферометричні схеми; використання методу подвійного радіооптичного резонансу (ПРР) в парах лужних металів [20].

В роботі [20] розроблено систему компарування для вимірювання фази коливань та квантових фазових шумів. Експеримент з генерації випадкових чисел методом ПРР здійснювався на основі метрологічних методів експериментального дослідження квантових генераторів шумів та квантових мір частоти (КМЧ). Висока стабільність КМЧ дозволяє стабілізувати фазу КГВЧ регулюванням його опорної частоти. При цьому проста і надійна конструкція такого генератора припускає його практичне застосування.

Мета статті – обґрунтування методу вимірювання квантового фазового шуму та ширини лінії робочого переходу радіооптичної системи генератора випадкових чисел (ГВЧ), що базується на методі подвійного радіооптичного резонансу.

Основна частина

Побудова перспективного КГВЧ на основі методу ПРР вимагає з'ясування оптимальних умов процесу оптичної накачки з точки зору отримання максимального параметра якості квантових дискримінаторів (КД). Для цього розраховують параметр якості при накачуванні атомів Rb^{87} природним світлом з урахуванням релаксаційних процесів, спектрального складу і поглинання світла накачування. Інтенсивність квантового фазового шуму є однією з найважливіших метрологічних характеристик КГВЧ як генератора шуму. Відомо [20], що серед усіх можливих типів шумів у схемі оптичного накачування і детектування резонансу прин-

ципово квантову природу мають тільки два типи шумів – дробовий шум світла і квантовий шум атомного ансамблю. Метод ПРР базується на двох складових: перша – це селективне оптичне збудження, що приводить до появи збуджених атомів, які нерівномірно заселяють підрівні збудженого стану; друга – це індукування радіочастотних переходів з допомогою допоміжного змінного поля і реєстрація цих переходів у оптичному каналі (подвійний резонанс названий подвійним, тому що є два види резонансу – на оптичних частотах і на радіочастоті).

Ширину лінії КД на парах лужних металів можна розглядати як наслідок випадкового фазового коливання оптичного поля. Залежність ширини лінії КД від вихідної потужності спектрального джерела лампи або лазера потребує розроблення теоретичної моделі генерації випадкових чисел. На практиці існують фактори, які важко враховувати при проведенні вимірювань методами інтерферометричного експерименту оптичної фази поля КД. Тому використання в якості джерела квантового фазового шуму КД на парах лужних металів потребує проведення експериментальних досліджень за допомогою сучасних еталонних КМЧ. При цьому необхідною умовою для генерації випадкових чисел за допомогою КД на парах лужних металів є стабілізація частоти вимірювання.

При визначенні параметрів квантових фазових шумів вихідного сигналу КГВЧ, що реалізований за допомогою методу ПРР, можна застосовувати два методи виміру – двогенераторний і тригенераторний. В роботі [20] при настроюванні та регулюванні перспективного КГВЧ використовувався метод групового еталонування за допомогою КМЧ. При цьому поряд з удосконалюванням самих методів вимірювання виникає необхідність розробляти математичні моделі, що враховують вплив різних дестабілізуючих факторів на флуктуації фази (частоти) ГВЧ.

У загальному випадку побудовою даних моделей займається теорія флуктуацій в автоколивальних системах. Найважливішим її розділом, що виділився за останні два десятиліття в самостійний напрямок, є теорія флуктуацій частоти квантових автогенераторів, що виявляє причини, характер і статистичні характеристики випадкових відхилень опорної частоти від її сталого значення [21]. При цьому облік впливу дестабілізуючих факторів пов'язаний з відомими труднощами, обумовленими їх різноманітністю не тільки по виду, але й по інтенсивності, напрямку дії, спектру тощо. Так, облік магнітного та електромагнітного впливів має важливе значення для пасивних КМЧ. Це обумовлено тим, що принцип дії квантових дискримінаторів побудований на розщепленні в магнітному полі надтонких рівнів квантових переходів з підбором такого значення магнітного поля, при якому частота генерованих кварцевим генератором коливань стає кратною номінальному значенню частоти. Крім стохастичної зміни магнітного та електромагнітного полів, можлива й квазирегулярна їх зміна, яка може впливати на стабільність частоти та фази вихідного сигналу КГВЧ. Зокрема, така ситуація має місце при об'єднанні пасивних КМЧ у груповий еталон часу та частоти. По суті, взаємний вплив КМЧ у групі еквівалентний тому, що кожна міра зазнає деякий зовнішній вплив з боку інших мір групи. І, оскільки такий вплив взаємний, то його результатом повинен бути якийсь векторний процес із взаємною кореляцією між його елементами. Це призводить до необхідності пошуку нових методів стабілізації частоти КГВЧ на основі використання інформації про залежність поведінки частоти вихідного сигналу з урахуванням похибки від взаємодії.

Проте, у теорії флуктуацій частоти квантових автогенераторів не розглядалися питання, пов'язані з вивченням їх взаємодії (взаємозв'язку), тобто опускалися питання впливу похибки від взаємодії на інтенсивність флуктуацій частоти. Це пов'язано з тим, що апріорі передбачається можливість компенсації даної похибки апаратними засобами. Тому усі відомі методи оцінки інтенсивності частотних (фазових) флуктуацій і експериментальні методи виміру нестабільності частоти: фазовий, інтерференційний, лічильно-імпульсний та ін., засновані на базовому припущенні про відсутність взаємних впливів КМЧ у процесі вимірювань.

Крім того, на основі результатів, отриманих у роботах [22, 23], можна стверджувати, що задача ідентифікації КГВЧ за результатами вимірювань належить до класу некоректних задач математичної фізики й вимагає рішення спеціальних питань – пошуку й визначення регуляризовувальних параметрів або факторів, що дозволяють одержувати стійкі рішення диференціальних рівнянь, які описують поведінку вихідних сигналів даних мір.

Звідси випливає така постановка задачі: розробити методику регуляризації задачі ідентифікації групи КМЧ та КГВЧ за наявності похибки від взаємодії через канали звірення з обліком їх адитивних внутрішніх шумів на основі застосування статистичних методів рішення двоточечних крайових завдань; обґрунтувати перетворення вектора стану групи КМЧ як неспостережуваного процесу до спостережуваного на основі стохастичної моделі системи пов'язаних осциляторів.

Оскільки для одержання оцінок поточних значень фаз вихідних сигналів КМЧ доводиться мати справу з диференціальними рівняннями, інтегрування яких класичними методами неможливо (тобто є некоректні задачі математичної фізики, що вимагають процедур регуляризації), найбільший інтерес у цьому випадку представляють статистичні методи рішення двоточечних крайових завдань. Одним з таких методів є метод фільтра Калмана, що здійснює статистичне згладжування рішень і, за певних вимог до параметрів розглянутої моделі, є асимптотично стійким. Проте, поряд з високою точністю, даний метод використовує й найбільшу кількість апріорної інформації: коваріацію помилок і математичного очікування правої частини й рішення [23]. Для отримання даної апріорної інформації будемо розглядати груповий еталон з позицій теорії нелінійних автоколивальних систем із близькими частотами. При цьому необхідно побудувати модель зміни фази (частоти) групового еталону як стохастичну модель системи пов'язаних осциляторів (мір частоти).

У роботі [24] показано, що зміна фази (повільна у порівнянні з періодом коливань) вихідного сигналу кожної міри в групі з N мір за наявності їх взаємодії може бути представлена диференціальним рівнянням

$$\dot{\psi}_i(t) = \Delta\omega_i + \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\frac{\alpha_{ij}}{2} \frac{A_j}{A_i} \cos\phi_{ij}(t) \right] + \xi_i(t), \qquad (1)$$

де A_j та A_i – амплітуди коливань вихідних сигналів *j*-ї та *i*-ї міри відповідно; α_{ij} – коефіцієнт електричного зв'язку між мірами та КГВЧ; $\phi_{ij}(t) = \psi_i(t) - \psi_j(t)$ – різниця фаз коливань вихідних сигналів, генерованих *i*-ю та *j*-ю мірами та КГВЧ; $\xi_i(t)$ – власні флуктуації частоти вихідного сигналу *i*-ї міри з математичним очікуванням $M[\xi_i(t)] = 0$ і кореляційною функцією $M[\xi_i(t)\xi_i(t+\tau)] = \sigma_i^2\delta(\tau)$ ($\delta(\tau)$ – дельта-функція, а σ_i – середнє квадратичне відхилення (СКВ) флуктуацій вихідного сигналу *i*-ї КМЧ); $\Delta\omega_i$ – відхилення частоти *i*-ї міри від номінального значення, кількісна оцінка якого може бути здійснена шляхом звірення вихідного сигналу даної міри з еталонними сигналами часу й частоти, переданими спеціальними системами.

У процесі взаємних звірень КМЧ вимірам піддаються різниці фаз $\phi_{ij}(t)$ або різниці частот $\dot{\phi}_{ij}(t)$ залежно від типу використовуваних компараторів. Вимірювання процесу $\psi_i(t)$ в групі, що складається із КМЧ однакової точності (як правило, найвищої), неможливе, тобто процес $\psi_i(t)$ є принципово неспостережуваним. Тому рішенням задачі ідентифікації групового еталону з максимально можливою точністю є розробка методик ідентифікації поточного стану процесу $\psi_i(t)$ за результатами вимірювань процесів $\phi_{ij}(t)$ або $\dot{\phi}_{ij}(t)$ з наступною компенсацією взаємного впливу КМЧ та КГВЧ. Дотримуючись мети даного завдання, систему рівнянь (1) щодо вимірюваних параметрів можна перетворити таким чином:

$$\dot{\phi}_{ij}(t) = \Delta_{ij} + \sum_{\substack{n=1 \\ n\neq i}}^{N} A_{in}\cos\phi_{in}(t) - \sum_{\substack{m=1 \\ m\neq j}}^{N} A_{jm}\cos\phi_{jm}(t) + (\xi_i - \xi_j), \quad (2)$$

де $\Delta_{ij} = (\Delta\omega_i - \Delta\omega_j)$ – різниця частот між *i*-ю та *j*-ю КМЧ та КГВЧ; $A_{in} = \frac{\alpha_{in}}{2}\frac{A_n}{A_i}$ – узагальнене значення амплітуди, обумовлене похибкою від взаємодії *i*-ї та *n*-ї мір; $A_{jm} = \frac{\alpha_{jm}}{2}\frac{A_m}{A_j}$ – аналогічно для *j*-ї та *m*-ї мір.

Залежно від співвідношення параметрів Δ_{ij}, $A_{ij} = \frac{\alpha_{ij}}{2}\frac{A_j}{A_i}$ і $\sigma_{ij} = \sqrt{\sigma_i^2 + \sigma_j^2}$ система рівнянь (2) допускає різні типи рішень. Якщо $\Delta_{ij} \le \sum_{\substack{n=1 \\ n\neq i}}^{N} |\alpha_{in}| + \sum_{\substack{m=1 \\ m\neq j}}^{N} |\alpha_{jm}|$, система рівнянь стає виродженою щодо членів з індексами *ij* та *ji*. Фізично це виявляється у тому, що відбувається взаємна синхронізація КМЧ із номерами *i* та *j*, у результаті чого вони поводяться як одне ціле. А за виконання умови $\Delta_{ij} > \sum_{\substack{n=1 \\ n\neq i}}^{N} |\alpha_{in}| + \sum_{\substack{m=1 \\ m\neq j}}^{N} |\alpha_{jm}|$ між усіма КМЧ виникає режим амплітудних биттів. При цьому задача ідентифікації групового еталону (оцінки фази КМЧ та КГВЧ) не є коректною в класичному сенсі (за Адамаром) [25, 26]. Проте, у роботі [24] показано, що для випадку знаходження усіх КМЧ групи та КГВЧ в режимі биттів, методами розщеплення вдається зробити редукцію даної задачі до суми більш простих, умовно коректних задач математичної фізики.

Для отримання апріорної інформації про тип регулярної складової рішення в режимі биттів КМЧ та КГВЧ скористаємося методом послідовних наближень (метод Крилова – Боголюбова) [27]. Припускаючи, що власні флуктуації частоти усіх КМЧ дорівнюють нулю, послідовне наближення рішення рівняння (2) і відповідно (1) буде відбуватися таким чином.

Початкове наближення обираємо у вигляді: $\tilde{\phi}_{ij,0}(t) = \Delta_{ij}t$, де $\tilde{\phi}_{ij,0}(t)$ – початкове наближення регулярної складової різниці фаз коливань $\phi_{ij}(t)$.

Тоді перше наближення рішення системи (1) щодо регулярної складової зміни фази вихідного сигналу *i*-ї міри

$$\tilde{\psi}_{i,1}(t) = \Delta\omega_i t + \sum_{\substack{j=1 \\ j\neq i}}^{N} A_{ij}\sin(\Delta_{ij}t + \varphi_{0i}), \quad (3)$$

і, відповідно, перше наближення рішення системи (2) щодо регулярної складової різниці фаз коливань між *i*-ю та *j*-ю мірами:

$$\tilde{\phi}_{ij,1}(t) = \Delta_{ij}t + \sum_{\substack{n=1 \\ n\neq i}}^{N} A_{in}\sin(\Delta_{in}t + \vartheta_{0in}) - \sum_{\substack{m=1 \\ m\neq j}}^{N} A_{jm}\sin(\Delta_{jm}t + \vartheta_{0jm}), \quad (4)$$

де φ_{0i} – початкова фаза *i*-ї КМЧ; ϑ_{0in} – різниця початкових фаз між *i*-ю та *n*-ю мірами.

Друге наближення рішення рівняння (2):

$$\tilde{\psi}_{i,2}(t) = \Delta\omega_i t + \sum_{\substack{j=1 \\ j\neq i}}^{N} A_{ij} \sin\left(\Delta_{ij} t + \varphi_{0i}\right) +$$

$$+ \sum_{\substack{j=1 \\ j\neq i}}^{N} A_{ij} \left\{ \sum_{\substack{n=1 \\ n\neq i}}^{N} A_{in} \left[\frac{\cos\left(\Delta_{jn} t + \vartheta_{0in}\right)}{\Delta_{jn}} - \frac{\cos\left(\Delta_{ij} t + \Delta_{in} t + \vartheta_{0in}\right)}{\Delta_{ij} + \Delta_{in}} \right] + \right.$$

$$\left. + \sum_{\substack{m=1 \\ m\neq j}}^{N} A_{jm} \left[\frac{\cos\left(\Delta_{im} t + \vartheta_{0\,jm}\right)}{\Delta_{im}} - \frac{\cos\left(\Delta_{ij} t + \Delta_{jm} t + \vartheta_{0\,jm}\right)}{\Delta_{ij} + \Delta_{jm}} \right] \right\}. \qquad (5)$$

Аналогічно записуються наступні ітерації рішення рівнянь (2) і (3). З (5) виходить, що кожне наступне наближення породжує появу майже періодичних складових рішення на нових комбінаційних частотах. Тому в спектрі випадкового процесу, утвореного з безперервного ряду вимірювань різниць частот (фаз) звірюваних мір, присутні "яскраві" спектральні лінії, породжені взаємним впливом вихідних сигналів на частотах аналізу, приблизно рівних різницям частот $\Delta_{ij} \approx 10^{-p}$ рад/с, де $p \geq 4$ для КМЧ. Спектральна густина потужності фазових флуктуацій на цих частотах буде дорівнювати величині A_{in}^2. Вплив даних квазигармонійних складових на загальну поведінку фаз вихідних сигналів КМЧ залежить від співвідношення Δ_{ij} й A_{in}. В [28] запропоновано спосіб ідентифікації необхідного типу рішення (відповідного режиму взаємодії КМЧ – режиму биттів) за експериментальними даними в спектральній області.

При цьому використання рівняння (1) як рівняння стану в методі фільтра Калмана недоцільно, тому що вимірюванням з необхідною точністю можуть бути піддані тільки різниці фаз $\phi_{ij}(t)$ вихідних сигналів КМЧ. Як показано в [29], матриця вимірювань, що описує процес взаємних звірень мір частоти й часу за допомогою компараторів, така, що умова спостереження не виконується (ранг матриці вимірювань на одиницю менше кількості КМЧ, задіяних у групі). Тому використання стандартних методик лінійної алгебри у фільтрі Калмана призведе до нестійкості одержуваних з його допомогою рішень. Спостережуваним є тільки рівняння стану (2).

Методики визначення кількісних значень Δ_{ij}, A_{ij} і початкових значень різниці фаз $\phi_{ij}(0)$ наведено в [29]. Рівняння (1) і (2) при переході до кінцевих різниць перетворяться відповідно до вигляду

$$\psi_i(k+1) = \psi_i(k) + \Delta\omega_i \tau + \sum_{\substack{j=1 \\ j\neq i}}^{N} \tau A_{ij} \cos\phi_{ij}(k) + \bar{\xi}_i(k); \qquad (6)$$

$$\phi_{ij}(k+1) = \phi_{ij}(k) + \tau\Delta_{ij} + \sum_{\substack{n=1 \\ n\neq i}}^{N} \tau A_{in} \cos\phi_{in}(k) - \sum_{\substack{m=1 \\ m\neq j}}^{N} \tau A_{jm} \cos\phi_{jm}(k) + \bar{\zeta}_{ij}(k), \qquad (7)$$

де τ – інтервал вимірювання різниць фаз $\phi_{ij}(k\tau)$ вихідних сигналів КМЧ та КГВЧ компаратора-

ми $\bar{\zeta}_{ij}(k) = \int_{k\tau}^{(k+1)\tau} \zeta_{ij}(t)dt$; $\bar{\xi}_i(k) = \int_{k\tau}^{(k+1)\tau} \xi_i(t)dt$ – середні значення відповідних шумів на інтерва-

лі вимірювання τ, інтеграли від випадкового процесу розуміються в сенсі, наведеному у роботі [23].

Відповідно до теореми Байеса, умовне середнє значення $\psi_i(k+1|k)$ i-ї КМЧ на $k+1$ кроці ітерації однозначно може бути визначене через умовне середнє значення $\phi_{ij}(k|k)$, оцінене за допомогою методу фільтра Калмана за результатами вимірювань різниці фаз $\phi_{ij}(k)$ на k-му інтервалі вимірювань, і через умовне середнє значення $\psi_i(k|k)$, оцінене на k-му кроці ітерації з виразу

$$\psi_i(k+1|k)=\psi_i(k|k)+\Delta\omega_i\tau+\sum_{\substack{j=1\\ j\neq i}}^{N}\tau A_{ij}\cos\phi_{ij}(k|k). \qquad (8)$$

При цьому, відповідно до теореми Байеса дана оцінка буде оптимальною. Для отримання оцінок поточного значення різниці фаз $\phi_{ij}(k)$ застосуємо метод фільтра Калмана як наслідок спостереження системи, рівняння стану якої описується рівнянням (7), а рівняння вимірювання – вектором виду

$$\vec{Y}(k)=\vec{\Phi}(k)+\vec{\eta}(k),$$

де $\vec{Y}(k)=\left[y_{ij}(k)\right]$ – вектор результатів вимірювань вектора різниці фаз $\vec{\Phi}(k)=\left[\phi_{ij}(k)\right]$; $\vec{\eta}(k)=\left[\overline{\eta}_i(k)\right]$ – вектор власної флуктуації фаз вимірювачів (компараторів) з математичним очікуванням $M\left[\vec{\eta}(k)\right]=0$ і кореляційною функцією $M\left[\vec{\eta}(k)\vec{\eta}(k+1)^T\right]=R\,\delta(\tau)$.

Тоді можна стверджувати, що з точністю до векторної константи $\vec{\Psi}(0)=\left[\psi_i(0)\right]$ можна одержати оптимальні оцінки поточних значень фаз вихідних сигналів усіх КМЧ та КГВЧ, що входять до складу групи.

Проведемо аналіз можливих підходів до регуляризацій процедури оцінки поточних значень різниць фаз між мірами за допомогою нелінійного фільтру Калмана. Так, рівняння (7), що визначає поведінку випадкового процесу різниці фаз у часі, відносить некоректну за Адамаром задачу оцінки $\phi_{ij}(k)$ до класу рішень рівнянь нелінійної фільтрації. Алгоритми нелінійної фільтрації базуються на одному з двох основних регуляризовувальних підходів: на локальній або на інтегральній апроксимації.

Клас наближених алгоритмів фільтрації на основі локальної апроксимації "точного" рішення нелінійного рівняння стану дозволяє одержувати оцінюване значення шуканої величини лише в малій області її варіації.

На відміну від локальної апроксимації, застосовної при малих похибках фільтрації, основна мета інтегральної (глобальної) апроксимації полягає в одержанні наближеного рішення в усій області можливих значень параметра, що фільтрується (шуканого процесу). Це особливо важливо при малих співвідношеннях сигнал/шум, а також у задачах, пов'язаних з виходом процесу за межі заданої області (СКВ випадкового процесу щодо свого умовного середнього).

Локальна апроксимація більш проста в реалізації й використовується у тих випадках, коли очікуване рішення (умовне середнє) є гладкою, повільно мінливою в часі функцією в порівнянні з кроком квантування розглянутого дискретного випадкового процесу. У тих випадках, коли очікуване рішення є функцією, що швидко змінюється, або має розриви першого роду, кращим є використання глобальної апроксимації.

З урахуванням цього звернемось до результатів моделювання процесів зміни фаз вихідних сигналів мір частоти, обумовлених їх взаємним впливом, проведеним в [30]. При аналізі видів рішень, що допускаються рівнянням (1) для взаємодії пари КМЧ і відповідно (7), що є його аналогом для випадку парної взаємодії групи мір, було показано, що дане рівняння

допускає різні типи рішень, умовне середнє, яке може бути як гладкою, повільно мінливою функцією часу, так і функцією, що має квазістрибкоподібну (у порівнянні з інтервалом дискретизації процесу) зміну в часі залежно від співвідношення параметрів.

В [30] наведено такі можливі режими взаємодії: 1 – режим твердої синхронізації; 2 – режим синхронізму з наявністю квазістрибкоподібних змін різниці фаз, що виникають у випадкові моменти часу; 3 – режим биттів; 4 – випадковий процес із лінійним дрейфом різниці фаз. Показано, що першому, третьому й четвертому режимам властиве гладке поводження в часі умовного середнього, тобто при роботі групового еталону в одному з цих режимів доцільним є застосування локальної апроксимації. За наявності хоча б однієї пари мір в групі, що працює в другому режимі, необхідне застосування інтегральної апроксимації.

Розглянемо процедуру лінеаризації нелінійного фільтра Калмана для оцінки поточних значень різниць фаз між мірами на основі стохастичної моделі системи пов'язаних осциляторів. З рівняння (7) виходить, що поточне значення вектора стану $\vec{\Phi}(k)$ формується під впливом двох процесів, породжених однією причиною – взаємодією мір між собою. Перший векторний процес, обумовлений наявністю різниці частот між взаємодіючими мірами та КГВЧ й ефектом перетворення (амплітудно-фазової конверсії) амплітудних биттів у частотні биття, визначає зміну умовного середнього. Другий векторний процес є чисто стохастичним і описує взаємну кореляцію адитивних шумів взаємодіючих мір. Передбачається, що даний процес обумовлений проникненням шумів вихідних сигналів кожної КМЧ та КГВЧ у радіочастотні сигнали їх квантових дискримінаторів.

Це дозволяє зобразити поточний вектор стану у вигляді

$$\vec{\Phi}(k) = \tilde{\vec{\Phi}}(k) + \hat{\vec{\Phi}}(k), \tag{9}$$

де $\tilde{\vec{\Phi}}(k) = \left[\tilde{\phi}_{ij}(k)\right]$ – вектор регулярних складових різниці фаз вихідних сигналів КМЧ та КГВЧ; $\hat{\vec{\Phi}}(k) = \left[\hat{\phi}_{ij}(k)\right]$ – вектор стохастичної складової різниці фаз вихідних сигналів КМЧ та КГВЧ, методологія ідентифікації якого описана в [24].

Підставляння (9) в (7) призводить до того, що рівняння (7) саме стає суперпозицією двох рівнянь, кожне з яких визначає внесок попереднього стану складових процесів (детермінованого і стохастичного) у поточний стан (10). Зважаючи на те, що аналізується стаціонарний режим роботи групової міри, а також те, що в стаціонарному режимі варіація $\hat{\phi}_{ij}(k) << 2\pi$ є незначною, систему (10) можна лінеаризувати розкладанням правої частини в ряд Тейлора відносно $\tilde{\phi}_{ij}(k)$ до лінійного (першого) члену (11)

$$\tilde{\phi}_{ij}(k+1) + \hat{\phi}_{ij}(k+1) = \tilde{\phi}_{ij}(k) + \tilde{\phi}_{ij}(k) + \tau\Delta_{ij} + \sum_{\substack{n=1 \\ n\neq i}}^{N} \tau A_{in} \cos\left[\tilde{\phi}_{in}(k) + \hat{\phi}_{in}(k)\right] -$$

$$- \sum_{\substack{m=1 \\ m\neq j}}^{N} \tau A_{jm} \cos\left[\tilde{\phi}_{jm}(k) + \hat{\phi}_{jm}(k)\right] + \bar{\zeta}_{ij}(k). \tag{10}$$

$$\hat{\phi}_{ij}(k+1) = \hat{\phi}_{ij}(k) + \tilde{\phi}_{ij}(k) + \tau\Delta_{ij} - \tilde{\phi}_{ij}(k+1) + \sum_{\substack{n=1 \\ n\neq i}}^{N} \tau A_{in} \left[\cos\tilde{\phi}_{in}(k) - \hat{\phi}_{in}(k)\sin\tilde{\phi}_{in}(k)\right] +$$

$$+ \sum_{\substack{m=1 \\ m\neq j}}^{N} \tau A_{jm} \left[\cos\tilde{\phi}_{jm}(k) - \hat{\phi}_{jm}(k)\sin\tilde{\phi}_{jm}(k)\right] + \bar{\zeta}_{ij}(k). \tag{11}$$

В (11) присутні усі складові рівняння, що і в (1). Тому віднімемо рівняння (1), попередньо представивши його в кінцево-різницевій формі, від рівняння (11).

Тоді для складових $\hat{\phi}_{ij}(k)$ справедливим буде таке співвідношення:

$$\hat{\phi}_{ij}(k+1)=\hat{\phi}_{ij}(k)-\sum_{\substack{n=1\\n\neq i}}^{N}\tau A_{in}\hat{\phi}_{in}(k)\sin\tilde{\phi}_{in}(k)-\sum_{\substack{m=1\\m\neq j}}^{N}\tau A_{jm}\hat{\phi}_{jm}(k)\sin\tilde{\phi}_{jm}(k)+\overline{\zeta}_{ij}(k). \qquad (12)$$

Вираз (12) є одним з рівнянь кінцевої різниці класичної системи стохастичних лінійних диференціальних рівнянь, що може бути зображено у векторній формі

$$\vec{\Phi}(k+1)=A(k)\vec{\Phi}(k)+\vec{\Xi}(k), \qquad (13)$$

де $A(k)=\left[a_{in}=\frac{\alpha_{in}}{2}\frac{A_n}{A_i}\sin\tilde{\phi}_{in}(k)\right]$ – фундаментальна перехідна матриця; $\vec{\Xi}(k)=\left[\overline{\zeta}_{in}(k)\right]$ – вектор адитивних шумів різниці фаз вихідних сигналів КМЧ.

Виходячи зі свого визначення, елементи a_{ij} матриці $A(k)$ складаються з добутку двох величин $\frac{\alpha_{in}}{2}\frac{A_n}{A_i}$ і $\sin\tilde{\phi}_{in}(k)$. Спосіб визначення першого співмножника викладено в [29]. Другий співмножник є тригонометричною функцією аргументу $\tilde{\phi}_{in}(k)$, що може бути виміряною прямими методами за допомогою частотного (фазового) компаратора. Проте, при цьому необхідно враховувати два фактори. По-перше, результат $y_{ij}(k)$ виміру компаратора має власні частотні і фазові шуми з коваріаційною матрицею R. По-друге, компаратор вимірює різницю фаз з точністю до деякої константи, що у цьому випадку є істотною, оскільки по суті визначає зрушення фази гармонійного співмножника елементів матриці $A(k)$.

З урахуванням зазначених факторів, стохастичний процес, що задовольняє рішенню рівняння (13), будемо розглядати як нестаціонарний випадковий марківський процес з перехідною матрицею, що змінюється у часі (від ітерації до ітерації). Ідентифікація стану еталона на основі рівняння (13) належить до класу некоректних задач математичної фізики, одержати рішення (оптимальної згладженої оцінки стану) якої можна шляхом застосування регуляризувальної процедури, що одержала узагальнену назву "лінійний фільтр Калмана". Для цього необхідно доповнити рівняння (13) рівнянням виміру з відомими параметрами.

Крім того, стосовно рівняння (13) необхідно знати: матрицю $A(k)$, початкове значення елементів $\hat{\phi}_{in,0}=\hat{\phi}_{in}(0)$ вектора $\vec{\hat{\Phi}}$, матрицю $R_{\Xi}(\tau)=M\left[\vec{\Xi}(t)\vec{\Xi}^{T}(t+\tau)\right]=Q\delta(\tau)$, коваріаційну матрицю шумів вимірювача R, початкове значення коваріаційної матриці $P(0)=M\left[\vec{\Psi}(0)\vec{\Psi}^{T}(0)\right]$, де $\vec{\Psi}(0)=\left[\hat{\psi}_i(0)\right]$ – вектор стохастичних складових фаз вихідних сигналів КМЧ при $k=0$.

Узагальнені амплітуди A_{in}, що входять до складу матриці $A(k)$, і початкові значення різниць фаз $\tilde{\phi}_{in,0}=\tilde{\phi}_{in}(0)$ можуть бути визначені за допомогою алгоритму, викладеному в [29].

Матриця $R_{\Xi}(\tau)$ являє собою діагональну матрицю. При цьому кожний діагональний елемент являє собою квадрат СКВ частоти відповідної міри, кількісне значення якого можна взяти з паспортних даних на КМЧ або з результатів останньої її повірки.

Як показує практика, застосування методу фільтра Калмана, значення вектора $\vec{\Psi}(0)$ й матриці $P(0)$ необхідні для ініціалізації процедури фільтрації й істотно впливають лише на

початковий етап оцінювання стану системи. У міру збільшення аргументу k (у стаціонарному режимі роботи фільтра) вплив похибок визначення початкових значень прагне до нуля. Зведення рекурентних співвідношень, які реалізують оцінку значень фаз вихідних сигналів КМЧ, що входять до складу групи, наведено в таблиці. Дані рекурентні співвідношення являють собою реалізацію методу фільтра Калмана, що належить до класу наближених алгоритмів фільтрації на основі локальної апроксимації "точного" рішення нелінійного рівняння стану.

Метод фільтра Калмана для вектора оцінок стану значень фаз сигналів КМЧ та КГВЧ

№ п/п	Послідовність дій
1	Початкові умови: $\vec{\tilde{\Phi}}(0,0)=0$; $\vec{\tilde{\tilde{\Phi}}}(0,0)=\left[\tilde{\phi}_{ij}(0)\right]$; $P(0,0)=Q$; $\vec{\tilde{\Psi}}(0)$=0.
2	Оцінка стану фаз КМЧ та КГВЧ: $\psi_i(k+1\|k)$ див. вираз (8) по отриманій оцінці $\phi_{ij}(k\|k)$ (див. п. 10).
3	Оцінка регулярної складової: $\vec{\tilde{\tilde{\Phi}}}(k\mid k-1)=F\left[\vec{\tilde{\tilde{\Phi}}}(k-1\mid k-1)\right]$, де $F[*]$ визначається рівнянням (4).
4	Оцінка вектора умовного середнього: $\vec{\tilde{\Phi}}(k\|k-1)=A(k-1)\vec{\tilde{\Phi}}(k-1\|k-1)$
5	Коваріаційна матриця умовного середнього: $P(k\|k-1)=A(k-1)P(k-1\|k-1)A(k-1)^T+Q$
6	Коефіцієнт підсилення: $K(k)=P(k\|k-1)\left[P(k\|k-1)+R\right]^{-1}$
7	Вектор нев'язок вимірів: $\vec{\upsilon}(k)=\vec{Y}(k)-\vec{\Phi}(k\|k-1)$
8	Оцінка стохастичної складової: $\vec{\tilde{\Phi}}(k\|k)=\vec{\tilde{\Phi}}(k-1\|k-1)+K(k)\vec{\upsilon}(k)$
9	Коваріаційна матриця: $P(k\|k)=[I-K(k)]P(k\|k-1)[I-K(k)]^T+K(k)RK(k)^T$
10	Оцінка повного вектора стану: $\vec{\Phi}(k\|k)=\vec{\tilde{\tilde{\Phi}}}(k\|k-1)+\vec{\tilde{\Phi}}(k\mid k)$

Результати моделювання ідентифікації вектора фаз вихідних сигналів групового еталону, що складається з двох КМЧ та КГВЧ, звірюваних між собою по повному графу звірень, на основі запропонованого методу, наведені на рис. 1. Штриховими лініями зображено поведінку фаз вихідних сигналів кожного КМЧ та КГВЧ, а безперервними лініями – відповідні фази після видалення з них результатів оцінок, отриманих за допомогою запропонованого методу. Безперервна лінія на рис. 1 визначає зсув в оцінках рішення вихідної системи рівнянь і по суті є похибкою лінеаризації вихідної нелінійної системи рівнянь.

Як показано в [29], критерієм наявності оптимальної незміщеної оцінки вектора стану $\vec{\Phi}(k\mid k)$ є рівність нулю величини $M[\vec{\upsilon}(k)]$.

На рис. 2 наведено поведінку елемента вектора нев'язок $\vec{\upsilon}(k)$ у часі. Видно, що його поточне значення еквівалентно поведінці центрованого випадкового процесу, що свідчить про відсутність зсуву в отриманих оцінках поточних значень фаз вихідних сигналів КМЧ та КГВЧ.

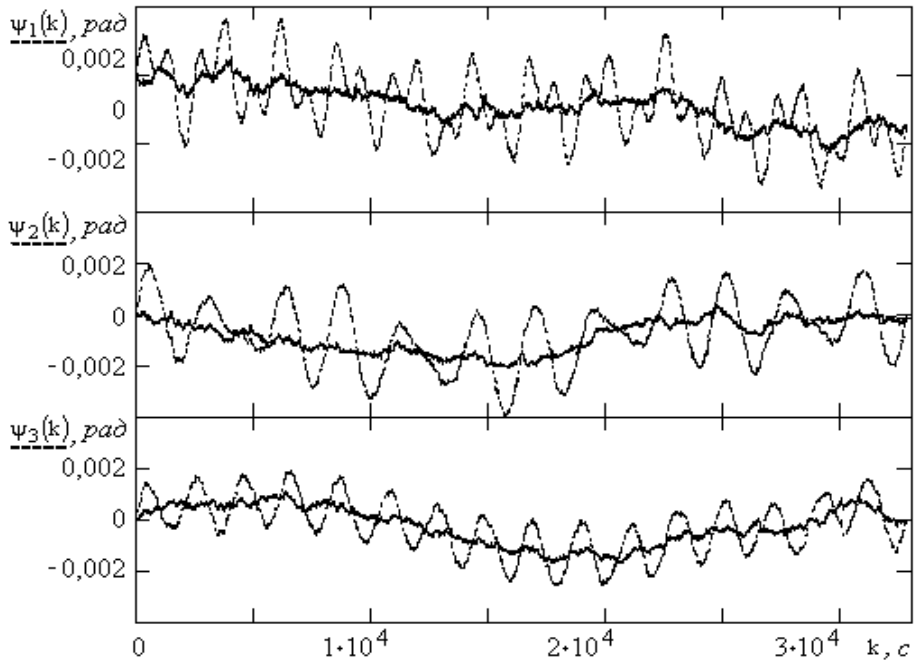


Рис. 1. Оцінка вектора фаз вихідних сигналів групового еталону, що складається з двох КМЧ та КГВЧ, звірюваних між собою по повному графу звірень

Проте, при застосуванні методу фільтра Калмана необхідно враховувати, що лінеаризація вихідної нелінійної системи рівнянь призводить до появи зсувів в оцінці рішення вихідної системи рівнянь. При цьому математичне очікування вектора нев'язок у фільтрі Калмана хоча й дорівнює нулю, але вектор нев'язок є кольоровим шумом. Цей факт може бути використаний у побудові адаптивних алгоритмів Калманівської фільтрації, що є проміжними між алгоритмами, заснованими на локальній апроксимації, і більш універсальними (але більш складними й громіздкими) алгоритмами, заснованими на інтегральній апроксимації.

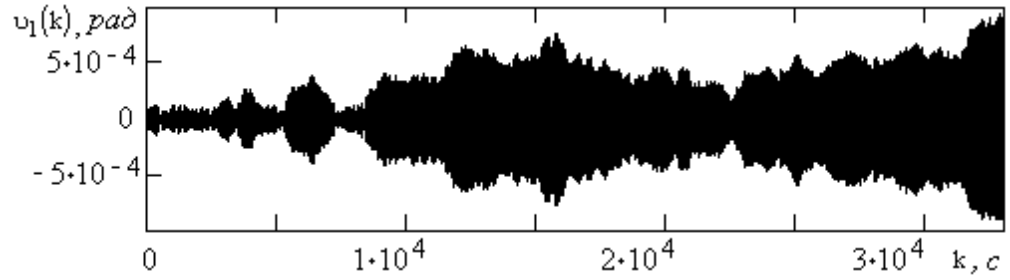


Рис. 2. Поведінка елементів вектора нев'язок у процесі ідентифікації системи стохастичної моделі системи пов'язаних осциляторів

Для розширення області існування оптимальних оцінок вектора стану $\vec{\Psi}(k)$ врахуємо ряд положень теорії оптимального керування [23]. Так, у теорії оптимального керування також доводиться мати справу з ідентифікацією систем за результатами вимірів і формуванням керуючих сигналів, пропорційних нев'язкам між оцінками умовного середнього й результатами вимірів.

Тому пропонується рівняння стану (6) доповнити складовою, що описує реакцію системи на уведене керування, тобто рівняння стану перепишемо у вигляді

$$\vec{\Psi}(k) = \Lambda(k, k-1)\vec{\Psi}(k-1) + B(k, k-1)\vec{U}(k) + \vec{\xi}(k), \qquad (14)$$

де $\vec{\Psi}(k)=[\psi_i(k)]$ – вектор стану (флуктуацій фази вихідних сигналів) КМЧ та КГВЧ; $\Lambda(k,k-1)$ – фундаментальна перехідна матриця; $B(k,k-1)$ – передатна матриця керування (виправлень); $\vec{U}(k)=[-\psi_i(k|k-1)]$ – вектор керування (виправлень); $\vec{\xi}(k)=[\xi_i(k)]$ – вектор адитивних шумів КМЧ та КГВЧ.

Порівнюючи (14) з (11), можна зробити висновок, що частину рівняння (11), яка описує поведінку різниць фаз вихідних сигналів мір частоти, може бути зображено як результат впливу керуючого сигналу $\vec{U}(k)$. В [22] показано, що фільтр Калмана дозволяє одержати оптимальну незміщену оцінку стану з мінімальною дисперсією тільки в тому випадку, якщо елементи матриць $\Lambda(k,k-1)$ і $B(k,k-1)$ задані коректно (оцінені правильно), а також показано, що при відхиленні оцінок елементів матриць $\Lambda(k,k-1)$ і $B(k,k-1)$ від оптимальних виявляються відхилення від оптимальності (зсув) у поведінці відновлень або нев'язок вимірів. Цей висновок є основним при розробці алгоритму ідентифікації системи, коли встановлюється й доводиться необхідна й достатня умова того, щоб помилки в елементах матриць $\Lambda(k,k-1)$ і $B(k,k-1)$ створювали зсув. Поширення даного положення на рівняння (11) дозволяє зробити твердження про те, що причиною виникнення зсувів у векторі нев'язок (вираз п. 7 у таблиці) є невірна оцінка передатної матриці $B(k,k-1)$ вектора керування. Дійсно, оцінки вектора керування, особливо відносно визначення дійсних значень різниць частот Δ_{ij}, можуть бути отримані як середнє на кінцевому інтервалі спостереження за системою й містити в собі як випадкові, так і систематичні складові похибок визначення.

Якщо матрицю $B(k,k-1)$ представити в діагональному виді, кожний діагональний елемент якої буде визначати відповідне середнє значення всіх різниць частот Δ_{ij}, то можна припустити, що обрана матриця неточно характеризує спостережувану систему, тобто точна матриця описується співвідношенням

$$\hat{B}(k,k-1)=B(k,k-1)-\delta B(k,k-1), \tag{15}$$

де $\hat{B}(k,k-1)$ – точне значення передатної матриці; $\delta B(k,k-1)$ – невідома похибка передатної матриці.

У цьому випадку дійсний стан системи (14) буде визначатися виразом

$$\vec{\Psi}_k=\Lambda(k,k-1)\vec{\Psi}_{k-1}+\hat{B}(k,k-1)\vec{U}_k+\vec{\xi}_k, \tag{16}$$

де $\vec{U}_k$ – точне значення вектора керування (виправлення) КМЧ та КГВЧ.

Оскільки матриця $B(k,k-1)$ задана не точно, у загальному випадку рівняння (15) стає неточним, тобто стають відмінними від нуля такі співвідношення, що описують умовне середнє:

$$\vec{m}(k\mid k)=M\left[\vec{\Psi}_k\right]-M\left[\vec{\Psi}(k\mid k)\right]; \tag{17}$$

$$\vec{m}(k\mid k-1)=M\left[\vec{\Psi}_k\right]-M\left[\vec{\Psi}(k\mid k-1)\right]. \tag{18}$$

Застосування операцій усереднення до рекурентних співвідношень фільтра Калмана з урахуванням (17), (18) дає такі зв'язки вектора середніх значень $\vec{m}(k\mid k)$, $\vec{m}(k\mid k-1)$ з матрицею помилок $\delta B(k,k-1)$:

$$\vec{m}(k\mid k)=[I-K(k)]\Lambda(k,k-1)\vec{m}(k-1\mid k-1)+[I-K(k)]\delta B(k,k-1)\vec{U}_k,$$

$$\vec{m}(k\mid k-1)=\Lambda(k,k-1)\vec{m}(k-1\mid k-1)-\delta B(k,k-1)\vec{U}_k.$$

В [22] показано, що існують граничні співвідношення:

$$\lim_{k\to\infty} M[\vec{\upsilon}(k)]\to\lim_{k\to\infty} M[\vec{m}(k\mid k-1)]=-[I-\Lambda(k,k-1)[I-K(k)]]^{-1}\delta B(k,k-1)\vec{U}_k.$$

Даний вираз являє собою функціонал, мінімізуючи який щодо елементів матриці $\delta B(k,k-1)$, можна одержати оцінки елементів даної матриці й тим самим мінімізувати похибку визначення вектора стану $\vec{\Psi}(k)$.

Вказане дозволяє записати рекурентну процедуру визначення елементів матриці $\delta B(k,k-1)$, яка органічно вписується в рекурентну процедуру фільтра Калмана (зведення розрахункових формул якого наведено в таблиці):

$$\frac{\partial \vec{m}(k\,|\,k)}{\partial \Delta_{ij}} = \left[I - K(k)\right]\left[\Lambda(k,k-1)\frac{\partial \vec{m}(k\,|\,k-1)}{\partial \Delta_{ij}} - \frac{\partial \delta B(k,k-1)}{\partial \Delta_{ij}}\right];$$

$$\frac{\partial \vec{m}(k+1\,|\,k)}{\partial \Delta_{ij}} = \Lambda(k,k-1)\frac{\partial \vec{m}(k\,|\,k)}{\partial \Delta_{ij}} - \frac{\partial \delta B(k,k-1)}{\partial \Delta_{ij}}.$$

Дані рівняння є стійкими різницевими рівняннями, тому вплив початкових умов згодом стає усе менш значним (ситуація, аналогічна з початковими умовами для фільтра Калмана). У зв'язку із цим, розглядаючи асимптотичну поведінку системи (принаймні поведінку системи після великої кількості вимірів), ці рівняння будемо вирішувати з початковою умовою $\frac{\partial m(k\,|\,k)}{\partial \Delta_{ij}} = 0$.

Внаслідок цього вирази (14) і (16) дозволяють враховувати похибку лінеаризації на основі визначення (уточнення) матриць $\Lambda(k,k-1)$ і $\delta B(k,k-1)$.

Через те, що оцінки знаходять шляхом проведення обчислювального експерименту на математичній моделі групового еталону, а не в результаті натурного експерименту, пропонуються такі заходи. Для реалізації локальної апроксимації «точного» рішення нелінійного рівняння стану в умовах проведення натурного експерименту необхідно використовувати оптимальний по збіжності й можливості апаратного керування (підстроювання частоти рубідієвої міри частоти полем «С») режим биттів. При цьому розведення частот вихідних сигналів КМЧ та КГВЧ у групі на інтервал, що перевершує смугу захвата частот, хоч і призводить до модуляції частоти вихідного сигналу кожної міри обмеженим набором квазігармонійних сигналів, проте дозволяє прогнозувати результуюче відхилення частоти й надалі компенсувати його. Внаслідок цього вдається одержати апріорну інформацію про тип рішення, одержати оцінки матриці коваріації похибки і вектора математичного очікування правої частини й рішення даної системи флуктуаційних рівнянь частоти (фази) коливань.

Принцип дії прототипу КГВЧ заснований на вимірюванні шумів вихідного сигналу КД на частоті електромагнітного поглинання при переході атомів рубідію з одного енергетичного стану в інший. Зовнішній вигляд експериментальної установки наведено на рис. 3. При виконанні робіт з дослідження квантових шумів КД виникла необхідність розробки плати IEEE 488 (керування), комутатора сигналів, компараторів (на рис. 3 частотомір Ч3-64/1 – 4 та цифровий осцилограф типу SDS1102CML – 5). Для одержання точного значення частоти переходу в атомах Rb87 вихідна напруга частотою 90 MHz у помножувачі прототипу КГВЧ (на рис. 3 прилад з приймачем сигналів GPS/GLONASS – 6) змішується з сигналом, що виробляється синтезатором А3 другої КМЧ типу СЧВ-74 (на рис. 3 прилад – 3). Варакторний діод КД одночасно виконує функції помножувача та змішувача, а резонатор КД (на рис. 3 прилад – 1), настроєний на частоту переходу, що дорівнює:

$$f_{0-0} = f_{vco} \cdot n_{mul} \cdot m_{mul} - f_{dds} \cdot k_m.$$

Тут f_{0-0} = 6834,682540 MHz, f_{vco} =5 MHz – частота керованого опорного генератору, n_{mul} =18 – коефіцієнт множення помножувача, m_{mul} =76 – коефіцієнт множення генератора гармонік, k_m =1, f_{dds} = 5,317460 MHz ± 4 kHz – частота цифрового синтезатора визначається

частотою 0-0 переходу Rb^{87} ячейки поглинання. Таким чином, змінюючи частоту синтезатора f_{dds}, можна з високою точністю налаштуватися на частоту збудження КД і, відповідно, виділити квантові фазові шуми.

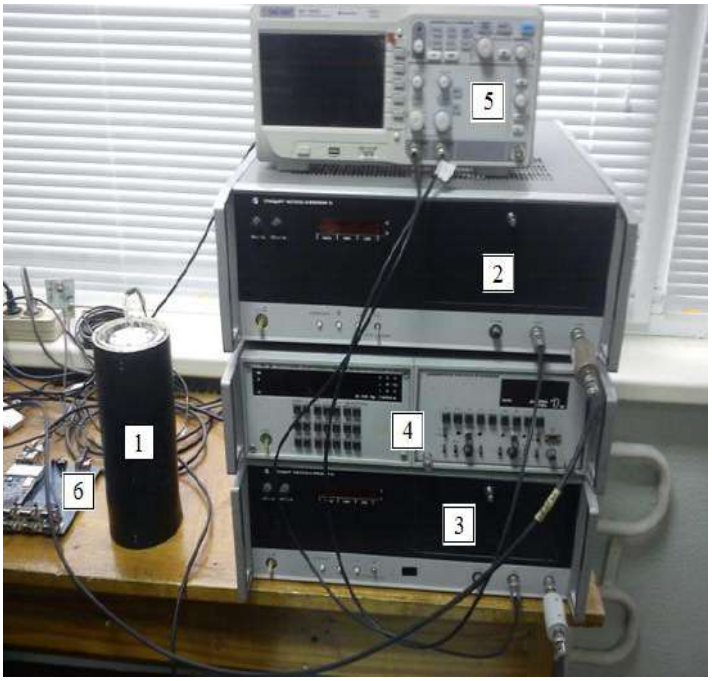


Рис. 3. Зовнішній вигляд експериментальної установки (групового еталону), яка складається з двох КМЧ та прототипу КГВЧ

На схемі приймач сигналів GPS/GLONASS синхронізується сигналом опорної (першої) КМЧ частотою 5 MHz. Радіонавігаційні сигнали, прийняті антеною, надходять на антенний вхід приймача сигналів GPS/GLONASS, з виходу якого сигнал "1 PPS" (апаратна мітка часу) подається на вхід "Б" частотоміра типу Ч3-64/1. Секундна мітка від опорного (першого) генератора типу СЧВ-74 подається на вхід "А" частотоміра Ч3-64/1. При цьому частотомір працює в режимі виміру інтервалів часу. Інформація з виходу частотоміра Ч3 – 64/1 через інтерфейс IEEE488 передається в ПЕОМ для подальшої обробки й зберігання. ПЕОМ обладнана платою контролера інтерфейсу IEEE°488.

З виходу КД сигнал помилки надходить на осцилограф SDS1102CML та підсилювач низької частоти, з виходу якого посилений сигнал помилки подається на синхронний детектор модулятора КД та далі на блок синхронізації від другої КМЧ типу СЧВ-74 (на рис.°3 прилад°–°3). З виходу синхронного детектора напруга постійного струму надходить на інтегруючий підсилювач, що збільшує коефіцієнт регулювання опорного генератора КГВЧ системи імпульсного цифрового фазового автопідстроювання частоти без порушення її стійкості. Напруга з виходу інтегруючого підсилювача подається на керуючий елемент опорного генератора КД і впливає на нього так, щоб звести розстройку КД до нуля. Опорний генератор КГВЧ засинхронізований від першої КМЧ типу СЧВ-74 (на рис. °3 прилад°–°2).

Таким чином, частота опорного генератора КД в режимі автопідстройки буде дорівнювати $f_{vco} = (f_{0-0} + f_{dds} k_m)(n_{mul} m_{mul})^{-1}$.

У першому контурі автопідстроювання по частоті використовувалася перша КМЧ типу СЧВ-74. Рубідієві КМЧ типу СЧВ-74 мають нестабільність частоти вихідного сигналу за добу: $\leq \pm 1 \cdot 10^{-12}$. Замість КМЧ для частотної синхронізації КГВЧ можна використовувати метод загального охоплення для синхронізації опорного генератора КД по сигналам TV каналів [31]. При цьому для синхронізації опорного генератора КГВЧ сигналами 1 PPS по сигналам

TV каналів необхідно застосовувати також комплект апаратури GPS/GLONASS з похибкою у межах $\leq \pm 30$ ns. Це дозволить виключити похибку затримки радіотелевізійного передавального центру й мінімізувати похибку траси поширення.

Встановлено, що квантовий фазовий шум має гаусовський закон розподілу імовірності, тому для збільшення мінімальної ентропії та швидкодії необхідно використовувати методи та засоби збільшення ентропії. Тому в прототипі КГВЧ використовується спеціальний засіб збільшення ентропії – квантовий екстрактор [32].

Таким чином, запропонований метод вимірювання квантового фазового шуму та ширини лінії робочого переходу радіооптичної системи ГВЧ заснований на регуляризації задачі ідентифікації параметрів КГВЧ групою КМЧ за наявності похибки від взаємодії. Даний метод за результатами їх взаємних звірень дозволяє одержувати оцінки неспостережуваного процесу зміни поточних значень фази вихідного сигналу (вектора стану) КГВЧ та кожної КМЧ на основі апріорної інформації про групу як стохастичної системи зв'язаних осциляторів.

Висновки

Застосування методу ПРР в КГВЧ має великий потенціал для досягнення високої швидкості формування квантових дискретних випадкових послідовностей, достовірність яких може бути підтверджена метрологічним сертифікатом відповідності. На відміну від інших схем КГВЧ, даний метод дозволяє відпрацювати практичні схеми більшості методів генерації випадкових чисел. Тому запропонована перспективна схема високошвидкісного квантового ГВЧ заснована на вимірі випадкових коливань оптичного поля у ізотопах рубідію (Rb^{87}та Rb^{85}) КД. Запропонований метод вимірювання квантових флуктуацій фази спектрального джерела та ячейки поглинання має швидкодію порядку 1000 bit/s на один фотоприймач. Одним з важливих переваг цього підходу є високий потенціал швидкості генерації випадкових чисел (більш десятка Mbit/s) за рахунок використання матриці з фотодетекторів, яка може нараховувати декілька мільйонів одиниць фотодіодів.

Однак на даний час недостатньо вивчені квантові шуми інтенсивності радіооптичного резонансу при різних режимах оптичного накачування випромінюванням лампового спектрального джерела. Крім цього, основним зовнішнім чинником, що впливає на криптографічні параметри макету КГВЧ, є температура навколишнього середовища. Її вплив зводиться до температурного зрушення опорної частоти резонансу через недостатньо точний підбір складу суміші буферних газів в резонансних ячейках та затягуванням частоти СВЧ резонатором. Даний вплив на криптографічні параметри перспективного КГВЧ також вивчається.

Практична цінність роботи полягає у вирішенні питань, спрямованих на створення КГВЧ нового покоління, здатних поліпшити як свої криптографічні, так і масогабаритні характеристики за рахунок застосування в них різних джерел оптичного накачування парів рубідію. З застосуванням в КД напівпровідникових лазерів типу VCSEL, що випромінюють в діапазоні 780-900 nm, можуть бути створені достатньо прості джерела оптичного накачування для КГВЧ. Перспективним напрямком у побудові малогаборитних і швидкодіючих КГВЧ є використання фазового методу вимірювань в Λ-схемах з електромагнітно-індукованої прозорістю. Це дозволить значно зменшити масогабаритні характеристики прототипу КГВЧ.

Експериментальне підтвердження отриманих результатів проведено на прототипі КГВЧ з КД на парах рубідію Rb^{87}та Rb^{85}. Вихідні послідовності екстрактора перспективного КГВЧ успішно проходять статистичні тести DIEHARD і NIST STS.

Список літератури:

1. Горбенко І.Д., Горбенко Ю.І. Прикладна криптологія. Теорія. Практика. Застосування / І.Д. Горбенко // Харків : Форт, 2012. 878 с.

2. Гріненко Т.О. Квантові генератори випадкових чисел в криптографії / Т.О. Гріненко, О.П. Нарєжній // Системи обробки інформації : зб. наук. праць. Харків : ХУПС, 2015. Вип. 10(135). С. 86-89.

3. A Fast and Compact Quantum Random Number Generator/ Thomas Jennewien, Ulrich Achleitner, Gregor Weihs, Harald Weinfurter and Anton Zeilinger 4/III D-80799 Munchen, Germany February 1, 2008. pp. 1–21. [Електронний ресурс] Режим доступу до матеріалів: https://arxiv.org/pdf/quant-ph/9912118.pdf.

4. U. Achleitner, Diploma Thesis, Innsbruck University (1997).
5. A. J. Martino, G. M. Morris, Applied Optics 30, 981 (1991).
6. G. M. Morris, Opt. Engin. 24, 86 (1985); J. Marron, A. J. Martino, G. M. Morris, Applied Optics 25, 26 (1986).
7. W. M. Itano, J. C. Bergquist, R. G. Hulet, and D. J. Wineland, Phys. Rev. Lett. 59, 2732 (1987).
8. Th. Sauter, W. Neuhauser, R. Blatt, and P. E. Toschek, Phys. Rev. Lett. 57, 1696 (1986).
9. Osung Kwon, Young-Wook Cho, and Yoon-Ho Kim. Quantum Random Number Generator using Photon-Number Path Entanglement. Department of Physics, Pohang University of Science and Technology (POSTECH), Pohang, 790-784, Korea-2013.
10. Kwon O., Cho Y.-W., Kim Y.-H. Quantum Random Number Generator using Photon-Number Path Entanglement // arXiv:0807.3440v2 [quant-ph] 4Aug 2008. pp. 1–4. [Електронний ресурс] Режим доступу: http://www.researchgate.net/publication/24218868.
11. Y.-H. Kim Phys. Rev. A 68, 013804 (2003).
12. Ritter T. // Cryptologia. Vol. 15, pp. 81 1991.
13. Stipčevića M., Medved Rogina B. Quantum random number generator based on photonic emission in semiconductors // Review of Scientific Instruments. Vol. 78 2007. pp. 1–7. [Електронний ресурс] Режим доступу: http://rsi.aip.org/rsi/copyright.jsp.
14. Stipčevića M. // Review of Scientific Instruments. Vol. 75 2004. pp. 4442.
15. Feihu Xu, Bing Qi, Xiongfeng Ma, He Xu, Haoxuan Zheng. Ultrafast quantum random number generation // Optics Express. 2012. Vol. 20. No. 11.
16. Qi B., Chi Y-M., Lo H-K., Qian L. Experimental demonstration of a high speed quantum random number generations cheme based on measuring phase noise of a single mode laser // Optics Letters. 2010. Vol. 35. pp. 312-314. [Електронний ресурс] Режим доступу: [arXiv:0908.3351v2 [quant-ph] 27 Aug 2009]: http://arxiv.org/abs/0908.3351.
17. V. Jacques, E. Wu, F. Grosshans, F. Treussart, P. Grangier, A. Aspect, and J.-F. Roch, Science 315, 966 (2007).
18. I. Goldbergand D. Wagner, Dr. Dobb's Journal, pp. 66-70 (1996).
19. ID Quantigue White Paper. Random number generation using quantum physics. Version 3.0, April 2010. http://www.idquantique.com.
20. Grinenko T. O., Narezhniy O.P., Gorbenko I.D. Methods for measuring the noise power spectral density of the random number generator quantum radio optical system // Telecommunications and Radio Engineering. 2017. Vol. 76. Issue 7. pp. 635-651. DOI: 10.1615/TelecomRadEng.v76.i7.60.
21. Стандарты частоты: принципы и приложения / Ф. Риле ; пер. с англ. Н. Н. Колачевского. Москва : Физматлит, 2009. 511 с.
22. Р.Ф. Оэп, А.Р. Стабберуд Фильтрация и стохастическое управление в динамических системах. Пер. с англ. ; под ред. К.Т. Леондеса. Москва : Мир, 1980. 408 с.
23. Л. Льюнг Идентификация систем. Теория для пользователя ; пер. с англ. ; под ред. Я.З. Цыпкина. Москва : Наука. Гл. ред. физ.-мат. лит., 1991. 432 с.
24. Чинков В.Н., Нарежний А.П. Математическая модель формирования групповой шкалы времени при условии взаимодействия атомных часов как системы связанных осцилляторов // Радіоелектронні і комп'ютерні системи. 2005. № 3(11). С. 5-9.
25. А.Ф. Верлань, В.С. Сизиков Интегральные уравнения: Методы, алгоритмы, программы. Справочное пособие. Киев: Наук. думка, 1986. 544 с.
26. Тихонов А.Н., Арсенин В.Я. Методы решения некорректных задач. Москва : Наука, 1974. 224 с.
27. Боголюбов Н.Н., Митропольский Ю.А. Асимптотические методы в теории нелинейных колебаний. Москва : Наука, 1974. 408 с.
28. Нарежний А.П. Идентификация скрытых периодичностей в нестационарных фазовых флуктуациях прецизионных мер частоты // Прикладная радиоэлектроника. 2005. Т.4. № 2. С. 148–152.
29. Евдокименко Ю.И., Нарежний А.П. Идентификация групповой меры частоты с использованием итерационных методов решения стационарных задач // Радиотехника. 1998. №109. С. 76–80.
30. Чинков В.Н, Нарежний А.П. Исследование режимов взаимодействия прецизионных мер частоты с близкими частотами // Авиационно-космическая техника и технология. 2005. №5(21). С. 52–56.
31. Гриненко Т.А. Устройство поддержки синхронизации по телевизионному сигналу для цифровой сети связи АСУ тп / Т.А. Гриненко, А.А. Костыря, А.П. Нарежний // Метрологія та прилади. 2014. №4. С. 44–50.
32. Нарєжній О.П. Метод побудови алгоритму екстрактора на основі багатомодульного перетворення для перспективного квантового генератора випадкових чисел / О.П. Нарєжній, Т.О. Гріненко // Математичне та комп'ютерне моделювання. Серія: Фізико-математичні науки : зб. наук. праць / Інститут кібернетики імені В.М. Глушкова НАНУ Кам'янець-Подільський : Кам'янець-Подільський нац. ун-т імені Івана Огієнка, 2017. Вип. 15. С. 126–132.

Харківський національний
університет радіоелектроніки;
Харківський національний
університет імені В.Н.Каразіна

Надійшла до редколегії 09.03.2018

УДК 004.652

В.И. ЕСИН, д-р техн. наук

ИНВАРИАНТНАЯ К ПРЕДМЕТНЫМ ОБЛАСТЯМ СХЕМА БАЗЫ ДАННЫХ И ЕЕ ОТЛИЧИТЕЛЬНЫЕ ОСОБЕННОСТИ

Введение

Результаты исследований состояния информатизации в различных компаниях, организациях, учреждениях свидетельствуют о том, что многие из них владеют определенными информационными системами организационного управления (ИСОУ). При этом для решения возникающих новых задач, как правило, связанных с расширением деятельности и, соответственно, рассматриваемыми предметными областями (ПрО), они хотят иметь более функциональные, с улучшенными характеристиками качества информационные системы (ИС), требующие меньших затрат по сопровождению. В этих условиях востребованными становятся проекты: по разработке новых ИСОУ и их интеграции с существующими информационными системами; разработке новых ИСОУ с целью замены существующих ИС; модернизации существующих ИСОУ. Суть данных проектов заключается в проведении процедур реинжиниринга существующих ИС и их основного функционального компонента – базы данных (БД).

Одним из важных требований, предъявляемых к реинжинирингу существующих ИСОУ и их БД, является своевременность завершения соответствующих проектов в рамках запланированного бюджета с заданными характеристиками качества, которое, как показывают результаты анализа IT-проектов, проведенного международными экспертами, к сожалению, не всегда выполняется. Более 60 % проектов были провалены или завершены с опозданием, причем с гораздо большими затратами, чем планировалось [1, 2, 3].

Таким образом, имеет место проблема, связанная с необходимостью своевременного создания, модернизации в рамках запланированного бюджета информационных систем, обладающих требуемыми качествами, и ограниченностью возможностей существующих методов проектирования. В отношении реляционных баз данных (РБД) как получивших наибольшее распространение в ИСОУ указанная ограниченность возможностей обусловлена ориентацией традиционной методологии их проектирования на итерационную, достаточно трудоемкую процедуру создания уникальных концептуальной модели, логической и физической схем при разработке новой БД либо на существенное их преобразование при модернизации. Что часто приводит к расходованию значительных, не всегда прогнозируемых, временных и финансовых ресурсов. В результате возникает объективная необходимость пересмотра существующих подходов, методологий и технологий создания, модернизации баз данных, которые позволят избавиться от необходимости затратной политики выполнения лишних работ при реинжиниринге БД ИСОУ, в том числе и на этапе физического проектирования РБД.

Стремление избежать излишних затрат, свойственных традиционной методологии создания РБД, не только на этапах концептуального и логического проектирования, предлагаемые решения для которых достаточно подробно излагаются в работах [4 – 8], но и на этапе физического проектирования, актуализировало задачу разработки инвариантной к предметным областям схемы БД.

Основные отношения схемы базы данных

Создание инвариантной к ПрО схемы БД стало возможным благодаря разработанной модели данных с универсальным базисом отношений [7, 8]. Данная схема является отображением универсального базиса отношений и множества ограничений целостности модели [7, 8], которая в свою очередь является отображением модели данных «объект-событие» [4 – 6].

Основными разработанными и отличающими элементами инвариантной к ПрО схемы БД как описания содержания, структуры и ограничений целостности, используемых для создания и поддержки БД, являются: состав, структура базовых отношений; реализации ограничений целостности; средства, обеспечивающие безопасность БД.

Для лучшего понимания принципиальных отличительных особенностей базовых отношений (таблиц) инвариантной к предметным областям схемы БД вначале целесообразно отметить, что в схемах, проектируемых по традиционной технологии РБД, каждое базовое отношение, как правило, ассоциируется с конкретной сущностью моделируемой ПрО, а атрибут (столбец) этого отношения – со свойством сущности. При этом информация о базовых отношениях с указанием их имен, имен атрибутов, связанная с описанием моделируемой ПрО, а также типах данных атрибутов, связанных с типами данных конкретной СУБД, на платформе которой реализуется БД, размещается в системном каталоге (словаре) СУБД (рис. 1). Поэтому при таком построении схемы БД различные изменения в ПрО, даже незначительные, вызывают необходимость модернизации словаря СУБД, то есть приводят либо к незначительной, либо значительной модификации существующей схемы, вплоть до разработки новой. В процессе же осуществления последовательного отображения: модель «объект-событие» [4 – 6] – модель данных с универсальным базисом отношений [7, 8] – инвариантная к ПрО схема БД, создается набор фиксированных базовых отношений с их именами и именами атрибутов (заголовками столбцов), непосредственно не ассоциируемых с конкретными сущностями рассматриваемой ПрО и их свойствами.

Это объясняется тем, что семантические понятия/концепты (semantic concepts) модели «объект-событие», связываемые с именами соответствующих базовых отношений инвариантной к ПрО схемы БД (R^{sh}), являющихся отображением отношений $\Re$ модели с универсальным базисом отношений: $M_{umd} = \langle \Re, \mathrm{Pr}, L \rangle$, где Pr – множество ограничений целостности; L – язык манипулирования данными, – есть общие понятия-категории для всех ПрО. А имена атрибутов базовых отношений либо ассоциируются с именами множеств, соотносимых с такими семантическими понятиями, либо с множествами соответствующих им идентификаторов, либо с некоторыми ограничениями им присущими (рис. 2).

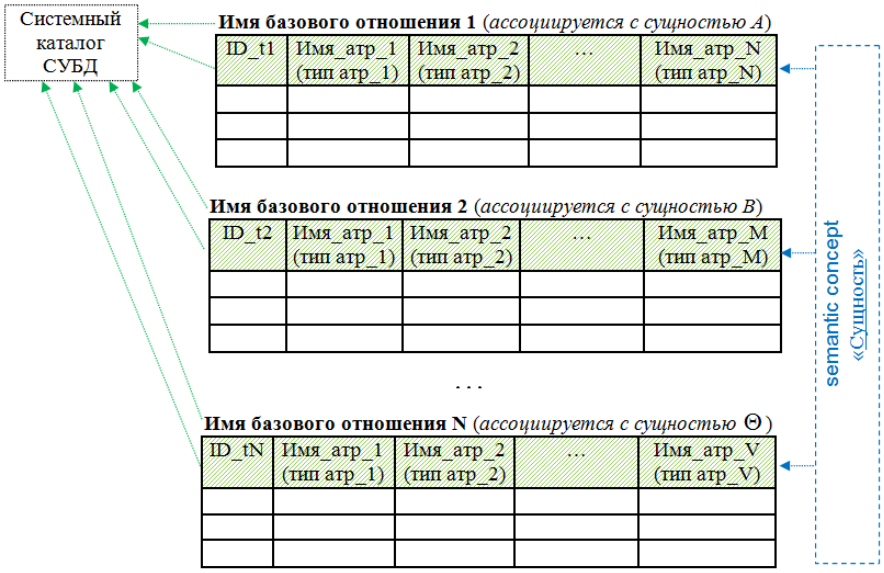


Рис. 1. Базовые отношения схемы БД, разрабатываемой по традиционной технологии, и их связь с сущностями моделируемой ПрО

Поэтому, когда такая информация вместе с указанием типа данных атрибутов, принятых в СУБД, на платформе которой реализуется инвариантная к ПрО схема БД, помещается в системный каталог СУБД, то различные изменения в предметной области не вызывают необходимость внесения в него корректив.

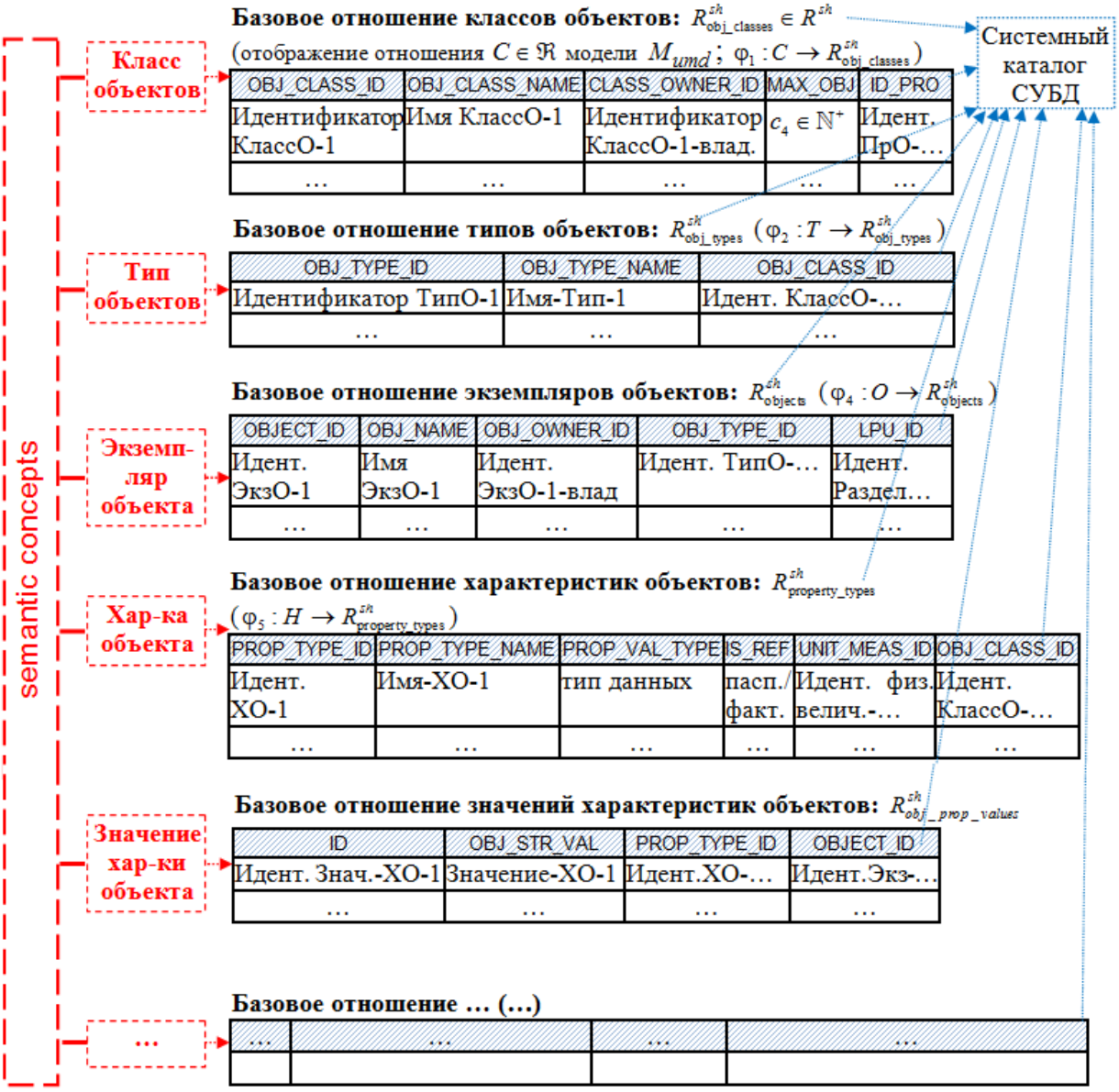


Рис. 2. Базовые отношения инвариантной к ПрО схемы БД
и их связь с семантическими концептами модели «объект-событие»

Описание моделируемой ПрО (сущностей, их свойств, связей), трансформированное при традиционной разработке РБД в набор именованных базовых отношений с атрибутами и связями между отношениями, в случае использования инвариантной к ПрО схемы БД отображается в область изменяемых данных фиксированных базовых отношений схемы, ассоциируемых с метаданными ПрО. Эта область определяется как область проблемных метаданных (выделена горизонтальными линиями в правых таблицах рис. 3, с их соответствием в левых таблицах, являющихся отображением моделируемой ПрО, выполненном при традиционной разработке РБД). В ней содержатся метаданные предметной области, используемые при определении ее данных. При этом данные моделируемой ПрО помещаются также в область изменяемых данных, но других отношений инвариантной к ПрО схемы БД, ассоциируемых с экземплярами объектов, событий, значений характеристик, конкретных документов. Эта область определяется как область данных (выделена точками рис. 3).

Такая организация данных обеспечивает совместное хранение метаданных и данных моделируемой ПрО в области изменяемых данных соответствующих базовых отношений инвариантной к ПрО схемы БД, в отличие от раздельного хранения метаданных ПрО в словаре СУБД и самих данных собственно в базе данных (в базовых отношениях ее схемы) при

традиционном построении РБД. В результате появляется гибкость – возможность описывать ПрО, в том числе и происходящие в ней изменения, на уровне проблемных метаданных, без изменения собственно структуры базовых отношений схемы БД. Это позволяет созданную базу данных достаточно просто адаптировать к изменениям в ПрО при стабильной структуре отношений схемы БД. Заранее неограниченное многообразие элементов ПрО распределяется по фиксированному набору базовых отношений инвариантной к ПрО схемы БД, обеспечивая при этом возможность одновременного хранения и использования данных различных существенно отличающихся ПрО.

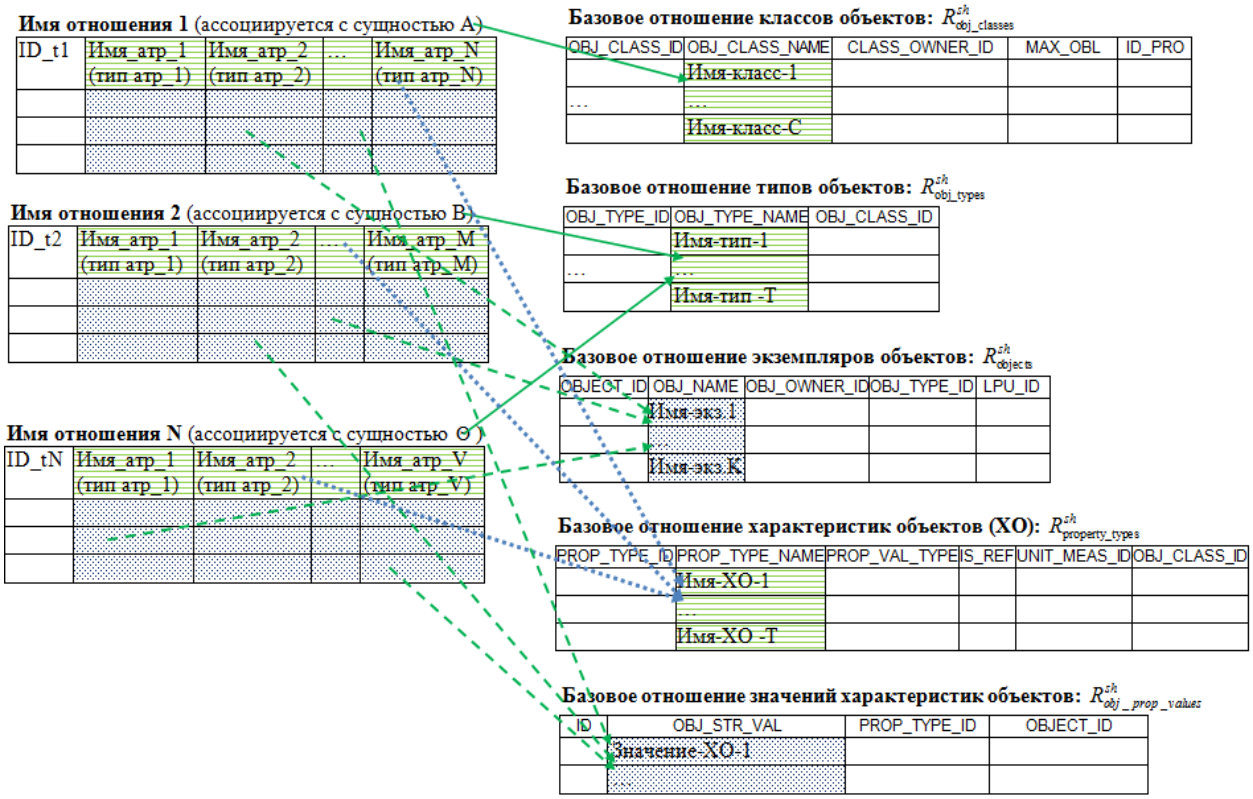


Рис. 3. Схематичное представление трансформации описания предметной области в инвариантную к ПрО схему БД и в схему РБД, спроектированную по традиционной технологии

Ниже приведены результаты отображения универсального базиса отношений $\Re$ модели M_{umd} в базовые отношения разрабатываемой схемы ($\varphi: \Re \to R^{sh}$), представленные в виде диаграммы в нотации IDEF1X (рис. 4).

Полученные базовые отношения инвариантной к ПрО схемы БД имеют принципиальные отличия в назначении, структуре, месте хранения описания метаданных моделируемой ПрО относительно создаваемых отношений при традиционной технологии проектирования реляционных БД. Их число, структура, в отличие от структуры и числа базовых отношений схем, разрабатываемых по традиционной технологии реляционных БД, не зависят от набора данных, они инвариантны к рассматриваемой ПрО.

Обеспечение поддержки ограничений целостности данных

С целью предотвратить появление в базе противоречивых данных с использованием средств декларативной и процедурной поддержки ограничений целостности были определены реализации ограничений целостности Pr^{sh} в создаваемой схеме как некоторый набор шаблонов конструкций операторов (программный код), реализующий ограничение, полученный в результате отображения множеств ограничений целостности Pr, которые специ-

фицируются в модели M_{umd} во множество ограничений целостности $\Pr^{sh}$ ($\gamma : \Pr \to \Pr^{sh}$), реализуемых в предлагаемой схеме.

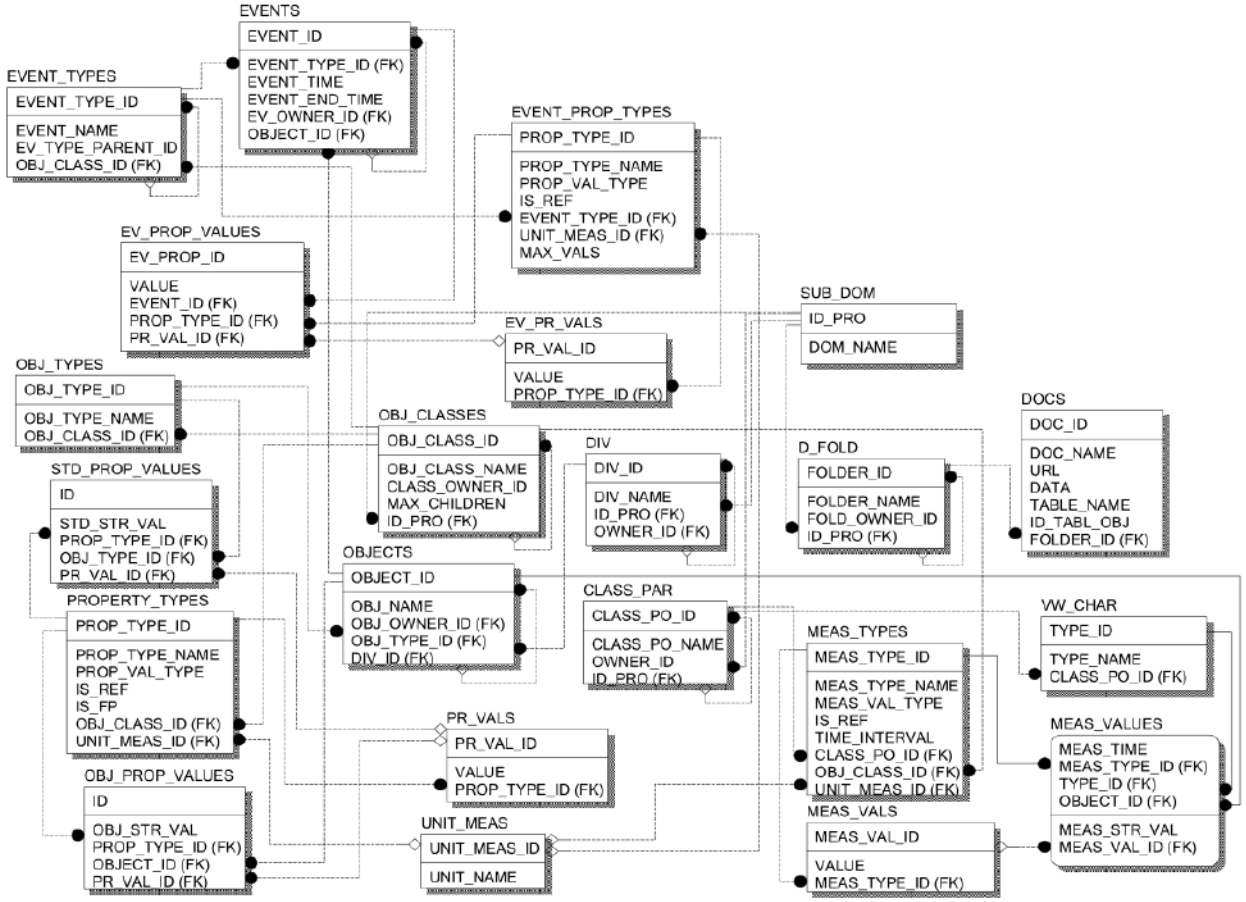


Рис. 4. Диаграмма основных базовых отношений инвариантной к ПрО схемы БД

Декларативная поддержка ограничений целостности заключается в определении ограничений средствами DDL (Data Definition Language) языка SQL. Средства декларативной поддержки целостности использовались при создании базовых отношений инвариантной к ПрО схемы БД для определения таких типов ограничений, как целостность сущностей, ссылочная целостность, обязательные данные (*not null*), ограничения доменов.

Как известно, целостность сущностей связывается в первую очередь с уникальностью и несократимостью первичных ключей [3, 9]. При создании основных базовых отношений схемы, представленных в виде основных строк кода языка DDL (как результат отображения – $\varphi : \Re \to R^{sh}$), подобные требования поддержки целостности данных были определены для всех отношений схемы с помощью конструкций стандарта ISO SQL *primary key, unique* операторов создания (изменения) базовых отношений. Формально этот факт можно представить как результат отображений:

$$\gamma_{PK} : \Pr_{PK} \to \Pr^{sh}_{constr_{\text{primary\_key}}} \; ; \; \gamma_{UK} : \Pr_{UK} \to \Pr^{sh}_{constr_{\text{unique}}} .$$

В результате отображения $\gamma_{FK} : \Pr_{FK} \to \Pr^{sh}_{constr_{\text{foreign\_key}}}$ (применения конструкции *foreign key* операторов *alter (create) table*) для обеспечения ссылочной целостности были определены внешние ключи отношений схемы и стратегии действий при удалении данных.

Для ускорения доступа к данным в соответствии с рекомендациями, приведенными в работах [3, 11, 12], на внешних ключах отношений предлагаемой схемы были определены

дополнительные индексы (с учетом, что на первичных и уникальных ключах СУБД, как правило, автоматически создает уникальный индекс).

В результате отображения $\gamma_{not\_null} : \mathrm{Pr}_{not\_null} \to \mathrm{Pr}^{sh}_{constr_{\mathrm{not\_null}}}$ (применения спецификатора *not null* в операторах *alter (create) table*) были заданы ограничения, запрещающие присваивание соответствующим атрибутам неопределенных значений *null.*

С использованием отображения множества ограничений целостности модели данных с универсальным базисом отношений M_{umd} в инвариантной к ПрО схеме БД были определены ограничения для доменов атрибутов признаков, типов данных характеристик объектов, событий, параметров объектов и некоторые другие (как результат отображения $\gamma_{dom} : \mathrm{Pr}_{dom} \to \mathrm{Pr}^{sh}_{constr_{\mathrm{check}}}$ – применения конструкции *check* оператора *alter table*).

Однако не все ограничения можно реализовать с помощью декларативной поддержки целостности. Поэтому наряду со средствами этого способа реализации ограничений целостности широкое применение нашли средства процедурной поддержки – триггеры, хранимые процедуры, функции, механизмы которых в течение нескольких последних лет во многих коммерческих СУБД были существенно расширены [10].

Необходимость применения в инвариантной к ПрО схеме БД триггеров, хранимых процедур и функций обусловлена многими факторами, в том числе невозможностью реализации средствами декларативной поддержки для некоторых отношений сложных ограничений целостности и ссылочной целостности, проверки корректности вводимых данных, а также потребностью автоматического формирования (генерации) первичных ключей. В связи с этим, следующие ограничения целостности были реализованы с помощью средств процедурной поддержки ($\mathrm{Pr}^{sh}_{constr_{\mathrm{proc}}}$):

– ограничение на возможность изменения занесенных в соответствующие отношения схемы метаданных ПрО (идентификаторов ассоциируемых с определенными базовыми понятиями, рис. 2; чисел максимальных значений в отношении $R^{sh}_{\mathrm{event\_prop\_types}}$ [max_vals], $\mathrm{max\_vals} \in \mathrm{at}(R^{sh}_{\mathrm{event\_prop\_types}})$); удаления списочных значений соответствующих характеристик из отношений $R^{sh}_{pr\_vals}$, $R^{sh}_{ev\_pr\_vals}$, $R^{sh}_{meas\_vals}$, если они присутствуют в отношениях R^{sh}, ассоциируемых с данными моделируемой ПрО, и т. д.);

– ограничение на возможность ввода новых данных, противоречащих введенным метаданным ПрО (для отношений R^{sh}, ассоциируемых с данными ПрО);

– реализация ссылочной целостности для отношений схемы R^{sh}, связанных с отношением R^{sh}_{docs} (конкретный документ из отношения R^{sh}_{docs} связывается с конкретным экземпляром соответствующего отношения R^{sh} (рис. 4));

– ограничение максимального количества экземпляров объектов (из отношения $R^{sh}_{\mathrm{objects}}$) для определенного класса объектов (из $R^{sh}_{\mathrm{obj\_classes}}$);

– ограничение максимального количества значений (из $R^{sh}_{\mathrm{ev\_prop\_values}}$), которые могут быть присвоены определенной характеристике события (из $R^{sh}_{\mathrm{event\_prop\_types}}$) для экземпляра события (из R^{sh}_{events}) заданного класса;

– ограничение на количество событий (из R^{sh}_{events}), происходящих с одним экземпляром объекта (из R^{sh}_{objects}):

а) в один и тот же момент времени с одним экземпляром объекта не может происходить больше одного события;

б) у одного события, происходящего с одним экземпляром объекта, может быть несколько подчиненных событий с разными экземплярами объектов, происходящими в один момент времени, но у конкретного экземпляра события, которое происходит с экземпляром объекта определенного класса, событие-владелец может быть только одно;

– генерация уникальных значений кодов первичных ключей для отношений схемы R^{sh}.

На рис. 5 приведена блок-схема алгоритма использования средств декларативной и процедурной поддержки ограничений целостности при разработке объектов инвариантной к ПрО схемы БД.

Следует заметить, что область применения процедур (в том числе триггерных), функций как объектов инвариантной к предметным областям схемы БД не ограничивается задачами поддержки целостности. Они используются также для решения следующих задач: выдачи предупреждающих сообщений об исключительных ситуациях или напоминающих сообщениях о необходимости выполнения некоторых действий; преобразования данных; обеспечения защищенности данных от нежелательного разглашения (нарушения конфиденциальности), искажения, потери или снижения меры доступности и других.

При решении проблемы защиты БД ИСОУ как важнейшего корпоративного ресурса в процессе создания инвариантной к ПрО схемы базы данных были разработаны специальные средства (в виде реализованных объектов схемы) и правила их использования, обеспечивающие: управление доступом к объектам схемы, защиту данных и сокрытие объектов, восстановление неправильно измененных или утраченных данных. Это отдельная и важная тема, которую целесообразно более подробно рассмотреть в рамках другой статьи.

В соответствии со сформулированными факторами успешного проектирования инвариантной к предметным областям схемы БД, в том числе предусматривающими необходимость учета в максимально возможной степени особенностей требуемой обработки данных, характеристик будущих пользователей, возможностей технических и программных средств, на которых планируется реализация БД ИСОУ, в дополнение к приведенным выше объектам схемы были разработаны и другие ее объекты. В их числе: хранимые процедуры, функции, представления (views), некоторые дополнительные базовые отношения. Таким образом, в состав предлагаемой схемы базы данных вошли следующие объекты (компоненты):

– фиксированный набор базовых отношений R^{sh}, основные из которых приведены на диаграмме (рис. 4);

– виртуальные отношения (представления – views);

– триггеры, хранимые процедуры (в том числе хранимые процедуры интерпретатора специального непроцедурного языка модели данных (ЯМД) [13, 14]), функции, пакеты программ;

– политики безопасности (набор декларативных команд, которые определяют, как и когда следует применять ограничения доступа пользователей к кортежам R^{sh});

– последовательности и т. д.

Диаграмма компонентов инвариантной к ПрО схемы БД с отношениями зависимости между ними в нотации языка UML приведена на рис. 6.

Следует отметить, что при необходимости, например, для решения некоторых задач, данная схема может дополняться новыми объектами. При этом обязательным условием является недопустимость внесения любых изменений в ее ядро (то есть во все выше рассмотренные элементы схемы). В настоящее время данная схема БД реализована на платформах СУБД Oracle и PostgreSQL.

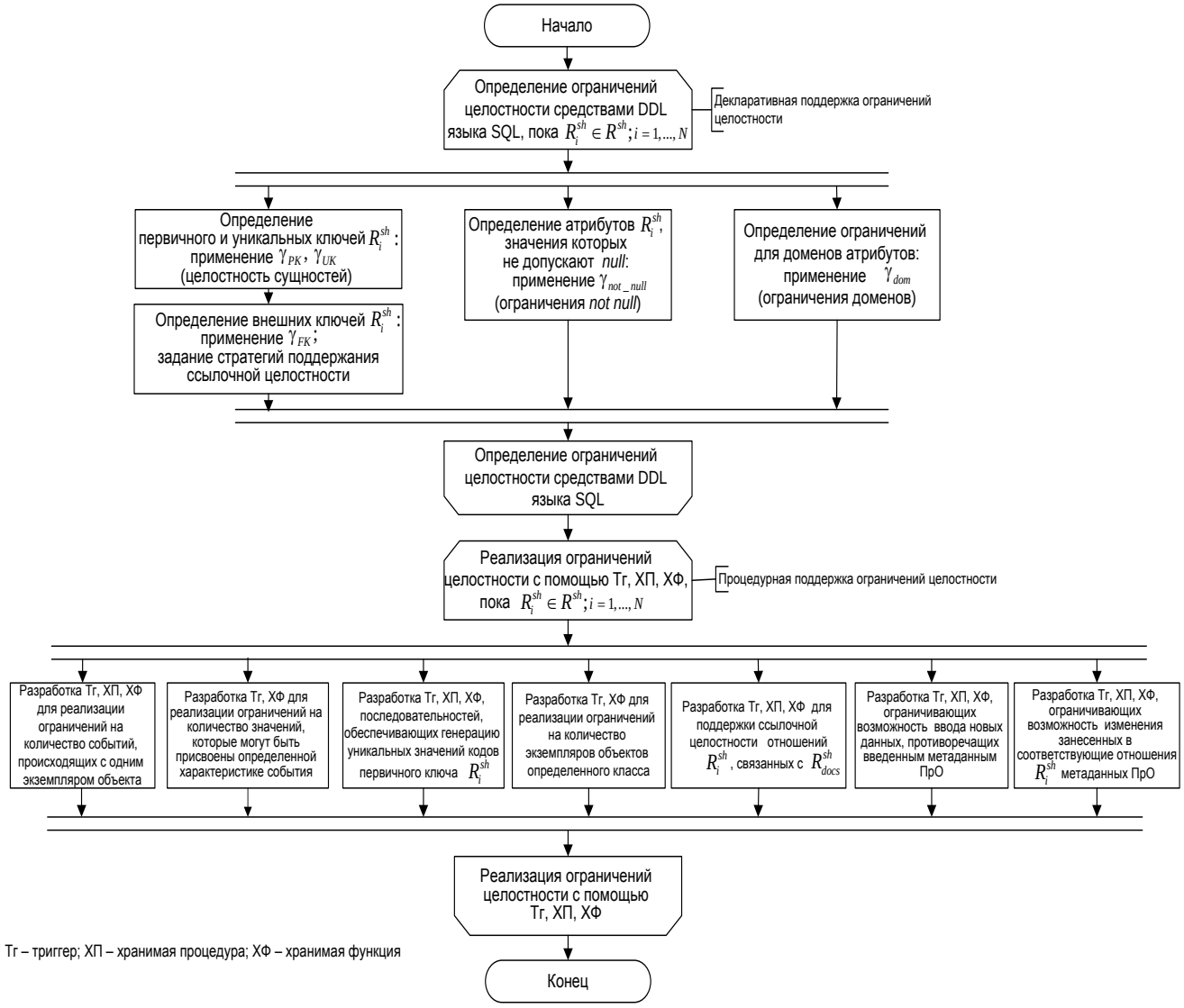


Рис. 5. Блок-схема алгоритма использования средств декларативной и процедурной поддержки ограничений целостности при разработке объектов инвариантной к ПрО схемы БД

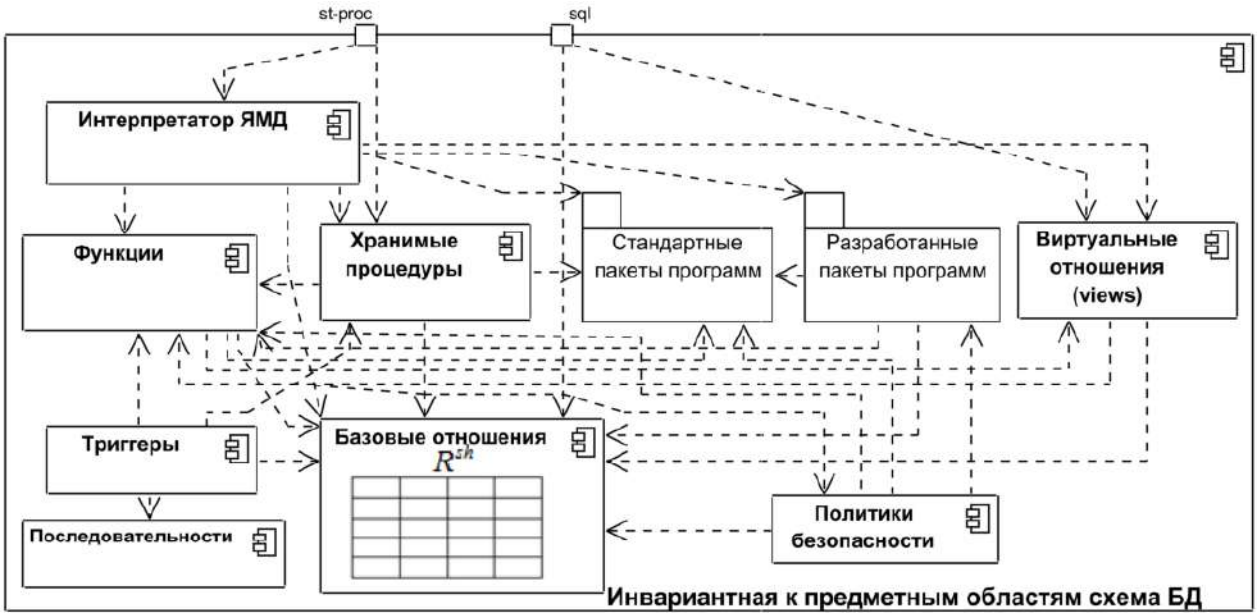


Рис. 6. Диаграмма компонентов инвариантной к ПрО схемы БД

Выводы

1. Число, структура базовых отношений инвариантной к предметным областям схемы БД в отличие от структуры и числа отношений схем, разрабатываемых по традиционной технологии РБД, не зависят от набора данных, они инвариантны к рассматриваемой ПрО.

2. Разработанная инвариантная к предметным областям схема базы данных позволяет упростить процесс адаптации РБД ИСОУ к изменениям условий функционирования. Это достигается за счет использования созданного набора базовых отношений, имеющих принципиальные отличия в назначении, структуре, месте хранения описания метаданных моделируемой ПрО относительно создаваемых базовых отношений при традиционной технологии проектирования реляционных БД. Предлагаемая организация данных обеспечивает совместное хранение метаданных и данных моделируемой ПрО в области изменяемых данных соответствующих базовых отношений схемы БД, в отличие от раздельного хранения метаданных ПрО в словаре СУБД и самих данных собственно в базе данных (в базовых отношениях ее схемы) при традиционном построении РБД. В результате появляется гибкость – возможность описывать ПрО, в том числе и происходящие в ней изменения, на уровне проблемных метаданных, без изменения собственно структуры базовых отношений схемы БД. При расширении набора объектов, событий, характеристик объектов, событий, параметров объектов моделируемой ПрО в БД не создаются новые базовые отношения, атрибуты, ключи или иные объекты схемы, а просто добавляется новая запись в одно из существующих базовых отношений предлагаемой схемы. Более того, в БД ИСОУ, построенных на основе инвариантной к предметным областям схемы базы данных, можно одновременно хранить данные нескольких ПрО.

3. Предлагаемая инвариантная к предметным областям схема БД при необходимости может быть расширена в интересах потребителя информационного продукта, путем дополнения различными новыми объектами: базовыми и виртуальными отношениями, процедурами, функциями и другими. Обязательным условием при этом является недопустимость любых изменений ядра схемы.

Список литературы:

1. Chaos Manifesto 2013: Think Big, Act Small online version. The Standish Group [Electronic resource]. Access mode : http://www.versionone.com/assets/img/files/ChaosManifesto2013.pdf.

2. Standish Group 2015 Chaos Report – Q&A with Jennifer Lynch [Electronic resource]. Access mode : https://www.infoq.com/articles/standish-chaos-2015.

3. Connolly T. M. Database systems: a practical approach to design, implementation, and management. Sixth edition / Thomas M. Connolly, Carolyn E. Begg. Harlow, Essex, England : Pearson Education Limited, 2015. 1329 p.

4. Security and noise immunity of telecommunication systems: new solutions to the codes and signals design problem. Collective monograph. Edited by Sergey G. Rassomakhin, Alexandr A. Kuznetsov. Minden, Nevada, USA : ASC Academic Publishing. 2017. 198 p., Chapter 8, Yesin V.I., Yesina M.V. Means for conceptual modeling of information system databases. P.160-196.

5. Есин В. И. Модель данных «объект-событие»: требования и синтез модели // Computer science and cyber security. International electronic scientific journal. 2017. Issue. 3 (7). P. 33-44. Access mode : http://periodicals.karazin.ua/cscs/article/view/10003/9527.

6. Есин В. И. Выразительные средства модели данных «объект-событие» / В. И. Есин // Радиотехника. 2017. Вып. 191. С. 99-112.

7. Есин В. И. Модель данных с универсальной фиксированной структурой / В. И. Есин // Теоретичні та прикладні аспекти побудови програмних систем : матеріали міжнародної наукової конференції, м. Київ, 15-17 грудня 2014 р. Кіровоград : ФО-П Александрова М. В., 2014. С. 112-116.

8. Есин В. И. Универсальная модель данных и ее математические основы / В. И. Есин // Системи обробки інформації. 2011. № 2(92). С.21-24.

9. Дейт К. Дж. Введение в системы баз данных. 8-е изд. ; пер. с англ. / К. Дейт. – Москва : Изд. дом "Вильямс", 2005. 1328 с.

10. Грофф Д. Р. SQL: полное руководство, 3-е изд. : Пер. с англ. / Грофф Д. Р., Вайнберг П. Н., Оппель Э. Дж. – Москва : ООО "И.Д. Вильямс", 2015. – 960 с.

11. Гарсиа-Молина Г. Системы баз данных // Г. Гарсиа-Молина, Д. Д. Ульман, Д. Уидом. Москва : Изд. дом "Вильямс", 2003. 1088 с.

12. Фиайли К. SQL ; пер. с англ. / К. Фиайли. Москва : ДМК Пресс, 2003. – 456 с.

13. Есин В. И. Язык для универсальной модели данных / В. И. Есин, М. В. Есина // Системи обробки інформації. 2011. № 5(95). С.193-197.

14. Есин В. И. Интерпретатор языка для универсальной модели данных / В. И. Есин, М. В. Есина // Наука і техніка Повітряних Сил Збройних Сил України. 2011. № 2(6). С. 140-143.

Харьковский национальный
университет имени В.Н.Каразина

Поступила в редколлегию 25.02.2018

УДК 681.142

В.А. КРАСНОБАЕВ, д-р техн. наук, А.А. ЗАМУЛА, д-р техн. наук,
А.С. ЯНКО, канд. техн. наук

ПРИМЕРЫ ОПРЕДЕЛЕНИЯ РАНГА ЧИСЛА, ПРЕДСТАВЛЕННОГО В НЕПОЗИЦИОННОЙ СИСТЕМЕ СЧИСЛЕНИЯ ОСТАТОЧНЫХ КЛАССОВ

Введение

Для решения вычислительных задач при использовании системы остаточных классов (СОК) часто возникает необходимость реализации немодульных (позиционных) операций, т.е. таких операций, которые требуют знания величин чисел [1, 2]. К таким операциям, прежде всего, относятся следующие: арифметическое и алгебраическое сравнение чисел, определение знака числа, определение местоположения числа на числовой оси, деление чисел, операции с дробной частью чисел, округление чисел, определение переполнения разрядной сетки, контроль данных в СОК и пр. [3, 4].

Для реализации позиционных операций в СОК используются так называемые позиционные признаки непозиционного кода (ППНК) (позиционные характеристики числа в СОК) [4-6]. В частности, в качестве ППНК может служить ранг r_A числа $A=(a_1, a_2, ..., a_n)$ [1]. В СОК существуют две разновидности ранга числа.

Определение 1. Истинным рангом r_A числа (или просто – рангом r_A числа) A называют натуральное число, показывающее, сколько раз числовой диапазон $M=\prod_{i=1}^{n} m_i$ системы обработки данных (СОД) был превзойден при переходе от представления числа A в СОК к его представлению в ПСС через систему ортогональных базисов B_i.

Определение 2. Ранг числа, являющийся результатом арифметической операции, полученный из рангов чисел называется расчетным рангом числа.

В статье рассматривается только методы определения истинного ранга r_A числа в СОК. Рассмотрим два метода определения ранга числа в СОК [1].

Первый метод. Пусть задана СОК своими основаниями m_i $(i=\overline{1,n})$. Данной СОК однозначно соответствует система ортогональных базисов, B_i $(i=\overline{1,n})$, при которой выполняется равенство

$$A_{ПСС}=\left\{\sum_{i=1}^{n} a_i \cdot B_i\right\} \bmod M \cdot \tag{1}$$

Соотношение (1) можно представить в виде

$$A_{ПСС}=\sum_{i=1}^{n} a_i \cdot B_i - r_A \cdot M \tag{2}$$

Первый метод определения истинного ранга r_A числа основывается на реализации соотношения (2), т.е. требуется осуществить операцию перехода от непозиционного к позиционному представлению числа.

Недостаток первого метода состоит в следующем. При реализации системой обработки данных (СОД) в СОК предполагается, что величина ранга r_A чисел A будет определяться непосредственно в процессе выполнения машинных операций. В этом аспекте рассмотрен-

ный первый метод определения ранга числа по формуле (2) требует выполнения позиционной операции определения величины числа A, что нарушает общую процедуру непозиционной обработки данных СОД (снижается время реализации арифметических операций в СОК).

Основная часть

Второй метод. Использование этого метода позволяет определить ранг r_A числа в СОК без перехода к позиционному представлению числа.

Предварительно, перед описанием метода определения ранга числа в СОК, рассмотрим следующее утверждение. Если в СОК заданы два числа $A=(a_1, a_2, ..., a_n)$ и $B=(b_1, b_2, ..., b_n)$ с соответствующими рангами r_A и r_B, то ранг r_{A+B} суммы двух чисел $A+B$ определится следующим образом:

$$r_{A+B} = r_A + r_B - \sum_{i=1}^{n}\left[\frac{a_i + b_i}{m_i}\right]\cdot \overline{m_i}, \tag{3}$$

где значение $\overline{m_i}$ определяет вес *i*-го ортогонального базиса B_i СОК.

Покажем правильность соотношения (3). Запишем выражения для определения рангов чисел $A_{ПСС}$ и $B_{ПСС}$ в виде (2):

$$A_{ПСС} = \sum_{i=1}^{n} a_i \cdot B_i - r_A \cdot M, \tag{4}$$

$$B_{ПСС} = \sum_{i=1}^{n} b_i \cdot B_i - r_B \cdot M \cdot \tag{5}$$

Сложим два соотношения (4) и (5):

$$A_{ПСС} + B_{ПСС} = \sum_{i=1}^{n} a_i \cdot B_i - r_A \cdot M + \sum_{i=1}^{n} b_i \cdot B_i - r_B \cdot M, \text{ или}$$

$$A_{ПСС} + B_{ПСС} = \sum_{i=1}^{n} (a_i + b_i) \cdot B_i - (r_A + r_B) \cdot M \cdot \tag{6}$$

С другой стороны, на основании правила вычисления суммы двух чисел в СОК для каждого соответствующего основания можно записать

$$A+B = \left\{\left(a_1 + b_1 - \left[\frac{a_1 + b_1}{m_1}\right]\cdot m_1\right), \left(a_2 + b_2 - \left[\frac{a_2 + b_2}{m_2}\right]\cdot m_2\right), ... \right.$$
$$\left. ..., \left(a_n + b_n - \left[\frac{a_n + b_n}{m_n}\right]\cdot m_n\right)\right\}. \tag{7}$$

Выражение (7) можно представить в виде (см. (2)):

$$A+B = \sum_{i=1}^{n}\left(\left\{\left(a_i + b_i\right) - \left[\frac{a_i + b_i}{m_i}\right]\cdot m_i\right\}\right) \times B_i - r_{A+B}\cdot M. \tag{8}$$

Преобразуем выражение (8) с учетом того, что $B_i = \dfrac{\overline{m_i}\cdot M}{m_i}$, и получим

$$A+B=\sum_{i=1}^{n}(a_i+b_i)\cdot B_i-\sum_{i=1}^{n}\left[\frac{a_i+b_i}{m_i}\right]\cdot m_i\cdot B_i-r_{A+B}\cdot M\cdot \tag{9}$$

или
$$A+B=\sum_{i=1}^{n}(a_i+b_i)\cdot B_i-\sum_{i=1}^{n}\left[\frac{a_i+b_i}{m_i}\right]\cdot m_i\cdot\frac{\overline{m_i}\cdot M}{m_i}-r_{A+B}\cdot M\cdot \tag{10}$$

Сравним правые части соотношений (6) и (10) и получим

$$\sum_{i=1}^{n}(a_i+b_i)\cdot B_i-(r_A+r_B)\cdot M=$$

$$=\sum_{i=1}^{n}(a_i+b_i)\cdot B_i-\sum_{i=1}^{n}\left[\frac{a_i+b_i}{m_i}\right]\cdot\overline{m_i}\cdot M-r_{A+B}\cdot M\text{, т.е.}$$

$$r_{A+B}=r_A+r_B-\sum_{i=1}^{n}\left[\frac{a_i+b_i}{m_i}\right]\cdot\overline{m_i}\ . \tag{11}$$

Соотношение (11) является основным аналитическим выражением, позволяющим определить ранг r_{A+B} суммы двух чисел A и B по значениям рангов r_A, r_B слагаемых A и B.

Очевидно, что

$$a_i+b_i\ge m_i\text{, то }\left[\frac{a_i+b_i}{m_i}\right]=1, \tag{12}$$

$$a_i+b_i<m_i\text{, то }\left[\frac{a_i+b_i}{m_i}\right]=0\cdot \tag{13}$$

Последовательность определения ранга r_A числа состоит в следующем.

К исходному числу $A=(a_1,a_2,...,a_n)$ в СОК, ранг r_A которого необходимо определить, последовательно прибавляются константы $t^{(i)}$ $(i=\overline{1,n})$, представленные в СОК, в виде минимальных чисел типа $t^{(i)}=(0,0,...,0,t_i,t_{i+1},...,t_n)$. В этом случае эта величина в ПСС $t^{(\mathrm{i})}_{ПСС}=m_1\cdot m_2\cdot...\cdot m_{i-1}$

В частности, получим

$$t^{(1)}=\min(t_1,t_2,...,t_n)=(1,1,...,1);$$
$$t^{(2)}=\min(0,t_2',...,t_n')=(0,m_1,m_2,...,m_{i-1});$$
$$t^{(3)}=\min(0,0,t_3'',t_4'',...,t_n'')=\{(0,0,m_1\cdot m_2(\mathrm{mod}\,m_3),$$
$$m_1\cdot m_2(\mathrm{mod}\,m_4),...,m_1\cdot m_2(\mathrm{mod}\,m_n)\}$$

и т.д., где $t^{(n)}=(0,0,...,0,t_n)$. В ПСС это значение равно $t^{(\mathrm{n})}_{ПСС}=m_1\cdot m_2\cdot...\cdot m_{n-1}$. Числа $t^{(i)}$ и их ранги r_i определяются основаниями $m_1,m_2,...m_n$ заданной СОК [3].

Покажем, процедуру получения значения $A_n=(0,0,...,0)$. Вначале процедуры к исходному числу A прибавляем константу $t^{(1)}=(t_1,t_2,...,t_n)$ столько раз, сколько потребуется

для того, чтобы выполнялось условие $a_1 = 0$. Пусть для этого потребуется k_1 сложений типа $A + t^{(1)}$. В этом случае получим

$$A_1 = A + k_1 \cdot t^{(1)}.$$

В результате число A_1 имеет ранг r_{A_1}. По формуле (11) получим, что $r_{A_1} = r_A + w_1$, где w_1 – известная величина. Далее производим k_2 раз сложений величины константы $t^{(2)} = (0, t'_2, ..., t'_n)$ с числом A_1 до получения нулевого остатка по основанию m_2, т.е. получим $a_2 = 0$. Имеем число $A_2 = A_1 + k_2 \cdot t^{(2)}$ с рангом $r_{A_2} = r_A + w_2$, где w_2 – известная величина. Алгоритм получения числа $A = (0, 0, ..., 0)$ представлен соотношением

$$\begin{cases} A_1 = A + k_1 \cdot t^{(1)}, \\ \Gamma_{A_1} = \Gamma_A + \omega_1; \\ A_2 = A_1 + k_2 \cdot t^{(2)}, \\ \Gamma_{A_2} = \Gamma_{A_1} + \omega_2; \\ \quad ... \\ A_i = A_{i-1} + k_i \cdot t^{(i)}, \\ \Gamma_{A_i} = \Gamma_{A_{i-1}} + \omega_i; \\ \quad ... \\ A_n = A_{n-1} + k_n \cdot t^{(n)}, \\ \Gamma_{A_n} = \Gamma_{A_{n-1}} + \omega_n. \end{cases} \tag{14}$$

Продолжая процедуру по всем остаткам числа A, получим число $A = (0, 0, ..., 0) = M$. В этом случае значение ω_i $(i = \overline{1, n})$ – это известная величина, которая определяется последовательно в процессе преобразования исходного числа $A = (a_1, a_2, ..., a_n)$ в число $A_n = M = (0, 0, ..., 0)$.

В соответствии с выражением (2) имеем:

$$\begin{aligned} A_{ПСС} &= \sum_{i=1}^{n} a_i \cdot B_i - r_A \cdot M, \\ A_n &= \sum_{i=1}^{n} a_i B_i - r_A \cdot M, \\ (0, 0, ..., 0) &= \sum_{i=1}^{n} a_i B_i - r_A \cdot M, \\ M &= 0 - r_A \cdot M, \\ r_A &= -1. \end{aligned} \tag{15}$$

Таким образом, промежуточный ранг числа $A_n = (0, 0, ..., 0)$ – $r_A = -1$. С другой стороны, показано (14), что ранг равен $r_A + w_n$. В этом случае

$$r_A + w_n = -1, \tag{16}$$

$$r_A = -1 - w_n. \tag{17}$$

Тогда ранг r_A числа $A=(a_1, a_2, ..., a_n)$ в СОК определится в соответствии с формулой (17).

Таким образом, суть второго метода определения ранга числа $A=(a_1, a_2, ..., a_n)$ состоит в следующем. К исходному числу $A=(a_1, a_2, ..., a_n)$ в СОК, ранг r_A которого необходимо определить, последовательно прибавляя константы $t^{(i)}$ $(i=\overline{1, n})$ до тех пор, пока в конечном результате не получим число $A_n=(0, 0, ..., 0)$, промежуточный ранг которого $r_A=-1$. Далее, по формуле (17), определяется истинный ранг числа.

Приведем примеры определения ранга числа A. В табл. 1 представлены основания СОК $\{m_i\}$, $i=\overline{1, 3}$, ортогональные базисы B_i и их веса $\overline{m_i}$. В табл. 2, для заданной СОК, даны минимальные константы $t^{(i)}$ и их ранги $r_{t^{(i)}}$. В этом случае $M=\prod_{i=1}^{3} m_i = 3\cdot 5\cdot 7 = 105$.

Таблица 1

$m_1=3$	$m_2=5$	$m_3=7$
$\overline{m_1}=2$	$\overline{m_2}=1$	$\overline{m_3}=1$
$B_1=70$	$B_2=21$	$B_3=15$

Таблица 2

$t^{(1)}=(1,1,1)$	$t^{(2)}=(0,3,3)$	$t^{(3)}=(0,0,1)$
$r_1=1$	$r_2=1$	$r_3=0$

Ранги минимальных констант $t^{(i)}$ вычисляются заранее по формуле (2). Так, определим значения минимальных констант для СОК, заданной в табл.1:

$t^{(1)}=(1,1,1)=1\cdot B_1+1\cdot B_2+1\cdot B_3=(70+21+15)\bmod 105=106-r\cdot M==106-1\cdot 105.$

В этом случае $r_{t^{(1)}}=1$ (табл. 2).

$t^{(2)}=(0,3,3)=0\cdot B_1+3\cdot B_2+3\cdot B_3=0\cdot 70+3\cdot 21+3\cdot 15=108==3(\bmod 105)=108-r\cdot 105.$

В этом случае $r_{t^{(2)}}=1$ (табл. 2).

$t^{(3)}=(0,0,1)=0\cdot B_1+0\cdot B_2+1\cdot B_3=15=15-r\cdot 105$. В этом случае $r_{t^{(3)}}=0$ (табл. 2).

Пример 1. В соответствии с данными табл. 1, 2 найти ранг r_A числа $A=(2,1,1)=71$.

I этап. Обнуление остатка $a_1=2$ по первому модулю $m_1=3$.

Сложим число A с $t^{(1)}$.

$$A+t^{(1)}=(2,1,1)+(1,1,1)=(0,2,2).$$

Ранг суммы определится по формуле (11)

$$r=(r_A+r_{t^{(1)}})-\sum_{i=1}^{3}\left[\frac{a_i+b_i}{m_i}\right]\cdot\overline{m_i}=$$

$$=(r_A+r_{t^{(1)}})-\left\{\left[\frac{a_1+b_1}{m_1}\right]\cdot\overline{m_1}+\left[\frac{a_2+b_2}{m_2}\right]\cdot\overline{m_2}+\left[\frac{a_3+b_3}{m_3}\right]\cdot\overline{m_3}\right\}=$$

$$= (r_A + 1) - \left\{ \left[\frac{2+1}{3} \right] \cdot 2 + \left[\frac{1+1}{5} \right] \cdot 1 + \left[\frac{1+1}{7} \right] \cdot 1 \right\} =$$
$$= (r_A + 1) - (1 \cdot 2 + 0 \cdot 1 + 0 \cdot 1) = r_A + 1 - 2 = r_A - 1.$$

При этом имел место один переход через первое основании m_1 (формулы (12), (13)).

II этап. Обнуление остатка $a_2 = 2$ по второму модулю $m_2 = 5$ числа $(0, 2, 2)$. Сложим два числа

$$(0, 2, 2) + t^{(2)} = (0, 2, 2) + (0, 3, 3) = (0, 0, 5).$$

Ранг суммы двух чисел определяется следующим образом

$$r = (r_A - 1) + r_{t^{(2)}} - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$
$$= (r_A - 1) + r_{t^{(2)}} - \left\{ \left[\frac{a_1 + b_1}{m_1} \right] \cdot \overline{m_1} + \left[\frac{a_2 + b_2}{m_2} \right] \cdot \overline{m_2} + \left[\frac{a_3 + b_3}{m_3} \right] \cdot \overline{m_3} \right\} =$$
$$= (r_A - 1 + 1 - \left\{ \left[\frac{0+0}{3} \right] \cdot 2 + \left[\frac{2+3}{5} \right] \cdot 1 + \left[\frac{2+3}{7} \right] \cdot 1 \right\} =$$
$$= r_A - (0 \cdot 2 + 1 \cdot 1 + 0 \cdot 1) = r_A - 1.$$

При этом имел место один переход через второе основание m_2.

III этап. Обнуление остатка $a_3 = 5$ числа $(0, 0, 5)$. Сложим два числа

$$(0, 0, 5) + t^{(3)} = (0, 0, 5) + (0, 0, 1) = (0, 0, 6).$$

Ранг суммы двух чисел

$$r = (r_A - 1) + r_{t^{(3)}} - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$
$$= (r_A - 1) + r_{t^{(3)}} - \left\{ \left[\frac{a_1 + b_1}{m_1} \right] \cdot \overline{m_1} + \left[\frac{a_2 + b_2}{m_2} \right] \cdot \overline{m_2} + \left[\frac{a_3 + b_3}{m_3} \right] \cdot \overline{m_3} \right\} =$$
$$= (r_A - 1) + 0 - \left\{ \left[\frac{0+0}{3} \right] \cdot 2 + \left[\frac{0+0}{5} \right] \cdot 1 + \left[\frac{5+1}{7} \right] \cdot 1 \right\} =$$
$$= r_A - 1 + 0 - 0 - 0 - 0 = r_A - 1.$$

Так как остаток $a_3 = 5$ числа $(0, 0, 5)$ не обнулился, то добавим еще раз значение $t^{(3)}$. Сложим два числа

$$(0, 0, 6) + t^{(3)} = (0, 0, 6) + (0, 0, 1) = (0, 0, 0).$$

Ранг суммы двух чисел

$$r = (r_A - 1) + r_{t^{(3)}} - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$
$$= (r_A - 1) + r_{t^{(3)}} - \left\{ \left[\frac{a_1 + b_1}{m_1} \right] \cdot \overline{m_1} + \left[\frac{a_2 + b_2}{m_2} \right] \cdot \overline{m_2} + \left[\frac{a_3 + b_3}{m_3} \right] \cdot \overline{m_3} \right\} =$$
$$= r_A - 1 + 0 + 0 + 0 + 1 \cdot 1 = r_A - 2.$$

При этом имел место один переход через третье m_3 основание СОК. В соответствии с (13) и (14) имеем

$$r_A - 2 = -1, \text{ или } r_A = 1.$$

Проверка (см.(2)).

$$A = (2,1,1) = 2 \cdot B_1 + 1 \cdot B_2 + 1 \cdot B_3 = 2 \cdot 70 + 1 \cdot 21 + 1 \cdot 15 = 176 - r_A \cdot M =$$
$$= 176 - 1 \cdot 105 = 176 - 105 = 71.$$

Пример 2. В соответствии с исходными данными (табл. 1, 2) найти ранг r_A числа $A = (1,1,5) = 61$.

I этап. Обнулим остаток $a_1 = 1$ по первому модулю m_1. Сложим два числа (табл. 2):

$$A + t^{(1)} = (1,1,5) + (1,1,1) = (2,2,6).$$

Ранг суммы определится по формуле (11)

$$r = (r_A + r_{t^{(1)}}) - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$

$$= (r_A + r_{t^{(1)}}) - \left\{ \left[\frac{a_1 + b_1}{m_1} \right] \cdot \overline{m_1} + \left[\frac{a_2 + b_2}{m_2} \right] \cdot \overline{m_2} + \left[\frac{a_3 + b_3}{m_3} \right] \cdot \overline{m_3} \right\} =$$

$$= (r_A + 1) - \left\{ \left[\frac{1+1}{3} \right] \cdot 2 + \left[\frac{1+1}{5} \right] \cdot 1 + \left[\frac{5+1}{7} \right] \cdot 1 \right\} =$$

$$= (r_A + 1) - 0 \cdot 2 - 0 \cdot 1 + 0 \cdot 1 = r_A + 1.$$

Так как остаток $a_1 = 1$ числа $A = (1,1,5)$ не обнулился, то добавим еще раз значение контакта $t^{(1)}$. Сложим два числа

$$(2,2,6) + t^{(1)} = (2,2,6) + (1,1,1) = (0,3,0).$$

Ранг суммы двух этих чисел

$$r = r_A + 1 + r_{t^{(1)}} - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$

$$= r_A + 1 + 1 - \left\{ \left[\frac{a_1 + b_1}{m_1} \right] \cdot \overline{m_1} + \left[\frac{a_2 + b_2}{m_2} \right] \cdot \overline{m_2} + \left[\frac{a_3 + b_3}{m_3} \right] \cdot \overline{m_3} \right\} =$$

$$= r_A + 1 + 1 - 1 \cdot 2 - 0 \cdot 1 - 1 \cdot 1 = (r_A + 1) + 1 - 2 - 1 = r_A - 1.$$

При этом имели место два переходаи – через первое основание m_1 и через третье основание m_3.

II этап. Обнуление остатка $a_2 = 3$ числа $(0,3,0)$. Сложим два числа

$$(0,3,0) + t^{(2)} = (0,3,0) + (0,3,3) = (0,1,3).$$

Ранг суммы двух чисел определяется так

$$r = (r_A - 1) + r_{t^{(2)}} - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$

$$= (r_A - 1) + r_{t^{(2)}} - \left\{ \left[\frac{a_1 + b_1}{m_1} \right] \cdot \overline{m_1} + \left[\frac{a_2 + b_2}{m_2} \right] \cdot \overline{m_2} + \left[\frac{a_3 + b_3}{m_3} \right] \cdot \overline{m_3} \right\} =$$

$$= (r_A - 1) + 1 - \left\{ \left[\frac{0+0}{3} \right] \cdot 2 + \left[\frac{3+3}{5} \right] \cdot 1 + \left[\frac{0+3}{7} \right] \cdot 1 \right\} =$$

$$= r_A - 1 + 1 - 0 \cdot 2 - 1 \cdot 1 + 0 \cdot 1 = r_A - 1.$$

Имел место один переход через основание m_2.

Так, как остаток $a_2 = 3$ числа $(0, 3, 0)$ не обнулился, то сложим два числа:

$$(0, 1, 3) + t^{(2)} = (0, 1, 3) + (0, 3, 3) = (0, 4, 6).$$

Ранг суммы двух чисел

$$r = (r_A - 1) + r_{t^{(2)}} - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$

$$= (r_A - 1) + 1 - \left\{ \left[\frac{0+0}{3} \right] \cdot 2 + \left[\frac{1+3}{5} \right] \cdot 1 + \left[\frac{3+3}{7} \right] \cdot 1 \right\} =$$

$$= r_A - 1 + 1 - (0 \cdot 2 + 0 \cdot 1 + 0 \cdot 1) = r_A.$$

Переходов (переполнений) по остаткам (основаниям) нет. Так как остаток $a_2 = 3$ числа $(0, 3, 0)$ не обнулился, то вновь сложим два числа

$$(0, 4, 6) + t^{(2)} = (0, 4, 6) + (0, 3, 3) = (0, 2, 2).$$

Ранг суммы двух чисел

$$r = r_A + r_{t^{(2)}} - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$

$$= r_A + 1 - \left\{ \left[\frac{0+0}{3} \right] \cdot 2 + \left[\frac{4+3}{5} \right] \cdot 1 + \left[\frac{6+3}{7} \right] \cdot 1 \right\} =$$

$$= r_A + 1 - (0 \cdot 2 + 1 \cdot 1 + 1 \cdot 1) = r_A + 1 - 2 = r_A - 1.$$

Имеют место два перехода (переполнения) через основания m_2 и m_3.

Так, как остаток a_2 не обнулился, то реализуется операция сложения двух чисел:

$$(0, 2, 2) + t^{(2)} = (0, 2, 2) + (0, 3, 3) = (0, 0, 5).$$

Ранг суммы двух чисел

$$r = (r_A - 1) + r_{t^{(2)}} - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$

$$= (r_A - 1) + 1 - \left\{ \left[\frac{0+0}{3} \right] \cdot 2 + \left[\frac{2+3}{5} \right] \cdot 1 + \left[\frac{2+3}{7} \right] \cdot 1 \right\} =$$

$$= r_A - 1 + 1 - 0 \cdot 2 - 1 \cdot 1 - 0 \cdot 1 = r_A - 1 + 1 - 1 = r_A - 1.$$

Имело место одно переполнение по основанию m_2.

III этап. Обнуление остатка $a_3 = 5$ числа $(0, 0, 5)$. Сложим два числа

$$(0, 0, 5) + t^{(3)} = (0, 0, 5) + (0, 0, 1) = (0, 0, 6).$$

Ранг суммы двух чисел

$$r = (r_A - 1) + 0 - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$

$$= r_A - 1 + 0 - \left(\left[\frac{0+0}{3} \right] \cdot 2 + \left[\frac{0+0}{5} \right] \cdot 1 + \left[\frac{5+1}{7} \right] \cdot 1 \right) =$$

$$= r_A - 1 + 0 - 0 \cdot 2 - 0 \cdot 1 - 0 \cdot 1 = r_A - 1.$$

Переполнений основ не было. Так как остаток a_3 не обнулен, то реализуется операция сложения двух чисел:

$$(0, 0, 6) + t^{(3)} = (0, 0, 6) + (0, 0, 1) = (0, 0, 0).$$

Ранг суммы определится следующим образом:

$$r = (r_A - 1) + 0 - \sum_{i=1}^{3} \left[\frac{a_i + b_i}{m_i} \right] \cdot \overline{m_i} =$$

$$= r_A - 1 + 0 - \left(\left[\frac{0+0}{3} \right] \cdot 2 + \left[\frac{0+0}{5} \right] \cdot 1 + \left[\frac{6+1}{7} \right] \cdot 1 \right) =$$

$$= r_A - 1 + 0 - 0 \cdot 2 - 0 \cdot 1 - 1 \cdot 1 = r_A - 2.$$

Имеется одно переполнение в остатке a_3 по основанию m_3. В соответствии с (15) и (16)

$$r_A - 2 = -1,\ r_A = 1.$$

Проверка (см. (2)). $A = (1, 1, 5)$. В ПСС имеем, что

$$A = a_1 \cdot B_1 + a_2 \cdot B_2 + a_3 \cdot B_3 = (1 \cdot 70 + 1 \cdot 21 + 5 \cdot 15) \bmod 105 =$$

$$= 166 - r_A \cdot 105 = 166 - 1 \cdot 105 = 61.$$

Выводы

Рассмотрены два метода определения ППНК СОК – ранга числа. Основное внимание уделено примерам реализации второго метода, преимуществом которого является то, что определение ранга числа можно проводить в динамике вычислительного процесса, т.е. без останова вычислений на время выполнения непозиционных операций перевода чисел из СОК в ПСС и обратно. Это дает возможность в полной мере использовать основное свойство СОК – высокое быстродействие выполнения арифметических операций. Сокращается количество оборудования, необходимого для реализации позиционных операций, что особенно важно для бортовых вычислителей баллистических ракет и космических аппаратов.

Список литературы:

1. Kuznetsov O., Gorbenko Y., Kolovanova I. Combinatorial properties of block symmetric ciphers key schedule // 2016 Third International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T), Kharkiv, 2016. pp. 55-58. DOI: 10.1109/INFOCOMMST.2016.7905334.

2. Sergey G. Rassomakhin. Mathematical and Physical Nature of Channel Capacity // ISCI'2017: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan D. Gorbenko and Alexandr A. Kuznetsov. ASC Academic Publishing, USA, 2017. 207 p.

3. Акушский И. Я., Юдицкий Д. И. Машинная арифметика в остаточных классах. Москва : Сов. радио, 1968. 440 с.

4. Торгашов В. А. Система остаточных классов и надежность ЦВМ. Москва : Сов. радио, 1973. 118 с.

5. Krasnobayev V. A., Koshman S. A., Mavrina M. A. A method for increasing the reliability of verification of data represented in a residue number system // Cybernetics and Systems Analysis. November 2014. Vol. 50, Issue 6. pp 969-976.

6. Краснобаев В.А., Кошман С. А., Маврина М. А. Метод исправления однократных ошибок данных, представленных кодом класса вычетов // Электрон. моделирование. 2013. Т. 35. № 5. С. 43–56.

Харьковский национальный
университет имени В.Н.Каразина

Поступила в редколлегию 07.03.2018

УДК 681.3.06:519.248.681

О.А. ЗАМУЛА, д-р техн. наук

ТЕХНОЛОГІЇ ФОРМУВАННЯ OFDM СИГНАЛІВ В СУЧАСНИХ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМАХ

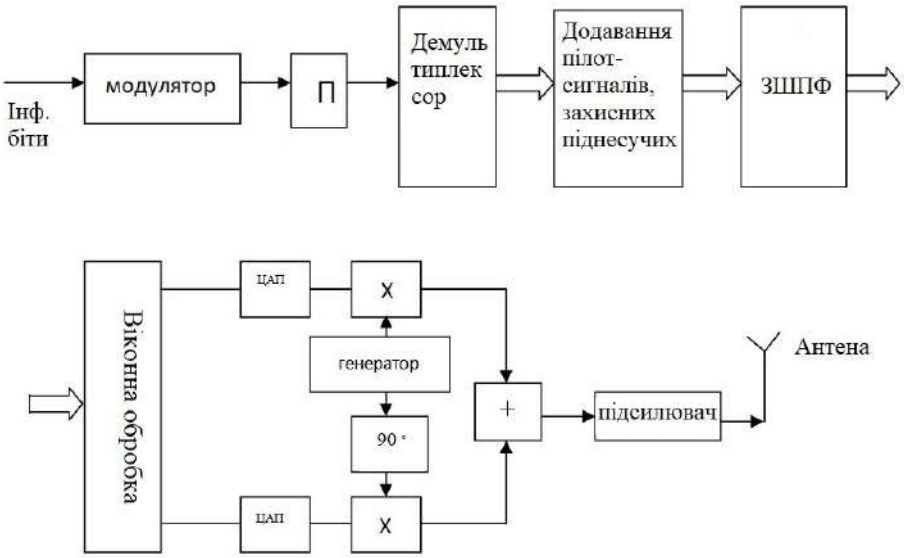
Вступ

Сучасні бездротові системи (наприклад, супутникові системи, системи мобільного телефонного зв'язку) відносяться до багатокористувачевих систем. При проектуванні таких систем основною проблемою є вибір способу множинного доступу, тобто можливості одночасного використання багатьма абонентами каналу зв'язку з мінімальним взаємним впливом [1, 2]. Широкосмугові сигнали активно використовуються в сучасних високошвидкісних системах стільникового зв'язку стандартів WiMax, Mobile WiMax, MBWA, бездротових дискретних комунікаційних системах, наприклад LTE і Wi-Fi, при передачі інформації цифрового телебачення (DVB-T) і радіо (DRM, DAB), в системах радіолокації тощо. Використання сигналів з ортогональним частотним поділом каналів і мультиплексуванням (Orthogonal frequency-divisionmultiplexing, далі - OFDM), в тому числі в зазначених системах передачі інформації дозволяє підвищити не тільки інформаційну ємність системи в умовах багатопроменевого поширення при обмеженій смузі пропускання, але і швидкість прийому-передачі даних, наблизивши її до пропускної здатності каналу, збільшити скритність передачі і завадостійкість системи. В даний час йде бурхливий розвиток, дослідження та стандартизація технологій для п'ятого покоління мереж стільникового зв'язку - 5G. Найбільш пріоритетними завданнями даного напрямку вважаються: досягнення максимальної швидкості передачі даних (до 20 Гбіт/с); забезпечення щільності призначених для користувача пристроїв (до 106 пристроїв / км^2); надання користувачам сервісів надійної комунікації з малою затримкою (URLLC) (затримка передачі даних не більше 1 ms) [3 - 4]. Як можливі рішення для досягнення зазначених завдань для 5G мереж розглядаються: використання спектра в міліметровому діапазоні [5]; нові види модуляції сигналів і методи кодування; методи множинного доступу; вдосконалені технології побудови архітектури антен і мереж [5 - 6]. Крім цього, заслуговують на увагу дослідження, які присвячені: ортогональному мультиплексуванню з частотним поділом каналів з фільтруванням (F-OFDM) [7 - 9]; технологіям просторового рознесення (MIMO) [10]; хмарним мережам радіозв'язку (CRAN) [11], технологіям ортогонального частотного поділу каналів з кодуванням (C-OFDM) [12] і багато інших.

Принципи технології OFDM

Розвиток технологій бездротових комунікацій постійно формувався на основі досліджень форм сигналів. Як приклад можна привести успіх четвертого покоління (4G) зв'язку, який базується, в тому числі, на використанні схеми цифрової модуляції OFDM. Основна ідея OFDM полягає в тому, що для досягнення високої швидкості передачі в частотній області застосовується розподіл повного діапазону частот сигналу на деяке число частотних підканалів, що не перекриваються з меншими швидкостями. При цьому кожен підканал (піднесуча) модулюється окремим символом, потім ці канали мультиплексуються по частоті і далі дані передаються паралельно по ортогональних підканалах. У порівнянні з передачею з однієї несучої цей підхід забезпечує підвищену стійкість до вузькосмугової інтерференції і спотворень в каналі. Більш того, з цього випливає високий рівень гнучкості системи, так як параметри модуляції, такі як розмір сузір'я, швидкість кодування, можуть бути незалежно вибрані для кожного підканала. Структуру модему OFDM складають передавач та прийомний пристрій. У процесі передавання вихідний послідовний потік інформаційних бітів (рисунок) кодується завадостійким кодом (згідно з рекомендацією LTE 3GPP TS 36.211 використ-

товується згортаючи турбокоди з базовою швидкістю 1/3), перемежовується (П) і демультиплексуються на N паралельних підпотоків.



Структурна схема OFDM приймача

Далі кожен з потоків відображається в потік символів за допомогою процедури фазового (BPSK, QPSK, 8-PSK) або амплітудно-фазової квадратурної модуляції (QAM). При використанні модуляції BPSK формується потік двійкових чисел (1 і -1), при QPSK, 8-PSK, QAM - потік комплексних чисел. Крім тих піднесучих, на яких передається інформація формують службові піднесучі. До останніх відносяться захисні інтервали, пілот-сигнали і додаткова службова інформація для синхронізації приймача і передавача, і режимів їх роботи. Пілот-сигнали можуть мати фіксоване положення на піднесучих, або змінне - від символу до символу OFDM в кадрах. При цьому завдяки вставці між суміжними підканалами достатнього (за тривалістю) захисного інтервалу виключається спектральне перекриття. В цьому випадку знижується міжканальна інтерференція (міжбітова інтерференція, ICI), зменшується ймовірність бітової помилки, а значить, підвищується пропускна здатність системи бездротового доступу.

Операція множення на комплексну експоненту з відповідною частотою підканалу і потім складання усіх підканалів для формування OFDM сигналу дуже схожа з операцією зворотного перетворення Фур'є. У зв'язку з цим для формування необхідного OFDM-символу застосовують апарат швидкого зворотного перетворення Фур'є (ЗШПФ), що значно спрощує реалізацію модуляторів.

Збереження ортогональності є необхідним для того, щоб приймач міг правильно розпізнати інформацію на піднесучих. Для цього необхідно виконати наступні умови:

- приймач і передавач повинні бути точно синхронізовані;
- аналогові компоненти передавача і приймача повинні бути дуже високої якості;
- канал не повинен бути багатопроміневим (багатошляховим).

На жаль, багатопроміневе спотворення практично неминуче в системах радіозв'язку, що призводить до помилок при прийомі сигналів. Для усунення такого роду перешкод необхідно вибрати захисний інтервал, тривалість якого більше, ніж максимальна затримка поширення в каналі. Таким чином, можна усунути більшість видів інтерференції між каналами (тобто інтерференцію між поднесучими (ICI)) і між суміжними блоками передачі (тобто міжсимвольну інтерференцію (ISI)). Для зменшення позаполосного випромінювання сигналів використовується віконна обробка сигналу, вікном типу «піднятий косинус».

Далі цифро-аналогові перетворювачі (ЦАП) перетворюють в аналоговий вигляд окремо дійсну і уявну компоненти. Після проходження через фільтр нижніх частот сигнал надходить на квадратурний змішувач, який переносить корисний спектр OFDM-сигналу на несучу частоту. Ці сигнали далі складаються, посилюються і формується власне сигнал ОФДМ.

Широке використання цифрової схеми модуляції OFDM обумовлено цілим рядом чудових властивостей даної технології:

- стійкість до наслідків багатопроменевого поширення;
- висока стійкість до вузькосмугових перешкод;
- стійкість до міжсимвольної інтерференції за рахунок того, що тривалість символу в допоміжній піднесучій значно більше в порівнянні з затримкою поширення, ніж в традиційних схемах модуляції;
- висока спектральна ефективність в порівнянні з традиційними системами з частотним поділом каналів за рахунок великої кількості піднесучих;
- можливість використання різних схем модуляції для різних піднесучих, що дозволяє адаптуватися до умов поширення сигналу і до різних вимог відносно якості прийнятих сигналів;
- проста реалізація із застосуванням методів цифрової обробки і ін.

Перспективні технології формування сигналів в сучасних мобільних системах телекомунікацій

Ефективність сучасного покоління мобільного зв'язку значною мірою ґрунтується на використанні OFDM модуляції. Однак для подальшого прогресу і переходу на більш досконалі технології п'ятого покоління зв'язку необхідно переглянути технології OFDM, які використовуються, так само як і досліджувати інші технології. Можна виділити наступні основні відмінності технології 5G від технологій мобільної комунікації попереднього покоління [4 - 5].

1. Однією з цілей 5G є забезпечити використання різних сервісів, зокрема eMBB, mMTC і URLLC. Передбачається, що технологія 5G повинна підтримувати більш гнучке використання доступної смуги частот для збільшення пропускної здатності. Для цього необхідно розробити і впровадити різні варіанти використання доступних частотних і часових ресурсів для різних послуг.

2. Зростання пропускної здатності. Передбачається триразове зростання ефективності використання спектру сигналу в 5G в порівнянні з сервісами eMBB [3]. Для підвищення пропускної здатності в мережах 5G передбачається зменшити захисні інтервали [4].

3. При асинхронній передачі даних в мережах 4G базова станція постійно синхронізується з призначеним для користувача обладнанням для зменшення взаємних перешкод між несучими (inter-carrier interference - ICI) [4]. Втрати, спричинені такими перешкодами, негативно позначаються на сервісах, зокрема mMTC, які пов'язані з масовим підключенням абонентів мережі. Таким чином, підтримка мережами 5G при асинхронній передачі необхідна з метою вирішення проблем, які пов'язані з ICI і забезпеченням роботи при множинних підключеннях [4].

Як зазначалося вище, ортогональне частотне ущільнення (OFDM) це схема доступу, яка використовується в сучасних мережах 4G. Для отримання доступу до мережі використовуються два окремих сигнали: сигнал доступу з ортогональним частотним ущільненням (OFDMA) в низхідному каналі і сигнал множинного доступу з частотним ущільненням і однією несучою (SC-FDMA) у висхідному каналі. Переваги даної схеми пов'язані з можливістю передачі сигналів на безлічі несучих. При цьому дана схема (OFDM) має ряд недоліків, зокрема: висока чутливість до зсувів тактової частоти; високе відношення пікового рівня потужності сигналу до середнього (пік фактор (PAPR)); використання захисних інтервалів знижує спектральну ефективність; метод чутливий до ефекту Доплера, що накладає деякі обмеження на його застосування в мобільних мережах; перекриття смуг піднесучих призводить до появи

міжбітової інтерференції; сигнал OFDM вразливий для спектральних продуктів перетворень, викликаних нелінійними підсилювачами, зміщенням постійної складової при використанні швидкого перетворення Фур'є. Крім того, чутливість до зсувів тактової частоти робить необхідним періодичне додавання сигналів синхронізації в загальний обсяг використовуваних сигналів і вимагає синхронізації пристрою і мережі перед початком зв'язку (обміну даними). Відсутність безперервності (фазовий перехід) між двома символами під час генерації символів OFDM ініціює спектральні скачки в частотної області, що призводить до інтенсивних позасмугових випромінювань і інше.

Обмежені можливості сигналів на основі схеми OFDM модуляції стали передумовою для досліджень з метою вибору кандидатів сигналів для наступних поколінь мобільного зв'язку, зокрема 5G. У зв'язку з цим, одним із завдань, що підлягають вирішенню, є виконання вимоги значного зменшення затримки при введенні нових служб і додатків. Поряд з цим, виникає потреба у формуванні циклічного префіксу і зменшенні тривалості символів. Ці міркування привели до створення цілого ряду технологій формування сигналів: з узагальненим частотним ущільненням (GFDM); з декількома несучими на базі набору фільтрів (FBMC); OFDM з тимчасовим поділом (w-OFDM); універсальний сигнал, що фільтрується з декількома несучими (UFMC); ортогональне мультиплексування з частотним поділом каналів з фільтруванням F-OFDM і іншим. Проводяться також дослідження нових схем множинного доступу, в тому числі: множинного доступу з розрідженим кодом (SCMA), неортогонального множинного доступу (NOMA) і множинного доступу з розподілом ресурсів (RSMA).

Технологія UFMC [13] рекомендована для подолання проблеми інтерференції (ICI) при множинному доступі користувачів в режимі асинхронної передачі і заснована на частотному поділу і мультиплексуванню за допомогою застосування операції фільтрації групи піднесучих. UFMC є узагальненою версією техніки фільтрування безлічі бічних смуг (БС). Бічні смуги обробляються фільтром одночасно, замість обробки кожної БС окремо. Таким чином, зменшуються взаємні перешкоди для БС в порівнянні з традиційним OFDM. Також, застосування операцій фільтрації БС націлене на збільшення ефективності ряду додатків комунікацій, таких як системи з малою затримкою пакетів. Даний вид модуляції виявляється кращим для подібних додатків по відношенню до схеми модуляції FBMC.

FBMC є одним з найбільш відомих форматів модуляції з розширенням спектру в бездротових комунікаціях [14]. Даний вид модуляції забезпечує значну перевагу у формуванні кожної піднесучої і полегшує гнучке використання спектрального ресурсу, дозволяє задовольнити різним системним вимогам, таким як низька затримка, множинний доступ і інші, що призводить до поліпшення показників завадозахищеності системи в умовах розсіювання сигналу у часовій і частотній областях [15]. Для прикладу, прямокутні фільтри більш кращі для каналів, які розподілені у часі, в той час як фільтр з характеристикою типу «піднятий косинус» більш стійкий проти частотного розсіювання. Незважаючи на всі вигоди від використання FBMC, значна довжина фільтрів призводить до великої тривалості символу, що є проблемою не тільки для додатків, до яких висуваються вимоги малої затримки та/або великої кількості користувачів в комунікаціях, але також призводить до збільшення обчислювальної складності для технології MIMO детектування, що, з зрештою, призведе до проблем в роботі всіх основних додатків 5G.

GFDM є блокової схемою модуляції з частотним ущільненням каналів, яка розроблена для роботи з різноманітними додатками 5G, забезпечуючи змінну форму сигналу [16]. Для поліпшення показників надійності і затримки в комунікаціях без корекції помилок, можна використовувати GFDM сигнали разом з перетворенням Уолша - Адамара. При комбінуванні GFDM з квадратурною амплітудною модуляцією в системах з множинним доступом вирішується проблема внутрішньосистемних перешкод за умови використання неортогональних фільтрів. З іншої точки зору, можна розглядати GFDM як схеми з гнучким настроюванням окремих блоків, а не тільки лише однієї несучої в цілому. При маніпуляції відповідних параметрів сигналу GFDM можливе отримання різних форм сигналу таких як OFDM, частотне

вирівнювання з єдиною несучою (SC-FDE) і ін. Незважаючи на досить перспективні можливості, які відкриваються завдяки застосуванню сигналів з GFDM, даний вид модуляції є обчислювально складним [16].

F-OFDM застосовується у каналах низхідної лінії зв'язку 4G технології. Для F-OFDM сконфігурований фільтр застосовується до символу OFDM в часовій області для зниження рівня позасмугового випромінювання сигналу, зберігаючи ортогональність комплексних доменів OFDM-символів. Оскільки смуга пропускання фільтра відповідає смузі пропускання сигналу, зачіпаються тільки кілька піднесучих, близьких до краю. Основне міркування полягає в тому, що довжина фільтру може перевищувати довжину циклічного префіксу для F-OFDM [6]. При цьому знижується рівень міжсимвольної інтерференції, що обумовлено обраною конструкцією фільтра з використанням віконної обробки (з м'яким урізанням). Генерація F-OFDM сигналу заснована на формуванні блоку з M прилеглих БС в ряді послідовних OFDM символів [17]. Зокрема, під час обробки кожного символу, в передавачі формуються параметри: значення розмірності зворотного швидкого перетворення Фур'є (ОБПФ) (N), тривалість M «інформаційних символів» разом з циклічним префіксом, де N > M. Інформаційні символи можуть бути точками сузір'їв (constellation points) як в OFDM. Вказане можна представити в наступному вигляді:

$$s(n) = \sum_{l=0}^{L-1} s_l(n - l(N + Ng)) \tag{1}$$

і

$$S_l(n) \equiv \sum_{m=m`}^{m`+M-1} d_{l,m} e^{j2\pi mn/N}, -N_g \le n < N, \tag{2}$$

де Ng – довжина циклічного префіксу (CP), d – інформаційний символ піднесучої m OFDM системи, L означає кількість OFDM символів, а $\{m_0, m_{0+1}, \cdots, m_{0+M-1}\}$ – обраний набір піднесучих. Сигнал F-OFDM формується при обробці сигналу $s(n)$ за допомогою відповідного фільтру, тобто

$$\tilde{s}(n) = s(n) * f(n). \tag{3}$$

Пропускна здатність фільтру дорівнює сумі пропускної здатності обраних БС, а часові витрати - це тривалість символу OFDM. У прийомному пристрої отриманий сигнал спочатку проходить через фільтр $f(-n)$, який ідентичний фільтру передавача. Прийнятий сигнал обробляється з використанням стандартних OFDM перетворень, а потім відфільтрований сигнал розділяється на послідовність окремих OFDM символів з видаленням циклічного префіксу. При цьому до кожного символу застосовується БПФ розмірності N і далі виділяють інформаційні символи з відповідних піднесучих.

Фільтр для F-OFDM повинен відповідати таким критеріям: мати плоску смугу пропускання; мати гостру перехідну смугу для мінімізації захисних смуг. Даним критеріям відповідають фільтри з прямокутним частотним відгуком. Щоб задовольняти зазначеним вимогам, фільтр нижніх частот реалізується за допомогою «вікна», яке ефективно обрізає імпульсну характеристику і забезпечує плавні переходи до нуля на обох кінцях [18] Таким чином, реалізація F-OFDM привносить додатково до існуючої процедури обробки CP-OFDM етап фільтрації як на стороні передачі, так і на стороні прийому.

Технологія W-OFDM. Для зменшення позасмугового випромінювання сигналів використовується віконна обробка сигналу в часовій області, вікном типу «піднятий косинус». Відомо, що спектр OFDM сигналу має безліч бічних пелюсток, які повільно загасають в частотній області, що призводить до збільшення позасмугового випромінювання. Для зниження позасмугового випромінювання OFDM символу використовують захисні поднесучи, які додають по краях OFDM сигналу. З цією ж метою застосовується віконна обробка сигналу. Така обробка сигналу дозволяє здійснювати плавний перехід між закінченням попереднього і початком наступного символу. Такий перехід здійснюється за допомогою перекриття в часі префі-

ксу поточного символу і суфіксом попереднього символу за допомогою їх підсумовування. Вікно «піднятий косинус» має вигляд

$$h(t)=\begin{cases} 1{,}0 \le |t| \le \dfrac{T(1-\beta)}{2}; \\ \dfrac{1}{2}\left(1+\cos\left[\dfrac{\pi}{\beta+T}\left(|t|-\dfrac{T(1-\beta)}{2}\right)\right]\right), \dfrac{T(1-\beta)}{2} \le |t| \le \dfrac{T(1+\beta)}{2}; \\ 0, \end{cases} \quad (4)$$

де T – тривалість символу, β – спад, який приймає значення в інтервалі від 0 до 1.

Для даної технології важливим є вибір тривалості вікна спаду. Значення тривалості вікна піднесеного косинуса необхідно вибирати рівним або меншим тривалості циклічного префіксу. У цьому випадку застосування віконної обробки для формування символів OFDM дозволяє значно знизити позасмугове випромінювання. На рівень позасмугового випромінювання також впливає вибір захисного інтервалу між піднесучими. Дослідження показали, що чим довше захисний інтервал, тим менше рівень позасмугового випромінювання [19].

Висновки

Представлено технології формування сигналів, які вже використовуються в системах зв'язку і телекомунікацій, а також наведено аналіз перспективних технологій, які можливо знайдуть застосування в різних створюваних системах, в тому числі бездротових системах зв'язку широкосмугового доступу. Показано, що схема модуляції OFDM, яка широко використовується, має ряд недоліків, які можуть призвести до зниження показників ефективності систем, в яких вони застосовуються, зокрема: зниження завадостійкості прийому сигналів, в слідстві спотворень, які викликані багатопроміневістю при поширенні електромагнітного поля між базовою і мобільною станціями, а також впливу міжсимвольних і міжканальних перешкод; нераціональне, в порівнянні з послідовними формами сигналів, використання потужності передавача, що пов'язано з використанням захисного інтервалу для захисту від межсимвольної інтерференції і високим пік-фактором сигналу та ін. Представлено альтернативні технології формування сигналів, зокрема технологія формування сигналів, яка заснована на віконній обробці сигналів (W-OFDM) і забезпечує низький рівень позасмугового випромінювання.

Список літератури:

1. Gorbenko I.D., Zamula A.A., Semenko Ye.A. Ensemble and correlation properties of cryptographic signals for telecommunication system and network applications // Telecommunications and Radio Engineering. 2016. Vol. 75, Issue 2. P. 169-178.

2. I Gorbenko I.D., Zamula A.A.Cryptographic signals: requirements, methods of synthesis, properties, application in telecommunication systems Telecommunications and Radio Engineering. 2017. Vol. 76, Issue 12. P. 1079-1100.

3. ITU-R, Recommendation M.2083-0, “IMT Vision - Framework and overall objectives of the future development of IMT for 2020 and beyond”, ITU recommendation, Sept. 2015.

4. Pen Guan et. al. 5G Field Trials: OFDM-Based Waveforms and Mixed Numerologies // IEEE Journal on Selected Areas in Communications, vol. 35, no. 6, pp. 1234-1243, March 2017.

5. Rappaport T. S. et al. Millimeter Wave Mobile Communications for 5G Cellular: It Will Work! // IEEE Access. , 2013. vol. 1, pp. 335-349.

6. Andrews J.G. et al. What will 5G be? // IEEE Journal on Selected Areas in Communications, vol. 32, no. 6, pp. 1065-1082, June 2014.

7. Abdoli J. et al. Filtered OFDM: A new waveform for future wireless systems // Proc. IEEE SPAWC, pp. 66-70, Jun. 2015.

8. Zhang X. et al. Filtered-OFDM – Enabler for Flexible Waveformin // The 5th Generation Cellular Networks”, Proc. IEEE GLOBECOM, pp. 1-6, Dec. 2015.

9. Li, Jialing, et al. A resource block based filtered OFDM scheme and performance comparison // Proc. IEEE ICT, pp. 1-5, May 2013.

10. Marzetta T. L. Noncooperative Cellular Wireless with Unlimited Numbers of Base Station Antennas // IEEE Transactions on Wireless Communications, vol. 9, no. 11, pp. 3590-3600, Nov. 2010.
11. China Mobile Research Institute C-RAN: The Road Towards Green RAN", white paper, 2011. [Online]. Available: http://labs.chinamobile.com/cran/.
12. Nikopour H. et al. Sparse code multiple access // Proc. IEEE PIMRC, pp. 332-336, Sept. 2013.
13. 5G Forum. (2016, Mar.). 5G white paper: 5G vision, requirements, and enabling technologies [Online]. Available: http://kani.or.kr/5g/whitepaper/5G%20Vision,%20Requirements,%20and%20Enabling%20Technologies.pdf.
14. Farhang Boroujeny B. Filter bank multicarrier modulation: a waveform candidate for 5G and beyond // Advances in Electrical Engineering, vol. 2014, Dec. 2014. doi:10.1155/2014/482805.
15. Zekeriyya Esat Ankaralı et. al. Enhanced OFDM for 5G RAN. June 2017, doi: 10.3969/j. issn. 1673-5188. 2017. S1. 002.
16. Şahin A., Güvenç I. and Arslan H. A survey on multicarrier communications: prototype filters, lattice structures, and implementation aspects // IEEE Communications Surveys & Tutorials, vol. 16, no. 3, pp. 1312-1338, Aug. 2014. doi:10.1109/SURV.2013.121213.00263.
17. Huawei and HiSilicon. f-OFDM scheme and filter design // 3GPP Standard Contribution (R1‑165425), Nanjing, China, May 2016.
18. R1-165425. F-OFDM scheme and filter design. 3GPP TSG RAN WG1 meeting 85. Huawei; HiSilicon. May 2016.
19. Федосов В.П., Ковтун Д.Г., Легин А.А., Ломакина А.В. Исследование модели OFDM сигнала с малым уровнем внеполосного излучения // Известия ЮФУ. Техн. науки. 2016. С. 6-16.

Харківський національний
університет імені. В.Н. Каразіна

Надійшла до редколегії 26.02.2018

УДК 004.056.53:621.39

В.І. ЗАБОЛОТНИЙ, канд. техн. наук, А.В. ЄРМОЛОВИЧ

ДОЦІЛЬНИЙ РОЗПОДІЛ ВИТРАТ НА ВПРОВАДЖЕННЯ ЗАХОДІВ ЗАХИСТУ ВІД ТЕХНІЧНИХ ЗАСОБІВ РОЗВІДКИ

Вступ

Матеріальне виробництво сучасної продукції, розробка новітніх технологій виготовлення споживчих товарів, надання різноманітних послуг потребує захисту від конкурентів, у тому числі і від технічних засобів конкурентних розвідок. Інформацію, що озвучується і, або обробляється технічними засобами, та відомості, які проявляються власне самою продукцією або технологіями її виготовлення, конкуренти спроможні виявляти за рахунок використання різних способів, у тому числі і за допомогою застосування засобів технічних розвідок (ЗТР).

Відомо два шляхи добування даних конкурентами за допомогою ЗТР.

По-перше, це добування інформації про об'єкти розвідки, яка існує у знаковій формі [1]. Знакову форму існування інформації про об'єкти можна віднести до віртуального світу існування продуктів праці. Знакова форма представляє сукупність символів, літер, цифр, звуків, які відображають предмети та явища реального світу у віртуальному світі. Носіями інформації з обмеженим доступом (ІзОД) є документи на папері, магнітна, кіно-, відео-, фотоплівка, інші носії. Також ІзОД може зберігатися, відображатися або передаватися у формі фізичних полів (електромагнітних, оптичних, акустичних), електричних сигналів, вібрацій (у твердих предметах), тобто у вигляді інформаційних сигналів, які є об'єктом діяльності ЗТР.

По-друге, конкуренти можуть спостерігати власне за самими матеріальними об'єктами розвідки. Форма існування відомостей про об'єкти захисту проявляється самими матеріальними об'єктами реального світу у процесі виробництва й застосування продукції, технологій різного призначення теж у вигляді електромагнітних, оптичних, гравітаційних, акустичних та інших полів й випромінювання, хімічних речовин [1]. Розвідка цих ефектів за допомогою ЗТР дозволяє конкурентам синтезувати дані щодо відомостей з обмеженим доступом (ВзОД), які захищають від конкурентів.

В обох випадках виникає потреба проводити заходи захисту від ЗТР. Одним із напрямів захисту інформації, яка виражається у знаковій формі, є технічний захист інформації (ТЗІ). Захист ІзОД від витоку є важливою задачею, і на її рішення потрібні певні витрати.

Напрям захисту ВзОД, що проявляються самими об'єктами матеріального світу, також потребує певних витрат, наприклад, на маскування, приховування певних характеристик, технічну дезінформацію тощо – тобто на захист від технічних розвідок (ЗвТР).

Комплексність захисту інформації та відомостей потребує певних витрат. Оптимальний розподіл витрат на ці напрями захисту дозволить, при обмежених загальних витратах, одержати максимальний ефект.

Захист інформаційних ресурсів повинен бути одним із пріоритетних завдань безпеки підприємств України, оскільки перехід до інформаційного суспільства змінив статус інформації. Наразі вона може бути як засобом забезпечення безпеки, так і загрозою та небезпекою.

За умов постіндустріального етапу інформація перетворилась на стратегічний ресурс економічного і науково-технологічного прогресу. Відтак, захист інформації на підприємствах потребує достатнього теоретичного та методологічного підґрунтя. Дослідження можливості застосування математичних методів для оцінювання захисту інформації на підприємстві є досить актуальним питанням за сучасних умов розвитку економіки.

Серйозна увага до питань конкурентного захисту спричиняє проблеми оцінки рівня безпеки інформації на підприємствах та в організаціях, визначення величини грошових коштів,

які потрібно виділити на вирішення проблем інформаційної безпеки, розподілу грошових коштів між напрямами, які забезпечують захист інформації, зниження інформаційних ризиків. Найчастіше зазначені проблеми розв'язують на інтуїтивному рівні, без обґрунтування фінансової доцільності рішень. Адже керівництво підприємств не завжди оцінює важливість цього питання та має інформацію про співвідношення витрат на забезпечення інформаційної безпеки та збитків від втрати інформації. Іноді на інформаційній безпеці економлять, хоч це найчастіше призводить до істотних фінансових і моральних втрат, які можуть бути причиною краху. Однією з проблем інформаційної безпеки підприємства є її кількісна оцінка, а також необхідність обґрунтування вартості створення корпоративної системи захисту інформації.

Якщо рішення завдань в галузі захисту інформації приймають на інтуїтивному рівні, то існує ризик їх неоптимальності, додаткових втрат, у тому числі і економічних. Сучасні вимоги, висунуті до організації режиму інформаційної безпеки, створюють необхідність використання в своїй роботі більшого рівня захисту, а також необхідність оцінити економічну ефективність витрат на інформаційну безпеку. Економічно обґрунтовані комплекси і системи захисту інформації будуються адекватно загрозам її безпеки, що описуються у відповідних моделях.

Витрати на заходи із захисту інформації передбачаються в кошторисах робіт на проектування, будівництво, реконструкцію, розробку, створення, модернізацію, впровадження та експлуатацію об'єктів (систем, зразків, технологій). Витрати на заходи із захисту інформації здійснюються за рахунок суб'єктів, які провадять таку діяльність.

Надалі сформульовано математичну постановку і запропоновано рішення задачі, обґрунтування таких затрат на захист інформації на підприємстві, що забезпечують максимальний прибуток загальної діяльності.

Мета роботи - визначення розподілу витрат між ТЗІ і ЗвТР на забезпечення заданого рівня безпеки підприємства методом математичного моделювання.

Постановка задачі

Постановка задачі на оптимальний розподіл витрат z на захист від технічних розвідок (ЗвТР) – x і на технічний захист інформації (ТЗІ) – y.

Введені позначення:

$P_x(x)$ – ймовірність успішного ЗвТР при витратах x;

$P_y(y)$ – ймовірність успішного ТЗІ при витратах y;

$z=x+y$ – сумарні витрати на рішення задач обох задач захисту;

$P_{xy}(z=x+y)=P_x(x)P_y(y)$ – ймовірність успішного одночасного рішення задач ЗвТР та ТЗІ при витратах $z=x+y$, обумовлене незалежністю рішення задач ЗвТР та ТЗІ

$P_x(x)$, $P_y(y)$, $P_{xy}(z)$ – обмежені від 0 до 1;

$P_x(x)$, $P_y(y)$, $P_{xy}(z)$ – функції, які не зменшуються з ростом відповідного аргументу, тобто їх похідні завжди позитивні.

Задача дослідження: мінімізувати витрати $z=x+y$ при заданій ймовірності успішного одночасного рішення задач ЗвТР та ТЗІ $P_z(z=x+y)$ при їх оптимальному розподілі між ЗвТР та ТЗІ:

$$P_{xy}(z = x + y) = P_x(x)\,P_y(y). \quad (1)$$

Рішення задачі

Знайти таке співвідношення витрат x і y при заданому $P_z(z)$, що дає мінімум величини $z= x + y$, можна аналітичним пошуком екстремуму.

Перехід до однієї величини $y=z - x$ дає змогу записати (1) як

$$P_{x,y}(z) = P_x(x)P_y(z - x). \quad (2)$$

Тоді для визначення екстремуму z при фіксованій величині $P_{x,y}(z)$ по значенню x потрібно знайти похідну $P_{xy}(z)$ по x і прирівняти її 0. Рішення відносно x дасть можливість знайти $y=z-x$.

Отже,

$$P'_{x,y}(z) = (P_x(x)P_y(z-x))' = P'_x(x)P_y(z-x) - P_x(x)P'_y(z-x) \quad (3)$$

З останнього, після порівняння з 0 можна записати

$$P'_x(x)P_y(z-x) = P_x(x)P'_y(z-x), \quad (4)$$

або після перетворення:

$$P'_x(x)/P_x(x) = P'_y(z-x)/P_y(z-x)\ . \quad (5)$$

З врахуванням суті похідних від $Px(x)$, $Py(y)$:

$$w_x(x)/P_x(x) = w(z-x)/P_y(z-x) \quad (6)$$

З (6) можна знайти x , y та їх суму для заданого z пару значень x та y.

Для створення моделі необхідно визначити залежності $P_x(x)$ – ймовірність успішного ЗвТР та $P_y(y)$ – ймовірність успішного ТЗІ при витратах відповідно x та y.

Наприклад, для ілюстрації можна скористатися даними, що відображені на рис. 1 та 2.

На рис. 1 зображено ймовірність ЗвТР при витратах у розмірі x. На рис. 2 зображено ймовірність ТЗІ при витратах у розмірі y.

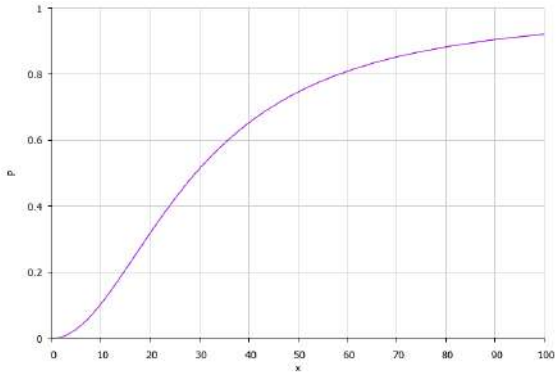


Рис. 1. Ймовірність ЗвТР при витратах x

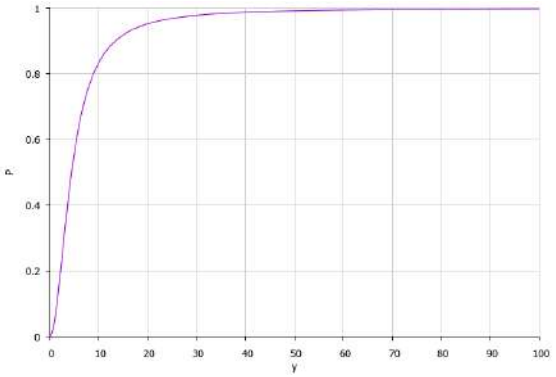


Рис. 2. Ймовірність ТЗІ при витратах y

Захист буде успішний при одночасному рішенні задач ТЗІ та ЗвТР, тобто $P_{xy}(z=x+y)=P_x(x)P_y(y)$.

Ілюстрація залежності розподілу витрат на захист у вигляді 3D-графіку функції $P_{xy}(z=x+y)$ для заданого рівня захисту, наприклад P_{xy} = 0,8, наведена на рис. 3 як перетин площини, P_{xy}=0,8 і поверхні $P_{xy}(z=x+y)$. На рисунку наведено також перетин площиною $x+y= z$, для мінімального значення z.

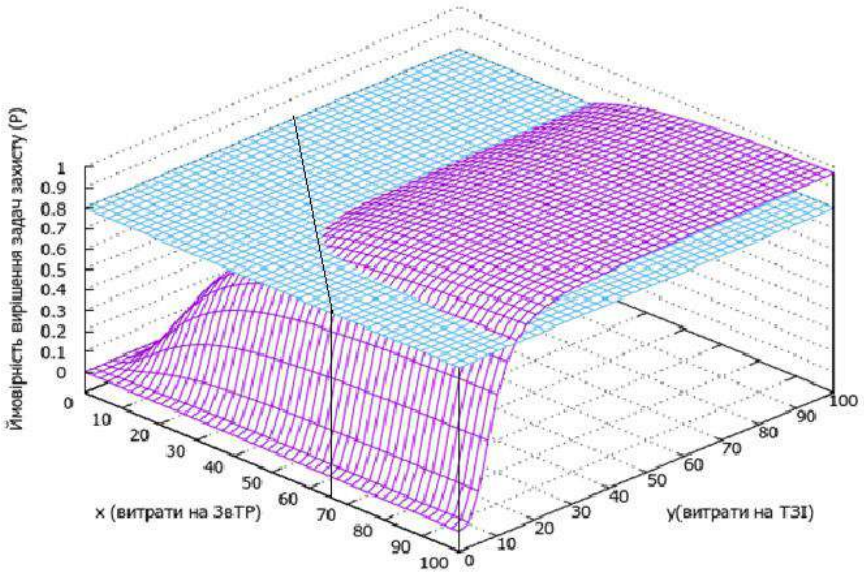


Рис. 3. Модель ймовірності захисту *P(z)* при обмежених загальних витратах *z*

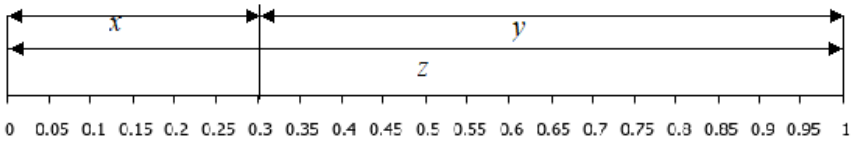


Рис. 4. Співвідношення розподілу витрат на ЗвТР та ТЗІ для наведеного прикладу

На рис. 5 наведено переріз площиною $x+y= z$ поверхні $P_{xy}(z=x+y)$. На проекції відображено розподіл витрат z між заходами ЗвТР та ТЗІ, x та y відповідно. Оптимальний розподіл визначається найбільшою ймовірністю захисту інформації. Слід зазначити, що невеликий градієнт зміни $P_{xy}(z=x+y)$ від співвідношення x та y в області оптимальних рішень дозволяє бути впевненим в надійності забезпечення задач захисту.

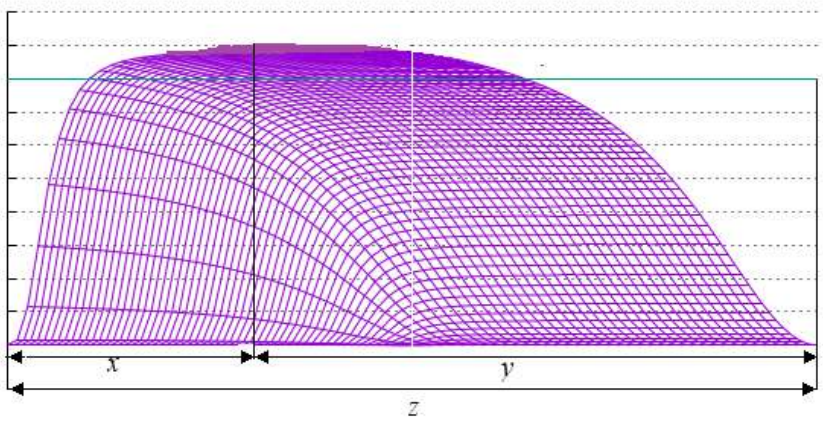


Рис. 5. Розділ витрат *z* між заходами ЗвТР та ТЗІ

Побудована модель відображає ймовірність забезпечення успішного вирішення задач захисту при сталих витратах на ТЗІ та ЗвТР. Аналізуючи схему, доцільно припустити, що лише за умов, наближених до отриманого розподілу витрат на обидва види захисту, можливо забезпечити високу ймовірність вирішення задач захисту. При проведенні розподілу витрат на два напрями є обґрунтування на розподіл коштів на ТЗІ та ЗвТР. Дуже часто ці задачі

захисту здійснюють різні підрозділи підприємства, які зацікавлені у результатах лише своєї роботи. Отже, запропонована модель дає об'єктивні засади на розподіл витрат між двома напрямами захисту таким чином, щоб вони в комплексі забезпечували задану ймовірність успішного вирішення задач захисту.

Висновки

Дослідження показали, що, незважаючи на різноманітність специфіки підприємств, існує єдність у підході до визначення витрат на створення системи захисту інформації.

Оцінка розміру показника захищеності інформації є найважливішою ланкою у виборі того чи іншого заходу. Реалізація запропонованих заходів технічного захисту інформації повинна постійно оцінюватися, як за своєчасністю й повнотою їх виконання, так і за їх ефективністю.

Визначення на основі моделі оптимізації економічно обґрунтованого обсягу коштів, що доцільно виділити на інформаційну безпеку підприємства, спрощує процес прийняття рішень керівництвом підприємства. Рішення задачі оптимального розподілу коштів між окремими напрямами захисту інформації дозволяє підприємству забезпечити мінімально можливий у межах виділеної суми рівень інформаційного ризику та витрат на проведення заходів захисту інформації. Тобто, обрати оптимальний розподіл витрат на заходи захисту, отримуючи максимальну ймовірність успішного захисту при загальних витратах.

Запропонований підхід до постановки задачі захисту інформації має деякі особливості.

По-перше, керівництво підприємства має усвідомлювати наявність існування певного рівня ймовірності захисту інформації.

По-друге, для одержання обґрунтованих рекомендацій щодо необхідних затрат на захист інформації потрібні вихідні дані про рівень захисту інформації.

Результати досліджень доцільно використовувати для визначення оптимального обсягу грошових коштів та його розподілу між окремими напрямами захисту інформації організації

Список літератури:

1. ДСТУ3396.1-96 Захист інформації. Технічний захист інформації. Порядок проведення робіт.

2. . Заболотний В.І. Варіант оптимізації витрат на систему захисту інформації / В.І. Заболотний, К.В. Петросян // Междунар. радиоелектрон. форум «Прикладная радиоэлектроника. Состояние и перспективы развития». Сб. науч. тр. Харьков, 2008. Т. 5.

3. Заболотний В.І. Обґрунтування вибору заходів захисту характеристик продукції від конкурентної розвідки / В.І. Заболотний, Є.В. Задорожна // Прикладна радіоелектроніка. 2013. Т. 12. №2. С. 351-356.

Харківський національний
університет радіоелектроніки

Надійшла до редколегії 04.03.2018

УДК 004.56:004.353.2

В.І. ЗАБОЛОТНИЙ, канд. техн. наук, В.І. ПЕРЕПАДЯ

ОСОБЛИВОСТІ МОДЕЛЮВАННЯ ПАРАМЕТРІВ ВІДЕОІМПУЛЬСУ ДЛЯ ДОСЛІДЖЕННЯ СПЕКТРІВ ПОБІЧНИХ ЕЛЕКТРОМАГНІТНИХ ВИПРОМІНЮВАНЬ

Вступ

В умовах бурхливого розвитку засобів обробки і передачі інформації, глобальної інформатизації суспільства невпинно зростають і можливості засобів технічної розвідки. Враховуючи це, актуальним питанням на сьогодні залишається необхідність вирішення проблем захисту інформації. Однією із основних загроз безпеці інформації, що обробляється технічними засобами, являється витік інформації технічними каналами, під яким розуміється неконтрольоване розповсюдження інформаційного сигналу від його джерела через середовище розповсюдження до засобу розвідки. Під час обробки інформації персональною електронно-обчислювальною машиною (ПЕОМ) технічні канали витоку інформації (ТКВІ) утворюються за рахунок побічних електромагнітних випромінювань (ПЕМВ), а також внаслідок наведень інформаційних сигналів на лінії електроживлення ПЕОМ, з'єднувальні лінії допоміжних технічних засобів і систем, кола заземлення і сторонні провідники.

Найбільш небезпечним, з точки зору витоку інформації, режимом роботи ПЕОМ є режим відтворення зображення на екрані монітору. Це обумовлено принципами роботи відеоадаптера, що складається зі спеціалізованих схем для генерування електричних сигналів управління апаратною частиною відтворення зображення. Основним елементом, в якому формується потужний сигнал, що створює ПЕМВ, є електрична схема, еквівалентом якої являється рамка зі струмом. Фізичні процеси та явища, що в ній протікають, можуть бути описані відповідними рівняннями Максвелла.

Особливий інтерес для оцінки дальності розвідки становить саме дальня зона, оскільки відстань, на яку поширюються електромагнітні коливання, може сягати десятків метрів, і саме в межах дальньої зони може знаходитися потенційний розвідувальний пристрій. Тому для розробки ефективних засобів захисту інформації від витоку каналами ПЕМВ вкрай важливо кількісно оцінити рівні випромінювань небезпечних сигналів. Відповідно до ТР ЕОТ – 95 [1] до узагальнених показників ТЗІ відносяться:

- відношення пікової напруги сигналу інформації з обмеженим доступом (ІзОД) до середньоквадратичної напруги шуму (для дискретних сигналів);
- відношення «інформаційний сигнал / шум»;
- відношення напруги небезпечного сигналу до напруги шумів (перешкод) у діапазоні частот інформативного сигналу.

Посилаючись на ТР ЕОТ – 95, до показників витоку ІзОД за рахунок ПЕМВ можна віднести абсолютні значення (на межі контрольованої зони):

E – напруженість електричного поля;

H – напруженість магнітного поля.

Якщо показник перевищує норми ефективності захисту інформації, то витік інформації можливий, якщо ні – інформація захищена. Перелічені показники можуть розраховуватись аналітично за формулами, визначатися експериментально за допомогою вимірювальної апаратури або експериментально-аналітичним шляхом із застосуванням апаратури контролю. Представляють інтерес розрахунки показників аналітичним шляхом. Такий підхід дозволяє уникнути обов'язковість вмикання ОТЗ для оцінки ТКВІ, що робить можливим розвідування ПЕМВ ще до його експлуатації. Крім того, вирішується можливість оцінки доступності ОТЗ ще до їх виготовлення.

Відомі два підходи до оцінки показників рівня ПЕМВ на ОІД: детермінований та імовірнісний [2]. До детермінованих методів відносяться методи оцінок, в яких залежність між

окремими факторами, які впливають на дальність розвідки, суворо задана максимальномож-ливими величинами. Тому оцінки призводять до однозначних результатів, які завжди суттєво перевищують реальні значення.

Суть імовірнісних методів випливає із самої назви. У теорії імовірностей імовірність – це кількісна міра, ступінь можливості появи деякої події. Даний метод, таким чином, дозволяє припустити істинність висновків і можливість існування ознак або їх відношень. Необхідність імовірнісних методів обумовлена тим, що в наукових дослідженнях доводиться мати справу з великим числом фактів, отриманих в результаті спостережень, або з великою кількістю об'єктів дослідження. Одне з основних завдань імовірнісних методів полягає у виявленні закономірностей на основі вивчення великого числа фактів, об'єктів або випадкових фактів. Незважаючи на те, що імовірнісні методи не дають однозначних відповідей, вони допомагають розрахувати ступінь достовірності та є єдино можливими при дослідженні масових явищ. Вони дозволяють встановити хоча і не сувору, і не жорстку, але стійку, повторювану закономірність, що виявляється в масі спостережень.

Оскільки не завжди існує можливість визначити величину напруженості електричного поля E або її значення неточне, єдиним правильним підходом для оцінки показників ТЗІ є імовірнісний. Його суть полягає у представленні шуканих величин у вигляді діапазону значень, які задовольняють певним вимогам. Тобто у випадку можливості ведення розвідки за допомогою даного методу можна визначити оцінку імовірності захисту об’єкта (рис. 1, де P_p – імовірність розвідки; $P_з$ – імовірність захисту; $\omega(E)$ – густина розподілу імовірності напруженості електричного поля на межі контрольованої зони; E_n – норма захисту по напруженості електричного поля; E – рівень напруженості електричного поля на межі контрольованої зони; $E_{A_д}$ – значення напруженості електричного поля при найбільш сприятливих умовах ведення розвідки).

Для цього необхідно визначити рівень випромінювання сигналу, тобто амплітуду напруженості електричного поля E на межі контрольованої зони (ось абсцис). Якщо дослідження проводяться аналітичним шляхом, з використанням детермінованого методу, враховуються максимально сприятливі умови для ведення розвідки і розраховується найбільш критичне значення напруженості електричного поля $E_{A_д}$. На осі абсцис також позначений допустимий рівень напруженості електричного поля E_n. Використовуючи метод статистичних випробувань Монте-Карло, генерують значення напруженості електричного поля у вигляді розподілу випадкової величини $\omega(E)$.

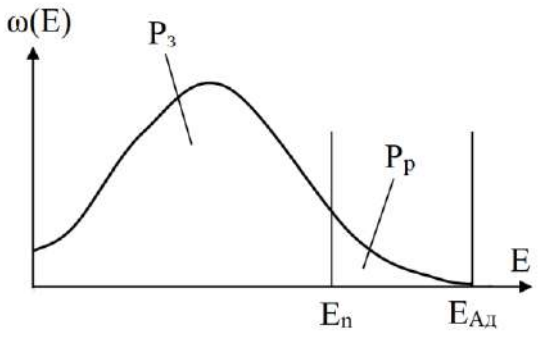


Рис. 1

Отже, імовірність захисту $P_з$ (1) визначається площею фігури під кривою в межах від 0 до E_n (рис. 1), а імовірність розвідки P_p (2) – від E_n до $E_{A_д}$ відповідно:

$$P_З(E) = \int_0^{E_n} \omega(E)dE\ , \tag{1}$$

$$P_p(E) = \int_{E_n}^{E_{A_Д}} \omega(E)dE\ . \tag{2}$$

Аналітично визначити вираз для $\omega(E)$ зазвичай не представляється можливим через складності виразів розподілів вихідних величин. Тому для розрахунків доцільно використовувати метод статистичних випробувань (метод Монте-Карло).

Метод Монте-Карло дає можливість вирішувати імовірнісні проблеми статистичними методами. Теорія цього методу вказує, як доцільно вибрати випадкові величини для розрахунків, як оцінювати одержані результати. Метод ґрунтується на багатократних прогонах (випадкових реалізаціях) на підставі побудованої моделі з подальшим статистичним опрацюванням отриманих даних з метою визначення числових характеристик досліджуваного об’єкта (процесу) у вигляді статистичних оцінок його параметрів.

Імітаційне моделювання за методом Монте-Карло дозволяє побудувати математичну модель з невизначеними параметрами, і, знаючи їх імовірнісні розподіли, а також зв'язок між змінами параметрів (кореляцію), отримати розподіл досліджуваної функції. Імовірнісний розподіл регулює імовірність вибору значень із певного інтервалу. В рамках моделі імовірнісного аналізу ризиків проводять велику кількість ітерацій, що надають можливість встановити, як поводиться результативний показник (у яких межах коливається, як розподілений) у разі підстановки в модель різних значень змінної відповідно до заданого розподілу.

Рішення задачі методом Монте-Карло

В практиці ТЗІ прийнято досліджувати ПЕМВ, використовуючи тестові сигнали. В якості тестових сигналів зазвичай обирають послідовність регулярних сигналів «піксель чорний – піксель білий» – тобто сигнал типу «меандр» (рис. 2).

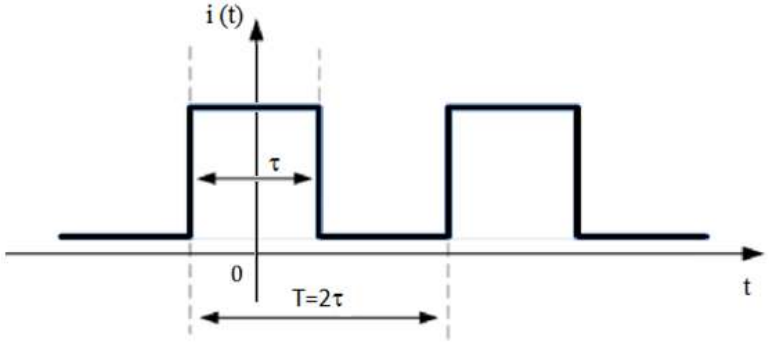


Рис. 2

Основними параметрами тестового сигналу являються: амплітуда імпульсу – A, довжина імпульсу – τ за рівнем половини амплітуди (рис. 2).

В результаті попередніх досліджень [2] встановлено, що в дальній зоні, на вході антени розвідприймача, форма розвіданого сигналу ПЕМВ визначається формою другої похідної від форми вихідного струму в колі електричної схеми випромінювача відеосигналу у вигляді рамки зі струмом (рис. 3). Окрім цього, характерною особливістю сигналів, які циркулюють в реальних електричних схемах ЗОТ, являється наявність таких параметрів: δ – довжина плавного переходу сигналу між лінійними частинами (між стаціонарним значенням і лінійною змінною, і навпаки); Δ – довжина апроксимації лінійної складової фронту імпульсу (зростання/спад сигналу від 0 до A), які і визначають форму сигналу (рис. 3). Також існує незначна відмінність між плавними переходами нижньої і верхньої частини імпульсів.

Відмінність обумовлена механізмом їх формування, а саме режимом відсічки або режимом насичення, які характерні для роботи транзистора в імпульсних схемах. Спектральна функція такого сигналу може бути представлена виразом

$$S\ (E) = \left| A\omega^2 \frac{s}{4\pi r c^2} \sin(\omega \frac{\delta}{2}) \frac{2\pi/\delta}{(2\pi/\delta)^2 - \omega^2} \sin(\omega\Delta/2) \sin(\omega\tau/2) \right|, \quad (3)$$

де – A амплітуда відео імпульсу; s – площа еквівалентної рамки випромінювача поля сигналу; r – відстань до точки оцінки рівня поля ПЕМВ; с – швидкість розповсюдження електромагнітного поля.

Крім того, в формулу розрахунку входять параметри впливу поляризаційних характеристик поля (не означені в (3)).

Аналізуючи вираз (3), можна зробити висновок, що наявність згаданих параметрів Δ та δ призводить до зменшення рівня випромінювання сигналу на високих частотах [2]. Це дозволяє мінімізувати рівень ПЕМВ, тим самим забезпечити необхідний рівень захисту інформації. У свою чергу, актуальним питанням залишається знаходження методу впливу на значення величин Δ та δ при формуванні сигналу в електричній схемі.

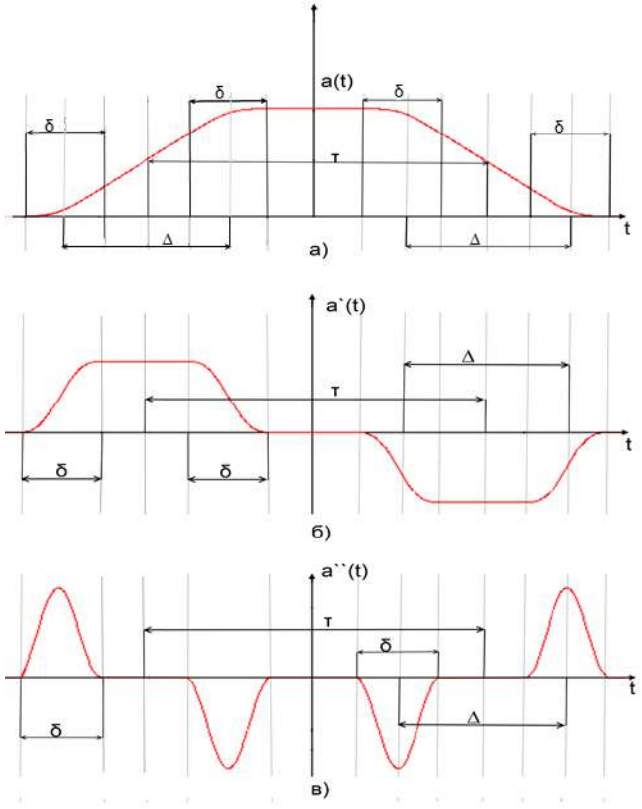


Рис. 3

Для цього, в першу чергу, необхідно змоделювати вибірку значень Δ та δ методом Монте-Карло у заданих межах. Розподіл величин при моделюванні береться рівномірний, чим забезпечується їх найбільша невизначеність. Враховуються певні обмеження, пов'язані з фізичними процесами, які протікають в ЗОТ:

$$0 \le \delta \le \Delta \le \tau. \quad (4)$$

Крім очевидних нерівностей (4) слід забезпечити виконання і нерівності (5):

$$\delta \le \tau - \Delta, \quad (5)$$

яка відкидає таке спотворення змодельованих відеоімпульсів, що зменшують їх амплітуду.

Генерація випадкових значень Δ та δ проводиться за формулами (6) та перевіряється на відповідність обмежень (4) та (5):

$$\begin{cases} \Delta = \Delta_{\min} + \xi_1(\Delta - \Delta_{\min}), \\ \delta = \delta_{\min} + \xi_2(\delta_{\max} - \delta_{\min}), \end{cases} \quad (6)$$

де $\Delta_{\min}$ – значення довжини апроксимації лінійної складової фронту імпульсу, визначається характеристиками напівпровідникових компонентів електронних схем відеотракту; $\delta_{\min}$ – мінімальне значення довжини плавних переходів в імпульсі, визначається паразитними реактивностями компонентів електронних схем відеотракту; ξ_1, ξ_2 – значення відповідних випадкових величин в межах 0 – 1.

На підставі зазначеного та (4) значення Δ та δ мають лежати в виділеній області (рис. 4).

Висновки

Одержаний підхід до генерацій значень Δ та δ дозволить обґрунтовано формувати реалізації випадкових параметрів сигналі відеотракту для коректного використання їх в оцінці рівнів ПЕМВ методом Монте-Карло.

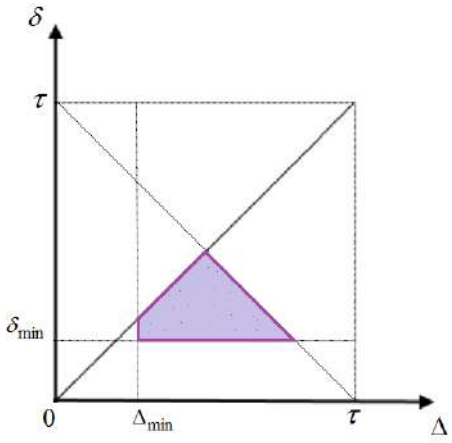


Рис. 4

Список літератури:

1. Тимчасові рекомендації з технічного захисту інформації у засобах обчислювальної техніки, автоматизованих системах і мережах від витоку каналами побічних електромагнітних випромінювань і наводок. (ТР ЕОТ-95)

2. Заболотний В.І. Забезпечення достовірності оцінки далекості виявлення випромінювань технічних засобів передачі інформації / В.І. Заболотний, О.Г. Лебедєв, О.П. Метелев // Радиотехника. 2002. Вып. 126. С. 222-226.

3 .Заболотний В.І. Дослідження змін форми сигналу у каналі побічних електромагнітних випромінювань монітору / В.І. Заболотний, Є.В. Герасименко, В.І. Перепадя // Радиотехника. 2014. Вып. 176. С. 116-121.

Харківський національний
університет радіоелектроніки

Надійшла до редколегії 19.03.2018

УДК 004.056.55

К.Ю. ШЕХАНІН, А.О. КОЛГАТІН, Є.Є. ДЕМЕНКО, О.О. КУЗНЕЦОВ, д-р техн. наук

ПРИХОВУВАННЯ ДАНИХ У СТРУКТУРУ ФАЙЛОВОЇ СИСТЕМИ СІМЕЙСТВА FAT

Вступ

Інформаційні технології визначають процеси передачі і розповсюдження, зберігання та обробки інформації, а також її використання у певних цілях [1 – 3]. Інколи факт виконання цих процесів повинен бути прихований від сторонніх осіб. Цим і займається галузь науки цифрова стеганографія [4 – 8].

Окремим розділом сучасної цифрової стеганографії, що вивчає методи і засоби вбудовування та вилучення інформаційних повідомлень в різні цифрові контейнери з використанням технічних особливостей зберігання, передачі і відображення даних, є технічна стеганографія. Вона вивчає такі методи, застосована надмірність в яких штучна і її поява зумовлена технічними особливостями обробки та передачі цифрових контейнерів [9 – 14].

На даний час відомо декілька напрямків розвитку технічної стеганографії: прихована передача інформації у мережевому трафіку [9]; приховування інформації у модель під час 3D-друку [10]; методи технічної стеганографії, що базуються на структурній особливості файлових систем у носіях інформації [11 – 14].

У роботі розглянуто третій напрямок, зокрема досліджуються методи приховування даних у структуру файлової системи сімейства FAT. Приховування реалізується шляхом перемішування кластерів певних покрівельних файлів (англ. Cover File). Пропонується удосконалений метод, який дозволяє значно збільшити обсяг прихованих даних за рахунок збільшення обчислювальної складності.

Приховування даних в кластерні файлові системи

Перші методи технічної стеганографії, які засновано на приховуванні інформаційних даних у структуру файлової системи, розглянуто в [11, 12]. Найпростіші методи застосовують вільні кластери (або певні службові поля даних) для запису прихованого повідомлення, але такий спосіб є ненадійним [13, 14]. Інші підходи застосовують надмірність, яка виникла штучно, у способі нумерації застосованих кластерів. Шляхом зміни нумерації окремих кластерів певних файлів (їх називають покрівельними файлами) вдається приховати невелику кількість інформаційних бітів [13, 14]. Розглянемо найбільш ефективний спосіб такого приховування з метою його подальшого розвитку та вдосконалення.

Для базового методу [13] застосуємо такі позначення. Нехай повідомлення M, яке буде вбудовано, позначається через масив $M=[b_0,b_1,...,b_{n-1}]$, де n – довжина повідомлення (кількість стеганоблоків), b_i – окремий стеганоблок (набір з m бітів), $i=0,1,...,n-1$. Покрівельні файли позначимо як $F_0,F_1,...,F_{p-1}$, де p – кількість покрівельних файлів, $p=2^m$ $m\in N$. Натуральне число m є ключовою інформацією.

Іменами (назвами) покрівельних файлів F_i є строки $t_i, i=0,1,...,\mathrm{p}-1$. Послідовність покрівельних файлів $F_i, i=0,1,...,\mathrm{p}-1$ (або їх назв $t_i, i=0,1,...,\mathrm{p}-1$) є ключовою інформацією.

Матриця $C=[\mathrm{c}_{ij}]$ містить номера кластерів покрівельних файлів. У такому випадку кожен покрівельний файл F_i може бути представлено у вигляді масиву

$$F_i=[\mathrm{c}_{i0},\mathrm{c}_{i1},...,\mathrm{c}_{iL_i}],$$

де L_i – загальна кількість кластерів файлу F_i.

У випадку, коли вбудовується псевдовипадкове повідомлення, довжина кожного покрівельного файлу F_i у кластерах повинна бути $L_i \approx n/m2^m$. У гіршому випадку, щоб гарантовано вбудувати повідомлення, довжина кожного покрівельного файлу F_i повинна бути $L_i \approx n/m$.

Для ***приховування повідомлення*** необхідно виконати наступну послідовність дій [13].

Повідомлення M розбивають на стеганоблоки по m біт кожен, $M=[b_0,b_1,...,b_{k-1}]$, де $k=n/m$. Якщо останній стеганоблок не повний, то необхідно доповнити його нульовими бітами.

Кожен блок b_j інтерпретується натуральним числом: $b_j \in N$, $j=0,1,...,k$; $0 \le b_j \le p-1$, яке відповідає певному покрівельному файлу $F_i, i=b_j$.

Виразити матрицю $C=[c_{ij}]$, у вигляді впорядкованого масиву $c_{ij}^0, c_{ij}^1, ...c_{ij}^w$, де $w=\sum_0^{p-1}(L_i)$. Тоді вбудовування повідомлення M задають у вигляді перестановки $\pi(c_{ij}^z)=c_{b_x y}$, де $z=0,1,...,w$; $x=0,1,...,n-1$; $y=0,1,...,L_y$. Величини x,y,z на початку роботи методу приймають нульові значення.

Повідомлення вважається вбудованим, якщо $z=w$ та $x \ge n-1$. Якщо $y>L_y$ та $x \le n-1$, то дане повідомлення не може бути вбудованим. Якщо $x>n-1$ та $z \le w$, то виконується перестановка з використанням кластерів файлів, які не брали участь у вбудовуванні повідомлення. Після кожної перестановки необхідно збільшити кожне значення $z=z+1$; $x=x+1$; $y=y+1$.

Для ***вилучення повідомлення*** необхідно мати ключову інформацію: m – натуральне число та набір покрівельних файлів $F_0, F_1, ..., F_{p-1}$. Далі необхідно виконати наступну послідовність дій [13].

Скомпоновати матрицю $C=[c_{ij}]$, що містить кластери покрівельних файлів. Дану матрицю необхідно виразити у вигляді впорядкованого масиву $c_{ij}^0, c_{ij}^1, ..., c_{ij}^w$ де $w=\sum_0^{p-1}(L_i)$.

Вилучене повідомлення $M^*=[b_0,b_1,...,b_w]$ містить стеганоблоки, вилучені із впорядкованого масиву C за таким правилом: $b_z=\pi^{-1}(c_{ij}^z)$, де $z=0,1,...,w$. Тобто значення блоку b_z дорівнює номеру покрівельного файлу кластера c_{ij}^z. Саме повідомлення вилучається шляхом конкатенації стеганоблоків $M=b_0|b_1|...|b_w$.

Для прикладу, якщо використовується два покрівельних файли ($p=2$), то значення кожного блоку b_i стеганограми може бути «0» або «1», де «0» відповідає кластеру, що належить до першого файлу, а «1» – до другого. Нехай покрівельні файли мають назви *A.txt* та *B.txt* і кожен з них займає по 10 кластерів у структурі файлової системи. Припустимо, що кластери цих файлів дефрагментовані, ланцюги кластерів файлів мають такі значення:

– для файлу *A.txt* маємо послідовність $\{3,4,5,6,7,8,9,10,11,12\}$;

– для файлу *B.txt* маємо послідовність $\{13,14,15,16,17,18,19,20,21,22\}$.

Нехай повідомлення, яке треба приховати, дорівнює 0x47. Розбивши це повідомлення на відповідні стеганоблоки, отримаємо двійковий масив $M=\{0,1,0,0,0,1,1,1\}$. Отже, виконавши етап перемішування кластерів у ланцюгах із відповідністю до масиву із стеганоблоків, отримаємо відповідні ланцюги кластерів:

– для файлу *A.txt* маємо нову послідовність $\{3,5,6,7,11,12,13,14,15,16\}$;

– для файлу *B.txt* також маємо нову послідовність $\{4,8,9,10,17,18,19,20,21,22\}$.

Отже приховане повідомлення міститься у зміненій нумерації окремих кластерів покрівельних файлів *A.txt* та *B.txt* , структуру файлової системи у спрощеному вигляді до та після приховування базовим методом наведено на рис. 1.

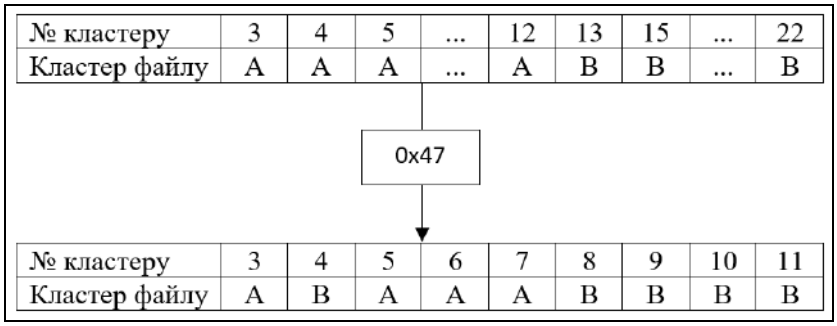


Рис. 1. Приклад приховування повідомлення 0x47 за рахунок перестановки кластерів

Слід відмітити переваги розглянутого методу. Перш за все це надійність приховування:

– обсяг інформації, що зберігається на носії, не змінюється;

– не змінюється кількість вільних чи помилкових кластерів;

– якщо носій був фрагментований, то після приховування рівень фрагментованості майже не змінюється [13, 14].

Отже, як стверджують розробники [13], немає ніяких зовнішніх ознак щодо детектування прихованого повідомлення. Збільшується лише рівень переплетеності файлів, але цей показник треба додатково досліджувати [13, 14].

До недоліків розглянутого методу слід віднести невеликий обсяг прихованих даних. Зокрема, при використанні $p = 2^m$ покрівельних файлів у кожному кластері можна приховати m інформаційних бітів. Далі пропонується удосконалений метод, застосування якого дозволяє збільшити цей показник.

Удосконалений метод приховування

Удосконалений метод приховування інформації використовує додаткову особливість розміщення файлів у структурі файлової системи, а саме – послідовність кластерів у межах одного покрівельного файлу. Тобто, зазвичай, кластери файлу $F_i = [c_{i0}, c_{i1}, ..., c_{iL_i}]$ впорядковані, так що $c_{iy} < c_{i(y+1)}$, $y = 0,1,..., L_i$. Нехай, якщо $c_{iy} < c_{i(y+1)}$ то це відповідає нульовому бітовому значенню – «0», якщо $c_{iy} > c_{i(y+1)}$, то це відповідає одиничному бітовому значенню – «1». Таким чином, перший кластер файлу не несе інформаційного значення, а є лише опорним кластером для наступних.

Для ***приховування повідомлення*** удосконаленим методом необхідно виконати наступну послідовність дій.

По-перше, необхідно повідомлення M розбити на стеганоблоки по m біт кожен, $M = [b_0, b_1, ..., b_{k-1}]$, де $k = n/m$. Причому кількість покрівельних файлів $p = 2^{m-1}$. Якщо $m \geq 2$, то повідомлення M необхідно умовно розбити на підповідомлення $M = Ml \,|\, Mr$, де $Ml = [bl_0, bl_1, ..., bl_{k-1}]$, $bl_i = b_i$ □ 1 – значення стеганоблоків без урахування найменш значущого біту, а $Mr = [br_0, br_1, ..., br_{k-1}]$, $br_i = b_i \,\&\, 0x1$ – значення найменш значущого біту кожного стеганоблоку, де $i = 0,1,..., k-1$. Після чого необхідно виконати перестановку впорядкованого масиву C за значеннями стеганоблоків повідомлення Ml як для базового методу.

Наступним кроком є перестановка кластерів у межах одного покрівельного файлу $F_i = [c_{i0}, c_{i1}, ..., c_{iL_i}]$. Першим етапом є встановлення опорного кластеру: $\pi(c_{i0}) = c_{i0}$, якщо $bl_j = i$ та $br_j = 0$, або $\pi(c_{i0}) = c_{iL_i}$, якщо $bl_j = i$ та $br_j = 1$ при $j = 0,1,...,k-1$.

Далі необхідно розглядати серії значень стеганоблоків br_j таких, що $\forall bl_j = i$ при $j = 0,1,...,k-1$ та $i = 0,1,..., p-1$. Якщо $br_j, br_{j+1}, ..., br_{j+k} = 0$, то виконується перестановка $\pi(c_{ji}) = c_{xi}$, $\pi(c_{(j+k)i}) = c_{yi}$, де $k \in N$ – розмір серії стеганоблоків із «0» значеннями, c_{xi} – перший кластер файлу F_i, який ще не брав участь у перестановці, c_{yi} – останній кластер файлу, який ще не брав участь у перестановці.

Якщо $br_j, br_{j+1}, ..., br_{j+k} = 1$, то виконується перестановка $\pi(c_{ji}) = c_{yi}$, $\pi(c_{(j+k)i}) = c_{xi}$, де $k \in N$ – розмір серії стеганоблоків із «0» значеннями, c_{xi} – перший кластер файлу F_i, який ще не брав участь у перестановці, c_{yi} – останній кластер файлу, який ще не брав участь у перестановці.

У разі, якщо усі інформаційні блоки br_i для файлу F_i були використані, то подальша перестановка виконується у довільному порядку.

Для ***вилучення повідомлення*** необхідно мати ключову інформацію: m – натуральне число та набір покрівельних файлів $F_0, F_1, ..., F_{p-1}$. Далі необхідно виконати наступну послідовність дій.

Необхідно скомпоновати матрицю $C = [c_{ij}]$, що містить кластери покрівельних файлів. Дану матрицю необхідно виразити у вигляді впорядкованого масиву $c_{ij}^0, c_{ij}^1, ..., c_{ij}^w$, де $w = \sum_0^{p-1}(L_i)$. Після чого вилучити масив стеганоблоків $M^* = [b_0, b_1, ..., b_w]$ як за базовим методом. Наступним кроком є конкатенація стегадоблоків із бітовим значенням, якщо $c_{ij}^x < c_{i(j+1)}^y$, то $b_z = b_z \,|\, 0$, та якщо $c_{ij}^x > c_{i(j+1)}^y$, то $b_z = b_z \,|\, 1$, при $x, y, z = 0,1,..., w$.

Для прикладу, якщо використовуються два покрівельних файли, то значення стеганоблоку може бути «00», «01», «10» або «11». Де «0…» відповідає кластеру, що належить до першого файлу, а «1…» – до другого. Нехай покрівельні файли мають назви *A.txt* та *B.txt*, кожен з них займає по 10 кластерів у структурі файлової системи, та ці кластери є дефрагментованими, тобто ланцюги кластерів файлів мають такі значення:

– для файлу *A.txt* маємо послідовність $\{3,4,5,6,7,8,9,10,11,12\}$;

– для файлу *B.txt* маємо послідовність $\{13,14,15,16,17,18,19,20,21,22\}$.

Нехай стеганограмою буде повідомлення 0x4747. Розбивши це повідомлення на відповідні стеганоблоки, отримаємо масив $M = \{01,00,01,11,01,00,01,11\}$.

Виконавши етап перестановки кластерів у ланцюгах із відповідністю до масиву із стеганоблоків, як для базового методу, отримаємо відповідні ланцюги кластерів:

– для файлу *A.txt* маємо нову послідовність $\{3,4,5,7,8,9,11,12,13,14\}$,

– для файлу *B.txt* також маємо нову послідовність $\{6,10,15,16,17,18,19,20,21,22\}$.

Виконавши наступний етап перестановки, отримаємо відповідні ланцюги кластерів:

– для файлу *A.txt* маємо $\{14,3,13,12,4,11,5,7,8,9\}$;

– для файлу *B.txt* маємо $\{22,21,6,10,15,16,17,18,19,20\}$.

Отже приховане повідомлення міститься у зміненій нумерації окремих кластерів покрівельних файлів *A.txt* та *B.txt*. В порівнянні із базовим методом змінено не лише відносну нумерацію кластерів окремих покрівельних файлів, але і відносну нумерацію окремих кластерів в кожному покрівельному файлі. Структуру файлової системи у спрощеному вигляді до та після приховування удосконаленим методом наведено на рис. 2.

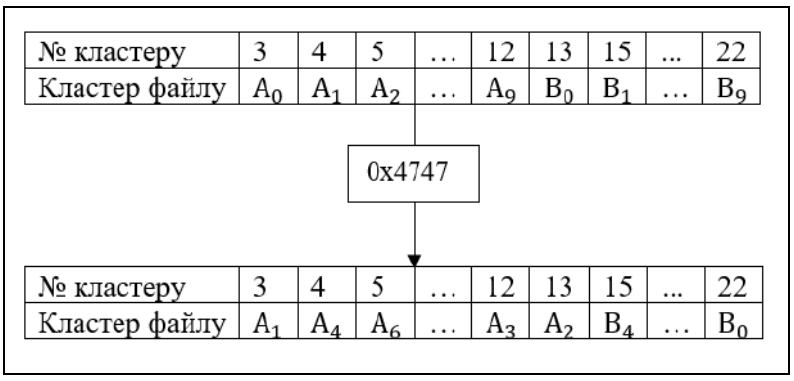


Рис. 2. Приклад приховування повідомлення 0x4747 удосконаленим методом

Порівняльний аналіз та оцінка ефективності удосконаленого методу

Для порівняльного аналізу розглянутих методів необхідно надати кількісну оцінку пропускної здатності організованого стеганоканалу. Під пропускною здатністю будемо мати на увазі максимально можливий розмір повідомлення, що приховується.

Пропускна здатність розглянутих методів залежить як від розміру одного кластеру, так і від кількості покрівельних файлів. Задля знаходження пропускної здатності припустимо, що:

– покрівельні файли займають усе вільне місце у структурі файлової системі;

– розміри покрівельних файлів рівні по відношенню один до одного;

– кількість кожного типу стеганоблоків однакова, наприклад, якщо стеганоблоки дорівнюють «0» та «1», то у повідомлені їх по 50 % кожного.

Приймемо такі позначення:

– $Data$ – загальний об’єм інформації (в байтах) у структурі файлової системі, що займають покрівельні файли, $Data = Const$;

– $Size_{Cl}$ – розмір одного кластеру у байтах, $Size_{Cl} \in \{2048, 4096, 8192, ..., 65536\}$;

– Num_{Cl} – загальна кількість кластерів, що займають покрівельні файли;

– $Size_{Fl}$ – розмір (у байтах) одного покрівельного файлу;

– Num_{Fl} – кількість покрівельних фалів;

– STG_{SIZE} – розмір стеганограми у байтах.

Так як $Data$ прийнято за константу, а покрівельні файли є рівнозначної величини, то можна стверджувати, що

$$Data = Size_{Fl} \times Num_{Fl}, \quad (1)$$

$$Data = Size_{Cl} \times Num_{Cl}. \quad (2)$$

Так як одним кластером можна відобразити один стеганоблок, а розмір одного стеганоблоку залежить (як логарифм двійковий) від кількості покрівельних файлів, то можна стверджувати, що розмір стеганограми у байтах

$$STG_{SIZE} = \frac{Num_{Cl} \times \log_2(Num_{Fl})}{8} \quad (3)$$

Для удосконаленого методу розмір одного стеганоблоку також залежить від кількості покрівельних файлів, але так як взято за увагу додаткову властивість, то кількість біт збільшується на 1 для кожного стеганоблоку, але так як перший кластер кожного покрівельного файлу не є кластером-повідомленням, то можна стверджувати, що розмір стеганограми у байтах

$$STG_{SIZE} = \frac{(Num_{Cl} - Num_{Fl}) \times \log_2(Num_{Fl} + 1)}{8} \quad (4)$$

Оцінимо *залежність пропускної здатності від розміру одного кластеру*, для цього зафіксуємо кількість покрівельних файлів, тобто $Num_{Fl} = Const$. А так як розмір покрівельних файлів однаковий, то й $Size_{Fl} = Const$. Із формул (1) та (2) можна вивести таку залежність:

$$Num_{Cl} = \frac{Size_{Fl} \times Num_{Fl}}{Size_{Cl}} \quad (5)$$

Надалі отриману залежність із формули (5) підставимо у формулу (3):

$$STG_{SIZE} = \frac{\frac{Size_{Fl} \times Num_{Fl}}{Size_{Cl}} \times \log_2(Num_{Fl})}{8} \quad (6)$$

Для спрощення аналізу у формулі (6) константні значення необхідно прийняти за одиницю та наближеними до нуля – знехтувати. $\log_2(Num_{Fl})$ – для коректності формули приймаємо рівним одиниці. Отримаємо, що $STG_{SIZE} = 1/Size_{Cl}$, тобто *чим більший розмір кластеру, тим менше повідомлення можна приховати, і як висновок – тим менша пропускна здатність методу*. Для удосконаленого методу залежність буде така сама.

Оцінимо *залежність пропускної здатності від кількості покрівельних файлів*, для цього зафіксуємо розмір кластеру, тобто $Size_{Cl} = Const$. А так як загальна кількість кластерів залежить від розміру кластеру, то й $Num_{Cl} = Const$. Узявши за основу формулу (3), замінимо константні значення одиницею та отримаємо, що $STG_{SIZE} = \log_2(Num_{Fl})$. Тобто, *чим більше покрівельних файлів, тим більше повідомлення можна приховати, і як висновок – тим більша пропускна здатність методу*. Для удосконаленого методу залежність така сама але розмір стеганограми у два рази більший при відповідних умовах. Тобто при рівних умовах, *використовуючи удосконалений метод, можна приховати у два рази більше інформації* (якщо знехтувати константними величинами). Можна стверджувати, що дана рівність $STG_{SIZE} = \log_2(Num_{Fl} + 1)$ вірна для удосконаленого методу.

Виявивши залежності, для більш коректної оцінки підставимо числові значення замість $Data$, $Size_{Cl}$, Num_{Fl}. Припустимо, що розглянуті методи будуть застосовані для приховування інформації у змінному флеш-накопичувачі. Нехай загальний розмір флеш-накопичувача займає 8 Гб, тобто $Data = 7780302848$ байт, кількість покрівельних файлів $Num_{Fl} = 2$, та розмір одного кластеру $Size_{Cl} = 2048$ байт. Тоді загальна кількість кластерів $Num_{Cl} = \frac{7780302848}{2048} = 3799952$. Підставивши отримані значення у формулу (3), знайдемо розмір стеганограми у байтах: $STG_{SIZE} = \frac{3799952}{8} = 474994$ байт.

Зафіксувавши кількість файлів, знайдемо розмір стеганограми відповідно для розміру кластеру $Size_{Cl} \in \{2048, 4096, 8192, ..., 65536\}$, отримані результати зведено до табл. 1.

Таблиця 1

Залежність розміру стеганограми від розміру кластеру

Розмір кластеру, байт	Кількість кластерів	Базовий метод, байт	Удосконалений метод, байт
2048	3798976	474872	949744
4096	1899488	237436	474872
8192	949744	118718	237436
16384	474872	59359	118718
32768	237436	29680	59359
65536	118718	14840	29680

Побудувавши графік з використанням даних табл. 1, можна впевнитись що, дійсно, розмір стеганограми залежить від розміру одного кластеру у зворотній пропорційності, як це зображено на рис. 3. Крім того, з рис. 3 наочно видно переваги удосконаленого методу, а саме – вдвічі збільшену пропускну спроможність організованого стеганоканалу.

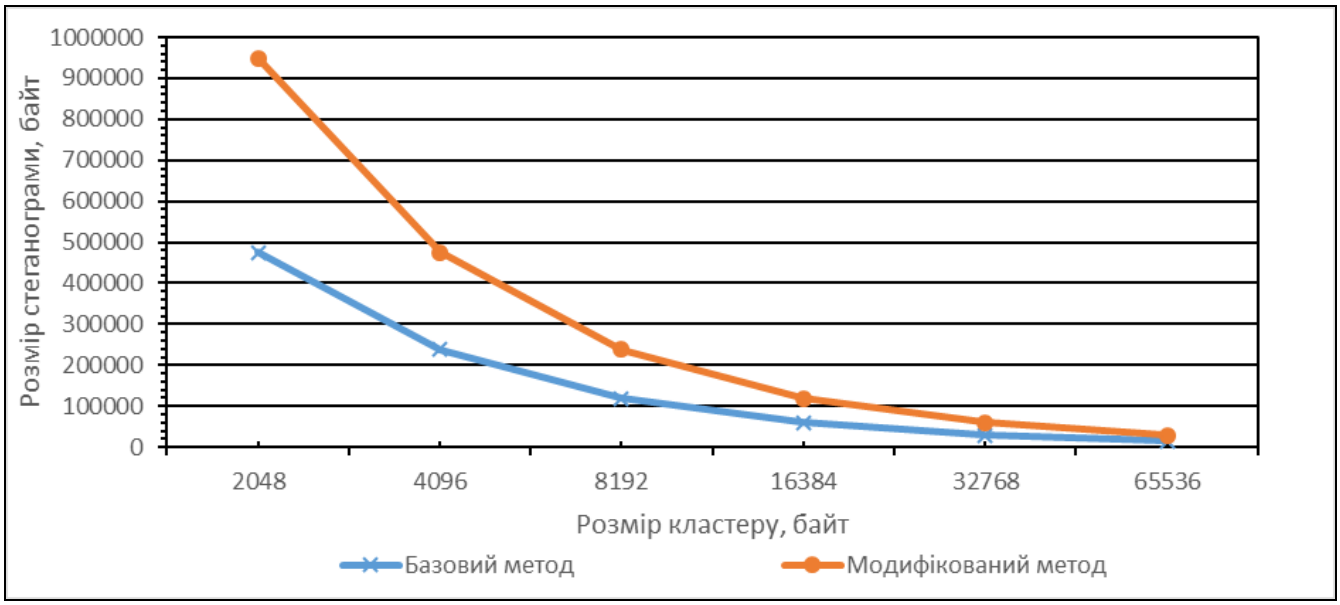


Рис. 3. Графік залежності розміру стеганограми від розміру кластеру

Аналогічну залежність виявимо і від кількості покрівельних файлів, для цього зафіксуємо загальний розмір накопичувача $Data = 7780302848$, та розмір одного кластеру – $Size_{Cl} = 2048$, кількість покрівельних файлів прийматиме такі значення – $Num_{Fl} \in \{2,4,8,16,32,64\}$. Результати розрахунків зведено до табл. 2.

Таблиця 2

Залежність розміру стеганограми від кількості покрівельних файлів

Кількість покрівельних файлів	Розмір одного файлу, байт	Базовий метод, байт	Удосконалений метод, байт
2	3 890 151 424	474872	949744
4	1945075712	949744	1424616
8	972537856	1424616	1899488
16	486268928	1899488	2374360
32	243134464	2374360	2849232
64	121567232	2849232	5698464

Згідно з даними табл. 2 побудуємо графік залежності розміру стеганограми від кількості покрівельних файлів, результати наведено на рис. 4.

Порівнюючи результати з табл. 1 та 2, можна стверджувати, що при рівних вхідних параметрах удосконалений метод дозволяє приховувати повідомлення вдвічі більшого розміру ніж базовий метод. До того ж удосконалений метод дозволяє використовувати лише один покрівельний файл, на відміну від базового методу.

Для оцінки обчислювальної складності необхідно визначити базові операції розглянутих методів.

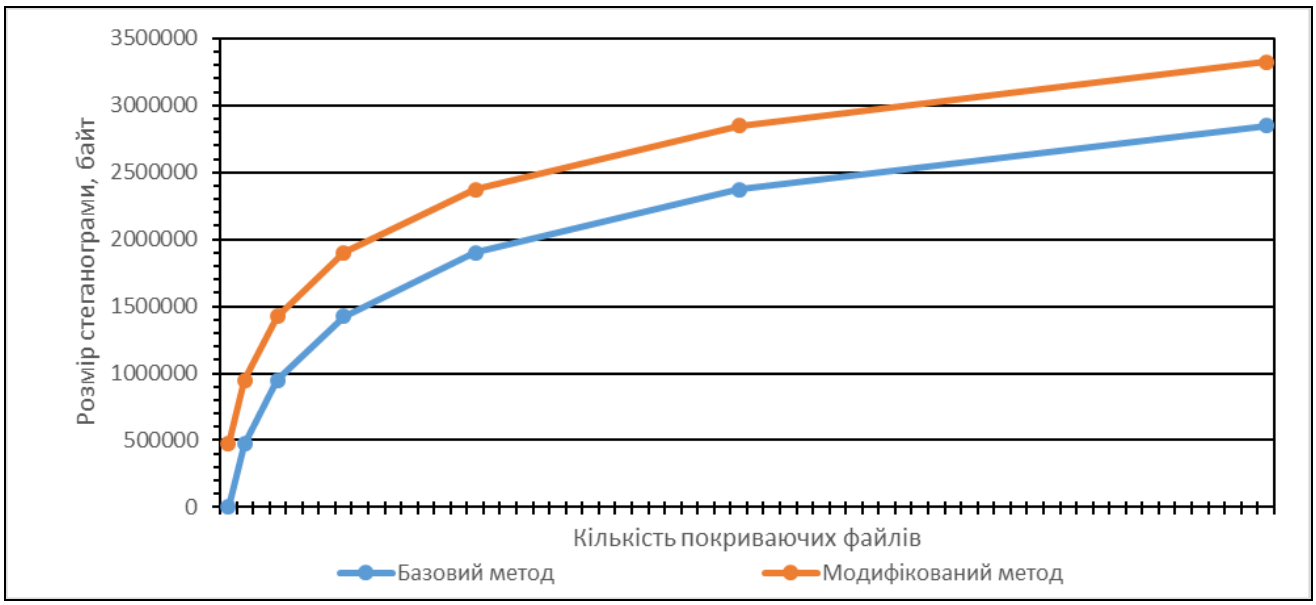


Рис. 4. Графік залежності розміру стеганограми від кількості покрівельних файлів

Оцінюючи опис методів, можна стверджувати, що базовою операцією є перестановка кластерів – $\pi(c)$. Таким чином обчислювальна складність безпосередньо залежить від кількості кластерів, які необхідно переставити. У найгіршому випадку необхідно виконати перестановку усіх кластерів покрівельних файлів, тобто $\pi_0^w(c_{ij})$, де w – сумарна кількість кластерів покрівельних файлів.

Для базового методу необхідно виконати перестановку один раз, тобто обчислювальна складність знаходиться у лінійній залежності від кількості кластерів, які необхідно переставити – $O(n)$.

Для удосконаленого методу необхідно спочатку виконати перестановку за базовим методом, а потім виконати перестановку в межах кожного покрівельного файлу. Тобто, у найгіршому випадку, кожен кластер може бути переставлено двічі. Таким чином обчислювальна складність знаходиться у лінійній залежності від кількості кластерів, які необхідно переставити – $O(2n)$.

Порівнюючи обчислювальні складності методів, можна стверджувати, що для удосконаленого методу необхідно у двічі більше обчислювальних ресурсів.

Для експериментальних досліджень ефективності розглянутих методів була використана програма «SteganoFAT», файлова система FAT32 на флеш накопичувачу JetFlash 350 Transcend ємністю 8 Гб, інтерфейс підключення USB2.0 та ноутбук Lenova Y510P із операційною системою Windows 8.1. Необхідно зазначити, що фактичний час реалізації методів приховування залежить як від апаратних особливостей носіїв інформації, так й від алгоритмічної реалізації. Проаналізуємо час роботи методів, у залежності від обраних параметрів:

– розмір кластеру;
– розмір файлу повідомлення;
– кількість покрівельних файлів;
– загальний розмір покрівельних файлів;

Для аналізу залежності затраченого часу на виконання методів приховування та вилучення повідомлення, від розміру кластеру зафіксуємо розмір повідомлення – 100 байт, кількість покрівельних файлів – 2, загальний розмір покрівельних файлів – 7 Мб. Будемо змінювати розмір кластеру – 2048, 4096, 8192 байт. Отримані результати експериментальних досліджень зведено у табл. 3. В цій таблиці і далі наводиться час, витрачений базовим методом та, через дріб, час, витрачений удосконаленим методом.

Таблиця 3

Часові витрати в залежності від розміру кластеру

Розмір одного кластеру, байт	2048	4096	8192
Час приховування повідомлення, с	3.341/6.276	2.87/5.41	2.37/3.96
Час вилучення повідомлення, с	0.022/0.021	0.012/0.012	0.008/0.009

Як видно із табл. 3, при збільшенні розміру кластеру час на приховування повідомлення зменшується. Удосконалений метод потребує вдвічі більше часу на приховування інформації, ніж базовий, при тій самій конфігурації. На час вилучення інформації обраний метод не впливає.

Для оцінки залежності витраченого часу на виконання приховування та вилучення повідомлення від розміру повідомлення зафіксуємо розмір кластеру – 2048 байт, кількість покрівельних файлів – 2, загальний розмір покрівельних файлів 7 Мб. Будемо змінювати розмір повідомлення: 100, 200, 400 байт. Результати часового аналізу вказані у табл. 4.

Таблиця 4

Часові витрати в залежності від розміру повідомлення

Розмір повідомлення, байт	100	200	400
Час приховування повідомлення, с	3.82/7.34	5.84/10.12	8.36/15.57
Час вилучення повідомлення, с	0.02/0.02	0.02/0.02	0.03/0.03

Як видно із табл. 4, при збільшенні розміру повідомлення час на приховування та вилучення повідомлення збільшується.

Для оцінки залежності витраченого часу на виконання приховування та вилучення повідомлення від кількості покрівельних файлів зафіксуємо розмір кластеру – 2048 байт, розмір повідомлення – 100 байт, загальний розмір покрівельних файлів – 7 Мб. Будемо змінювати кількість покрівельних файлів: 2, 4, 8. Отримані результати зведено у табл. 5.

Таблиця 5

Часові витрати в залежності від кількості покрівельних файлів

Кількість покрівельних фалів	2	4	8
Час приховування повідомлення, с	4.693/8.601	2.76/5.31	2.704/5.01
Час вилучення повідомлення, с	0.022/0.017	0.025/0.026	0.032/0.031

Як видно із табл. 5, при збільшенні кількості покрівельних файлів час на приховування повідомлення зменшується. Це пов’язано із тим, що кількість інформаційних кластерів при збільшенні кількості покрівельних файлів зменшується та відповідно збільшується кількість кластерів, які будуть записані впорядковано, а не перемішано. При вилученні повідомлення затрачений час збільшується із кількістю покрівельних файлів.

Для оцінки останньої залежності витраченого часу на виконання приховування та вилучення повідомлення від загального розміру покрівельних файлів зафіксуємо розмір кластеру – 2048 байт, кількість покрівельних файлів – 2, розмір повідомлення 100 байт. Будемо змінювати загальний розмір покрівельних файлів: 1.7, 3.5, 7 Мб. Отримані результати наведено у табл. 6.

Як видно із табл. 6, при збільшенні загального розміру покрівельних файлів час на приховування та вилучення повідомлення збільшується.

Таблиця 6

Часові витрати від загального розміру покрівельних файлів

Загальний розмір покрівельних файлів, Мб	1.7	3.5	7
Час приховування повідомлення, с	2.083/4.201	2.417/5.01	3.293/6.71
Час вилучення повідомлення, с	0.007/0.006	0.012/0.013	0.021/0.02

Висновки

Запропоновано та досліджено удосконалений метод приховування та вилучення інформаційних даних у структуру файлової системи сімейства FAT. Роблячи висновки із отриманих експериментальних результатів, можна стверджувати:

- на час приховування та вилучення повідомлення здебільшого впливає кількість кластерів, над якими необхідно виконати перестановку;
- вилучення виконується значно швидше за приховування повідомлення, це пов'язано із відсутністю необхідності виконувати перестановку над кластерами під час вилучення повідомлення;
- час вилучення повідомлення не залежить від обраного методу.

Таким чином, доцільним є використання обох методів при приховуванні інформації у структуру файлової системи. Вибір методу приховування залежить від: наявності часу для приховування та вилучення повідомлення; наявності обчислювальних ресурсів; вхідних параметрів методу (розмір та структура повідомлення, кількість та розміри покрівельних файлів). Ці та інші налаштування потребують подальших розробок та дослідження.

Список літератури:

1. Горбенко І.Д., Горбенко Ю.І. Прикладна криптологія. Теорія. Практика. Застосування : підручник для вищих навч. закладів. Харків : Форт, 2013. 880 с.
2. Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone. Handbook of Applied Cryptography – CRC Press, 1997. 794 p.
3. N. Ferguson and B. Schneier. Practical Cryptography. John Wiley & Sons, 2003, 432 p.
4. Petitcolas F. Information Hiding / F. Petitcolas, R. J.Anderson, M. G. Kuhn // Proceedings IEEE. 1999. Vol. 87, №. 7. pp. 1069-1078.
5. Конахович Г.Ф., Пузиренко О.Ю. Компьютерная стеганография. Теория и практика. К. : МК-Пресс, 2006. 288 с.
6. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография. Москва : Солон-Пресс, 2002. 272 с.
7. Хорошко В.А., Шелест М.Е. Введение в компьютерную стеганографию. К., 2002. 140 с.
8. Горбенко І.Д. Захист інформації в ІТС. Криптографічний захист інформації : навч. посібник. Харків : ХНУРЕ, 2004. 376 с.
9. Пескова О.Ю., Халабурда Г.Ю. Применение сетевой стеганографии для защиты данных, передаваемых по открытым каналам интернет // Информ. системы для науч.х исследований (IMS-2012). Тр. XV Всерос. объединенной конф. "Интернет и современное общество" (IMS-2012). Санкт-Петербургский нац. иссл. ун-т информ. технологий, механики и оптики, 2012. С. 348-354. http://ojs.ifmo.ru/index.php/IMS/article/download/132/132
10. Кузнецов А.А., Коваленко О.Ю. Стеганографическая защита информации с использованием 3D-печати // Інформаційна безпека держави, суспільства та особистості : зб. тез доповідей Всеукр. наук.-практ. конф., 16 квітня 2015 року. Кіровоград : КНТУ, 2015. С. 91-92.
11. Hassan Khan, Mobin Javed, Syed Ali Khayam, Fauzan Mirza. Designing a cluster-based covert channel to evade disk investigation and forensics. Computers & Security Volume 30, Issue 1, January 2011.
12. Hassan Khan, Mobin Javed, Fauzan Mirza. Evading Disk Investigation and Forensics using a Cluster-Based Covert Channel. National University of Science & Technology (NUST), Islamabad 44000, Pakistan.
13. Nerijus Morkevičius, Grigas Petraitis, Algimantas Venčkauskas, Jonas Čeponis. Covert Channel for Cluster-based File Systems Using Multiple Cover Files. Information Technology and Control, 2013, Vol.42, No.3. – p 32.
14. Кузнецов А.А., Швагер А.С., Фесенко Д.А. Сокрытие данных в кластерных файловых системах // Радиотехника. 2015. Вып. 181. С. 86-100.

Харківський національний
університет імені. В.Н. Каразіна

Надійшла до редколегії 15.03.2018

УДК 004.056.55

Є.В. БРОШЕВАН, О.В. ПОТІЙ, д-р техн. наук

ОСОБЛИВОСТІ РЕАЛІЗАЦІЇ EDELIVERY В КОНТЕКСТІ ЕЛЕКТРОННИХ ДОВІРЧИХ ПОСЛУГ. ДОСВІД ЄВРОСОЮЗУ

Вступ

Електронні довірчі послуги згідно eIDAS [1] включають в себе електронні підписи, електронні штампи, електронні мітки часу, електронні сертифікати, послуги автентифікації веб-сайтів і електронні реєстровані служби доставки. Такі послуги тепер будуть застосовуватися в ході судового розгляду в рамках ЄС. eIDAS також регулює юридичний статус організацій, які надають зазначені послуги з метою забезпечення їх надійності та юридичної валідності в разі виникнення суперечок.

Регулювання eIDAS закріплює, що електронні підписи мають обов'язкову юридичну силу і є допустимими в суді. Стаття 25 встановлює, що «юридична сила і допустимість як доказ в суді електронного підпису не може заперечуватися лише на тій підставі, що вона існує в електронному вигляді або що вона не відповідає вимогам, що пред'являються до кваліфікованого електронного підпису».

Ухвалення регулювання eIDAS відкриває нові можливості для громадян ЄС, дозволяючи транскордонно здійснювати за допомогою інтернету такі дії, як подача податкових декларацій, вступ до іноземного університету або дистанційне відкриття рахунку в банку. Взаємно визнані схеми ідентифікації дозволяють людині взяти участь в таких транскордонних взаємодіях з урядами інших країн, використовуючи свої власні національні схеми ідентифікації. Довірчі послуги, такі, як автентифікація веб-сайту, використання тимчасових міток і електронних підписів, також гарантуватимуть, що громадяни можуть безпечно взаємодіяти в середовищі онлайн-бізнесу.

Для підприємств регулювання eIDAS також відкриває нові можливості, наприклад проведення юридично значущих цифрових транзакцій по всій території ЄС, створення бізнесу в іншій державі-члені, здійснення перевірки автентичності інтернет-платежів або пропозицію ціни на торгах в Інтернеті.

На підставі Регулювання eIDAS був розроблений Закон України «Про електронні довірчі послуги» [2]. Даний проект спрямований:

- на створення умов для розвитку і функціонування сфери електронних довірчих послуг;
- вільного обігу електронних довірчих послуг в Україні, а також можливості вільного доступу до електронних довірчих послуг постачальникам електронних довірчих послуг, які здійснюють діяльність в інших державах;
- підвищення рівня довіри громадян до електронних послуг, в тому числі транскордонних;
- рівних можливостей для доступу до електронних довірчих послуг, в тому числі для осіб з обмеженими можливостями;
- свободи договору в сфері електронних довірчих послуг;
- захисту прав і законних інтересів користувачів електронних довірчих послуг;
- відповідності вимог до надання електронних довірчих послуг європейським і міжнародним стандартам;
- інтероперабельності та технологічної нейтральності національних технічних рішень, а також недопущення їх дискримінації;
- захисту персональних даних, які обробляються при наданні електронних довірчих послуг;
- відкритості для інновацій в сфері електронних довірчих послуг.

Для того щоб Україні бути активним учасником на європейському ринку і конкурувати з існуючими гравцями, необхідно також впровадити належні умови для електронного документообігу між учасниками з різними юрисдикціями. Сьогодні впровадження такого обміну неможливе з двох причин: правової (відсутність законодавчого регулювання питання довірчих послуг інших держав) і технічної (у кожної держави свій так званий ключ держави).

Прийнятий Закон "Про електронні довірчі послуги" вирішує першу проблему – надає можливість урядам держав проводити переговори щодо впровадження єдиної системи загального визнання, яка, в свою чергу, відкриє можливість реалізувати дійсну інтероперабельність (сумісність технічних рішень). Саме після реалізації технічної частини у всіх суб'єктів, які співпрацюють з європейськими партнерами, з'явиться реальна можливість обмінюватися важливими електронними документами і не чекати місяцями пересилання підписаних паперів.

Саме тому розглядаються можливості впровадження електронної довірчої послуги eDelivery, її технічні та організаційні особливості та аналізується досвід впровадження ЄС.

Огляд послуги eDelivery як частини CEF

Європейська комісія прийняла рішення сприяти впровадженню eDelivery в Європі через одну зі своїх програм фінансування, пов'язаних із засобами зв'язку (CEF). У 2014 році інфраструктура цифрових послуг eDelivery (DSI) була включена до Програми роботи CEF Telecom 2014 року, в результаті чого фінансувалася в розмірі восьми мільйонів євро протягом чотирьох років, до 2018 року [3].

Технічне управління DSI eDelivery здійснюється Генеральним директоратом з інформатики (DIGIT) Європейської комісії. За реалізацію політики ЄС, безпосередньо пов'язаної з eDelivery, несе відповідальність Генеральний директорат мереж зв'язку, контенту і технологій (DG CNECT) Європейської комісії.

Будь-яка область політики ЄС (правосуддя, закупівлі, захист споживачів), особи, які потребують надійний, транскордонний та міжсекторний обмін документами і даними (структуровані, неструктуровані та/або виконавчі), можуть використовувати технічну специфікація, запропоновану eDelivery DSI.

eDelivery підтримує основоположний принцип епохи цифрових технологій шляхом сприяння узгодженню між його технічними специфікаціями і нормативною базою eIDAS. Деякі ключові поняття, викладені в eIDAS, безпосередньо пов'язані з eDelivery:

Стаття 3 – Визначення "електронна реєстрована служба доставки" означає службу, яка надає можливість передавати дані між третіми особами електронними засобами і забезпечує докази, пов'язані з обробкою переданих даних, включаючи підтвердження відправлення та отримання даних, і це захищає передані дані від ризику втрати, крадіжки, пошкодження або будь-якої іншої несанкціонованої зміни.

Стаття 43 – Юридичний вплив на електронну реєстровану службу доставки. Дані, відправлені та отримані за допомогою служби електронної реєстрованої доставки, не можна позбавити юридичного ефекту і допустимість доказів як законних в судочинстві виключно на підставі того, що вони в електронній формі або що вони не відповідають вимогам служби кваліфікованої електронної реєстрованої доставки. "Дані, відправлені та отримані за допомогою служби кваліфікованої електронної реєстрованої доставки, користуються презумпцією цілісності даних, відправлення цих даних ідентифікованим відправником, їх отримання визначеним адресатом та точність дати і часу відправки і квитанція, зазначена службої кваліфікованої електронної реєстрованої доставки ".

Регламент eIDAS. Стаття 46 – Правові наслідки електронних документів. "Від електронного документу не повинні відмовитися в юридичних процесах та не прийняти як докази в судочинстві виключно на підставі того, що він в електронному вигляді ".

eDelivery – це мережа вузлів для цифрового зв'язку (рис. 1). Вона заснована на розподіленій моделі, де кожен учасник стає вузлом, використовуючи стандартні транспортні протоколи і політики безпеки [3]. eDelivery допомагає державним адміністраціям обмінюватися електронними даними і документами з іншими державними адміністраціями, підприємствами та громадянами на сумісному, надійному та безпечному шляху.

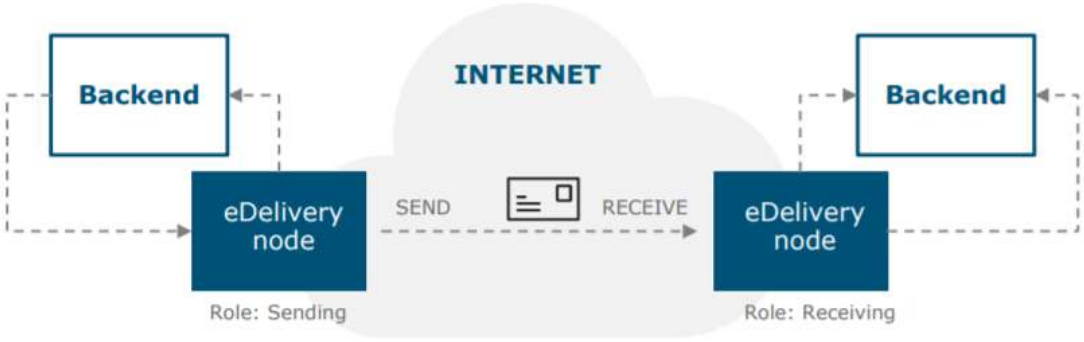


Рис. 1. Схема роботи eDelivery

eDelivery є одним з будівельних блоків Європейського союзу, пов'язаного з Європейським фондом (CEF). Ці будівельні блоки є багаторазовими специфікаціями, програмним забезпеченням і послугами, які стануть частиною широкого спектра IT-систем в різних областях політики ЄС. Структурний блок CEF eDelivery заснований на протоколі обміну повідомленнями AS4, відкритому і вільному для всіх, розробленому організацією з розробки стандартів OASIS. Щоб полегшити його прийняття в Європі, eDelivery використовує керівні принципи впровадження AS4, визначені державами-членами в пілот-центрі e-SENS. Організації повинні встановити точку доступу або використовувати постачальника послуг для обміну інформацією з протоколом обміну повідомленнями AS4.

eDelivery працює як сукупність розподілених вузлів, які відповідають тим самим технічним правилам і тому здатні взаємодіяти один з одним. eDelivery надає технічні специфікації, які можуть використовуватися в будь-якому Політичному домені ЄС (юстиція, закупівлі, захист споживачів і т. д.), щоб забезпечити безпечний і надійний обмін документами і даними (структурований, неструктурований і / або двійковий), крізь кордони і сектори. В результаті організації, які розробили свої IT-системи незалежно один від одного, можуть почати безпечно зв'язуватися один з одним, як тільки вони підключилися до вузла eDelivery.

Важливо відзначити, що немає єдиного вузла eDelivery для кожної держави-члена, крім кількох. Кожен з цих вузлів розгортається для певного пан'європейського проекту в рамках даного домену політики: eJustice, eProcurement. Зазвичай вузли eDelivery є уні-доменними і уні-проектними.

eDelivery покриває велику кількість доменів, до яких воно може бути застосовано, а саме – сільське господарство, рибальство та продукти харчування, бізнес, кліматична дія, різні політики, культура, освіта та молодь, економіка, фінанси та податки, працевлаштування та соціальні права, енергетика та природні ресурси, навколишнє середовище, споживачі та здоров'я, зовнішні зв'язки, правосуддя, внутрішні справи та права громадян, регіони та місцевий розвиток, наука і технології. Рис. 2 показує домени, в яких здійснюється eDelivery протягом 2015 – 2017 років.

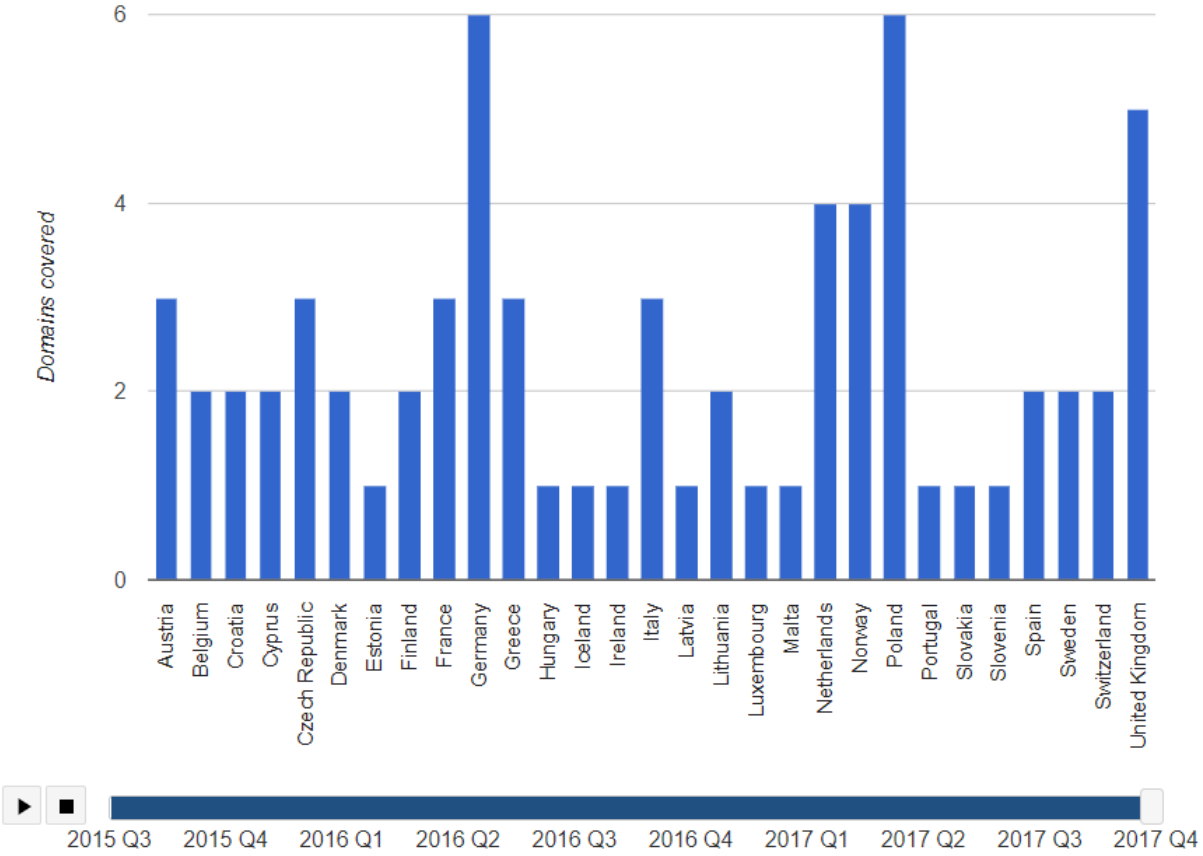


Рис. 2. Кількість доменів eDelivery по країнах

Організаційні особливості

Вузли eDelivery можуть бути реалізовані на будь-якому адміністративному рівні: національному, регіональному, місцевому або єдиному. Модель розгортання повинна бути визначена заздалегідь пан'європейським проектом. Власники доменів політики, які беруть участь в розгортанні ЄС або національних політик, що вимагають безпечного обміну документами і даними, є головною метою послуг DSI DDS[4].

Діаграма на рис. 3 роз'яснює позиціонування і обсяг eDelivery.

Прийнявши eDelivery, ці суб'єкти забезпечують таку ситуацію, щоб державні адміністрації могли обмінюватися будь-якими типами даних і документів через кордони. Це означає, що комунікація адміністрація-адміністрація (A2A) сприяє створенню єдиного європейського ринку, який підходить для цифрової епохи. eDelivery також може використовуватися в сценаріях адміністрація-бізнес (A2B) і бізнес-адміністрація (B2A), що підтверджується реалізацією eDelivery в домені eProcurement PEPPOL.

Він також може забезпечити приєднання державних адміністрацій до громадян (A2C і C2A), наприклад, коли вони користуються веб-сайтами. Наприклад, eDelivery дозволяє порталу eJustice обмінюватися даними з іншими інформаційними системами. Зв'язок між громадянами (C2C) виходить за рамки eDelivery.

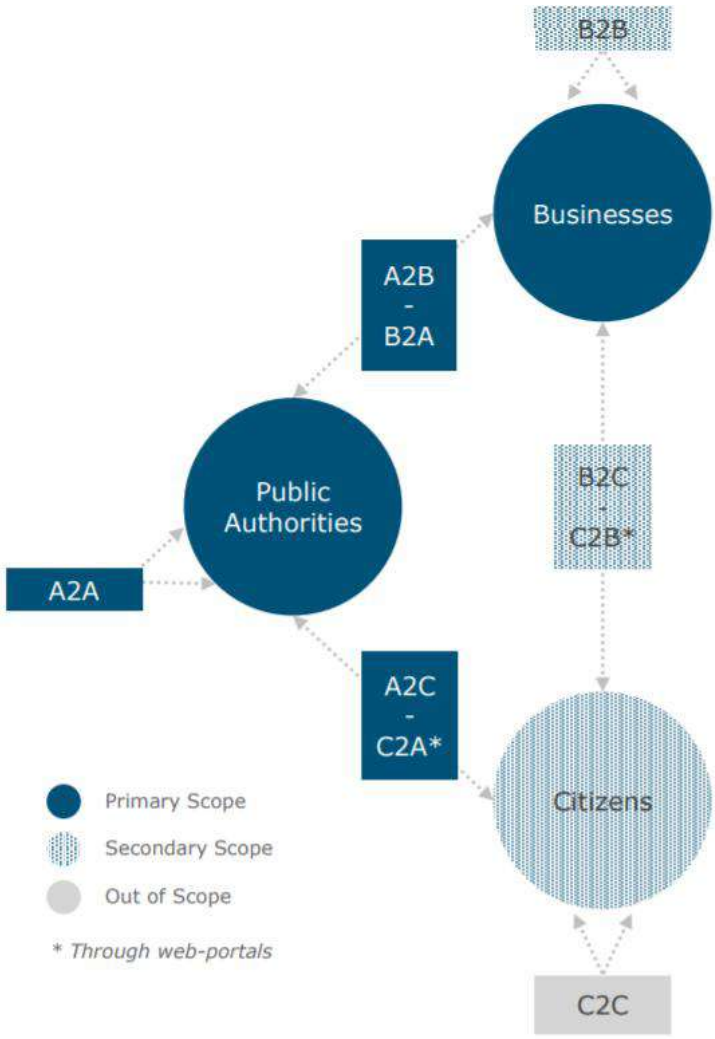


Рис. 3. Обсяг покриття eDelivery

Архітектурні особливості

Архітектура системи eDelivery (рис. 4) складається з трьох ключових компонент: точка доступу, служби визначення метаданих і служба публікації метаданих [5].

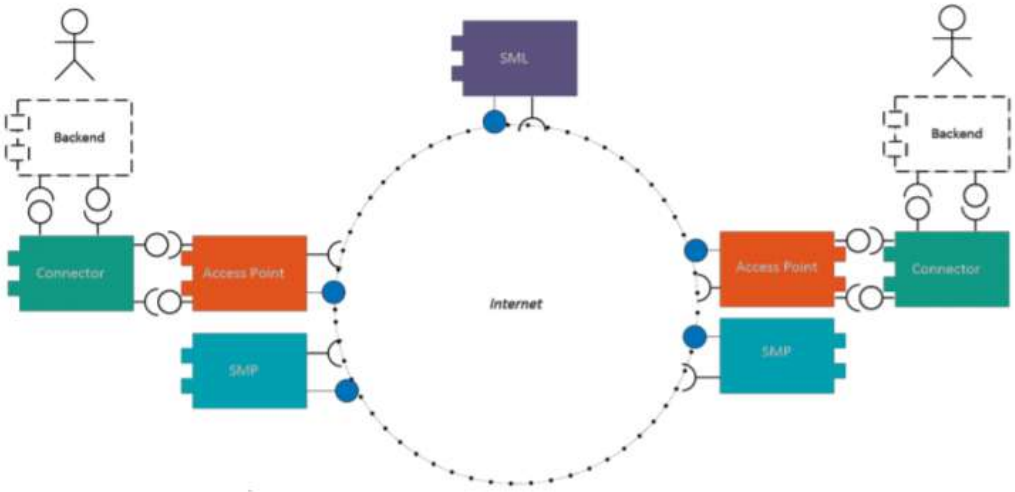


Рис. 4. Архітектура eDelivery

1. Точка доступу електронної доставки CEF (AP) реалізує стандартизований протокол обміну повідомленнями, який забезпечує сумісний, безпечний та надійний обмін даними. CEF eDelivery AP – це реалізація профілю AS4, розробленого e-SENS або профілю AS2, розробленого OpenPEPPOL. AS4 – це відкрита технічна специфікація для безпечного та корисного обміну даними з використанням Web-сервісів. Згідно OASIS, протокол AS4 є сучасним послідовником протоколу AS2. В рамках тимчасового періоду, очікуваного процесу конвергенції в відношенні профілю E4 SENS AS4, CEF eDelivery підтримує профіль OpenSEPPOL AS2, який в даний час використовується в області електронних закупівель. Domibus – це проект з відкритим кодом точки доступу AS4, який підтримується Європейською комісією. Інші постачальники програмного забезпечення пропонують альтернативні реалізації профілю e-SENS AS4 (комерційні або відкриті). Кожен постачальник програмного забезпечення також надає різноманітні послуги з доданою вартістю від інтеграції до підтримки щоденних операцій. Приклад програмного забезпечення Domibus можна використовувати для ознайомлення з профілем e-SENS AS4 в тестовому середовищі або в якості робочого рішення у виробничому середовищі.

2. Служба метаданих CEF eDelivery Service (SML) дозволяє точкам доступу динамічно виявляти IP-адресу кінцевої точки доступу. Замість того, щоб дивитись на статичний список IP-адресів, точка доступу звертається до видавця метаданих послуг (SMP), в якому є інформація про кожного учасника мережі обміну документами/даними, яка підтримується в актуальному стані, включаючи IP-адреси їх доступу. У будь-який момент часу в одній мережі може бути один або декілька активних SMP. Для того щоб динамічне виявлення працювало, кожному учаснику має бути присвоєно унікальний ідентифікатор у вигляді URL-адреси веб-сайту, який повинен бути знайдений в системі доменних імен (DNS) в Інтернеті, завдяки розташуванню служб метаданих (SML). Враховуючи цей URL, точка доступу може динамічно знаходити правильний SMP і, отже, правильну точку доступу. Цей програмний компонент SML підтримується Європейською комісією, реалізує специфікацію розташування служби метаданих бізнес-документа OASIS (BDXL). Транспортна інфраструктура буде підтримуватися відповідно до специфікацій SML PEPPOL [6].

3. CEF eDelivery Service Metadata Publisher (SMP) дозволяє учасникам інфраструктури електронної доставки eDelivery динамічно виявити можливості один одного (юридичні, організаційні та технічні). Щоб це сталося, кожний учасник повинен опублікувати свої можливості та настройки в SMP. SMP містить ключову інформацію про учасників великих інфраструктурних елементів eDelivery, наприклад:

- бізнес-процеси, які підтримує учасник;
- конфігурація безпеки (сертифікат відкритого ключа);
- транспортний протокол (AS2 або AS4);
- розташування точки доступу приймача.

SMP звичайно розподіляє метадані для декількох учасників, тим не менше, кожен учасник публікує свої метадані в одній і єдиній SMP. Через цю розподілену архітектуру кожен учасник повинен мати унікальний ідентифікатор. Центральний компонент, що називається Локатор метаданих послуг (SML), використовує ці ідентифікатори для створення URL-адрес, які при дозволі спрямовують точки доступу до електронної доставки в бік конкретного SMP учасника. Поточний програмний компонент SMP підтримується Європейською комісією, реалізує специфікацію PEPPOL SMP. Програмний компонент SMP буде оновлено для того, щоб стати сумісним з профілем SMP e-SENS на основі специфікації OASIS Service Metadata Publishing (BDX SMP).

Висновки

Таким чином, ми бачимо, що розглянута технологія широко використовується в країнах ЄС та надає гнучкі механізми інтеграції з існуючими системами для реалізації реєстрованої доставки документів. Серед переваг, які надає eDelivery, є зниження вартості створення, об-

слуговування та експлуатації мереж обміну документами та даними, оскільки деякі з цих витрат можуть бути надані іншим постачальникам послуг eDelivery, підвищення якості мереж обміну даними та документами, прискорення часу доставки робочого документа та мережі обміну даними, оскільки eDelivery виходить за межі специфікацій та програмних компонентів.

Підхід, який використовується eDelivery, – це сприяти використанню існуючих технічних специфікацій та стандартів, а не спроби визначати нові. Виходячи з проаналізованих організаційних та архітектурних особливостей, можна зробити висновок, що українська нормативна та технологічна база є достатньою для цього впровадження.

Саме це рішення є універсальним і потребує подальшого вивчення з метою впровадження на українському ринку електронних довірчих послуг.

Список літератури:

1. eIDAS Regulation. [Електронний ресурс]. Режим доступу: https://www.eid.as/home/.
2. Закон Украины “Про електронні довірчі послуги”. [Електронний ресурс]. Режим доступу: http://zakon2.rada.gov.ua/laws/show/2155-19
3. eDelivery Overview [Електронний ресурс]. Режим доступу: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eDelivery
4. Introduction to the Connecting Europe Facility eDelivery building block, CEF eDelivery (2015)
5. Electronic Registered Delivery Service (ERDS) and the eIDAS Regulation (2016)
6. What is PEPPOL? [Електронний ресурс]. Режим доступу: https://peppol.eu/what-is-peppol/

Національний аерокосмічний університет
імені М.Є. Жуковського “ХАІ”;
Акціонерне товариство
«Інститут інформаційних технологій»

Надійшла до редколегії 05.03.2018

UDC 004.056

A.V. POTII, A.S. KARPENKO

REALIZATION OF THE MECHANISM OF CONTROL SOFTWARE INTEGRITY IN POST QUANTUM PERIOD

1. Introduction

Digital signature is important primitive of modern cryptography. Most security protocol such as SSH, TLS, SSL are using digital signature for verify the integrity and authenticity of the information. The resistance of the cryptographic algorithms with the public key is based on the computational complexity of the problems of factorization of large integers, discrete logarithms and transformation of the points on the elliptic curve. The known algorithms are RSA, DSA, and ECDSA (Table 1) [1].

Investigations in the sphere of quantum calculations form up new challenges in the given sector of the cryptography. With using of the quantum computer and the Shor algorithm the known at present crypto algorithms with the public key would be compromise. Today, regional organizations such as NIST and ETSI are already research in this field. The workgroups of ETSI and NIST determined the promising trends, within the framework of which there could be obtained acceptable solutions – the supersingular elliptic curves, the multi-variative cryptography, the cryptography on the basis of the noise immunity encoding, and the cryptography based on the hash functions. Recently, NIST open a competition for the standardization of the digital signature algorithm in the post quantum stage. This publication focuses on algorithms based on the use a hash function. Their main advantage is that they rely on simple assumptions on hash functions, such as collision or second-preimage resistance, instead of a specific algebraic structure. In particular, if the attack is detect in a hash function, one can replace it by another function without modifying the overall structure of the scheme. Most hash-based schemes also come with relatively simple proofs of security reductions to the hash function's properties. Their main drawback is signature size, which typically grows with the number of messages signed by a key pair [2-5].

A significant part of the research is focused on increasing of their efficiency. Besides, the simplest hash-based schemes are stateful, which means that a signer must maintain a state that is modify every time a signature is issued. This requirement can be a burden because trivial forgeries become possible if it is violated once, e.g. if two signatures are issued in the same state. Stateful schemes must therefore guarantee that this kind of misuse will not happen, which can be non-trivial for practical systems. In theory, somebody can rolling back the state of a machine after a crash, cloning virtual machines, or maintaining a pool of signing machines working in parallel. Hence, it is advisable to use a stateless scheme GRAVITY [6].

Table 1

Security level of the applied algorithms

Algorithm	**Key length**	**Security level**	
		Classical computer	**Quantum computer**
RSA-1024	1024 b	80 b	0 b
RSA-2048	2048 b	112 b	0 b
ECC-256	256 b	128 b	0 b
ECC-384	384 b	256 b	0 b

2. Stateless algoruthm

GRAVITY scheme have three stage: key generation, signature, verification and batch signature and verification.

An instance of the GRAVITY signature scheme requires the following parameters: the hash output bit length n, a positive integer; the Winternitz [7] depth w, a power of two such that w>=2; the PORS [6] set size t, a positive power of two; the PORS subset size k, a positive integer such that k<=t; the internal Merkle [7] tree height h, a positive integer; the number of internal Merkle [7] trees d, a non-negative integer; the cache height c, a non-negative integer; the batching height b, a non-negative integer; the message space M, usually a subset of bit string {0,1}*. From this parameters are derive the follow values [5]:

– The Winternitz width l

$$l=\mu+\lfloor \log_2(\mu(w-1))/\log_2 w \rfloor+1 \text{ where } \mu=n/\log_2 w \quad (1)$$

– The PORS set $T=\{0,..., t-1\}$.

– The address space A

$$A=\{0,\ldots,d\}\times\{0,...,2^{c+dh}-1\}\times\{0,...,\max(l,t)-1\} \quad (2)$$

– the public key space $PK=B_n$.

– The secret key space $SK=B_n^2$.

– The signature space SG

$$SG=B\times B^k\times B^{k(\log_2 t - |\log_2 k|)}\times (B^l\times B^l)^d\times B^c \quad (3)$$

– The batched signature space SG_B

$$SG_B=B_n^b\times\{0,\ldots,2^b-1\}\times SG \quad (4)$$

– The public key size, of n bits.

– The secret key size, of 2n bits.

– The maximal signature size, of
sigsz = $(1 + k + k(\log_2 t - \lfloor \log_2 k \rfloor) + d(l + h) + +c)n$ bits.

– The maximal batched signature size, of
sigsz + bn + b bits.

2.1 Primitives

GRAVITY signature scheme based on next primitives that depend on scheme parameters [6]:

– a length-preserving hash function $F : B_n \rightarrow B_n$

– a length-halving hash function $H : B_n^2 \rightarrow B_n$

– a pseudo-random function $G : B_n \times A \rightarrow B_n$

– a general-purpose hash function $H^* : M \rightarrow B_n$

2.2 Key generation

Key generation takes as input 2n bits of random numbers and outputs the secret key and public key.

– Generate the secret key from 2n bits of random numbers and put to by address (seed, salt) in the Merle tree

$$sk = (\text{seed}, \text{salt}) \leftarrow B_n^2 \quad (5)$$

– For each i that $0 \le i < 2^{c+h}$ generated a Winternitz public key(WOTS-genpk) and generate next to new address for keys pair (make-addr(0, i))

$$p_i \leftarrow \text{WOTS-genpk}(\text{seed}, \text{make-addr}(0, i)) \quad (6)$$

– Generate the public key (root of Merkle-tree) where x is array of hashes leaf

$$pk \leftarrow \text{Merkle-root}_{c+h}(x_0, \ldots, x_{2^{c+h}-1}) \quad (7)$$

2.3 Signature

Message takes as input a *m* hash and secret key
sk = (seed, salt) and outputs a signature computed as follow [6]:

– Compute the public salt s ← H(salt, m).

– Compute the hyper-tree index and random subset as j, $(x_1, \ldots, x_k) \leftarrow$ PORS(s, m).

– Compute the PORST [6] signature and public key where oct is a parameter of authentication path[6]

$$(\sigma_d, \text{oct}, p) \leftarrow \text{PORST-sign}(\text{seed}, \text{make-addr}(d, j), x_1, \ldots, x_k) \quad (8)$$

For $i \in \{d-1, \ldots, 0\}$ do the following:

– Compute the WOTS (Winternitz one time signature) [7] signature

$$\sigma_i \leftarrow \text{WOTS-sign}(\text{seed}, \text{make-addr}(i, j), p) \quad (9)$$

– compute p ← WOTS-extractpk(p, σ_i).

– Set $j' \leftarrow \lfloor j/2^h \rfloor$.

– for $u \in \{0, \ldots, 2^h-1\}$compute the WOTS public key (WOTS-genpk)

$$p_u \leftarrow \text{WOTS-genpk}(\text{seed}, \text{make-addr}(i, 2^h j + u)) \quad (10)$$

– Compute the Merkle authentication path (Merkle-auth_h)

$$A_i \leftarrow \text{Merkle-auth}_h(p_0, \ldots, p_{2^h-1}, j - 2^h j) \quad (11)$$

– set $j \leftarrow j'$.

– For $0 \le u < 2^{c+h}$ compute the WOTS public key

$$p_u \leftarrow \text{WOTS-genpk}(\text{seed}, \text{make-addr}(0, u)) \quad (12)$$

– Compute the Merkle authentication

$$(a_1, \ldots, a_{h+c}) \leftarrow \text{Merkle-auth}_{h+c}(p_0, \ldots, p_{2^{h+c}-1}, 2^h j) \quad (13)$$

– Set $A \leftarrow (a_{h+1}, \ldots, a_{h+c})$.

– The signature is

$$(s, \sigma_d, \text{oct}, \sigma_{d-1}, A_{d-1}, \ldots, \sigma_0, A_0, A) \quad (14)$$

2.4 Verification

Verification function takes a hash message, public key and a signature $(s,\sigma_d,\text{oct},\sigma_{d-1},A_{d-1},\ldots,\sigma_0,A_0,A)$ and verifies it as follows[6]:

– Compute the hyper-tree index and random subset as j, $(x_1 \ldots x_k) \leftarrow$ PORS(s, m).

– Compute the PORST public key

$$p \leftarrow \text{PORST-extract pk}(x_1 \ldots x_k, \sigma_d, \text{oct}) \quad (15)$$

– If p = ⊥, then abort and return 0.

For $i \in \{d-1, \ldots, 0\}$ do the following:

– Compute the WOTS public key p ← WOTS-extract pk(p, σ_i).

– Set $j' \leftarrow \lfloor j/2^h \rfloor$.

– Compute the Merkle root

$$p \leftarrow \text{Merkle-extract}_h(p, j - 2^{h_j'}, A_i) \quad (16)$$

– set $j \leftarrow j'$.
– Compute the Merkle root

$$p \leftarrow \text{Merkle-extract}_c(p, j, A) \quad (17)$$

– The result is 1 if p = pk, and 0 otherwise.

2.5 Batch signature

Batch signature takes as input a sequence of messages $(M_1,...,M_i) \in M^i$ with $0 < i \leq 2^b$ and a secret key
sk = (seed,salt) along with its secret cache, and outputs *i* signatures σ_j, computed as follows[5]:

– For $j \in \{1,...,i\}$ compute the message digest $m_j \leftarrow H^*(M_j)$.
– For $j \in \{i+1,\ldots,2^b\}$ set $m_j \leftarrow m_1$.
– Compute $m \leftarrow \text{Merkle-root}_b(m_1, \ldots, m_{2^b})$.
– Compute $\sigma \leftarrow S(sk, m)$.
– For $j \in \{1,...,i\}$ the j-th signature is $\sigma_j \leftarrow (j,\text{Merkle-auth}_b(m_1,...,m_{2^b},j),\sigma)$.

For b = 0, we simplify $S_B(sk,M)$ to $S(sk,H^*(M))$.

2.6 Batch verification

Batch verification function V takes as input a public key pk, a message $M \in M$ and a signature (j, A, σ), and verifies it as follows [6]:

– Compute the message digest $m \leftarrow H^*(M)$.
– Compute the Merkle root.
$m \leftarrow \text{Merkle-extract}_b(m, j, A)$.
– The result is V (pk, m, σ).

For b = 0, we simplify $V_B(pk,M,\sigma)$ to $V(pk,H^*(M),\sigma)$.

3. Security proofs

The security of hash based signature depend on using in this scheme hash function. In this article suggests using DSTU 7564:2014 [7].

Ukrainian national hashing function standard is capable of operation with the hash value lengths of 256/384/512 bits. The above hash function base on the Rijndael structure. The DSTU 7564:2014 cryptographic strength is provide in Table II [7].

Table 2

Strength against the cryptographic attacks

Type of attack	**Kupyna – 256**
Collision	2^{128}
Preimage	2^{256}
Second preimage	2^{256}
Fixed point	2^{256}
Increase of the length	2^{256}

4. Development cycle

In this project, propose to use Continuous integration/Continuous delivery (CI/CD) [9] software development practice. It methodology consist of a continuous cycle of plan, code, build, test, release, deploy, operate, monitor. Continuous integration focuses on blending the work products of individual developers together into a repository. Often, this is done several times each day, and the primary purpose is to enable early detection of integration bugs, which should eventually result in tighter cohesion and more development collaboration. The aim of continuous delivery is to minimize the friction points that are inherent in the deployment or release processes. Typically, the implementation involves automation of each of the steps for build deployments such that a safe code release can be done—ideally—at any moment in time.

For development applications propose is use CI/CD tools, in particular, Jenkins, Gitlab on Amazon web service (Fig.1) [9].

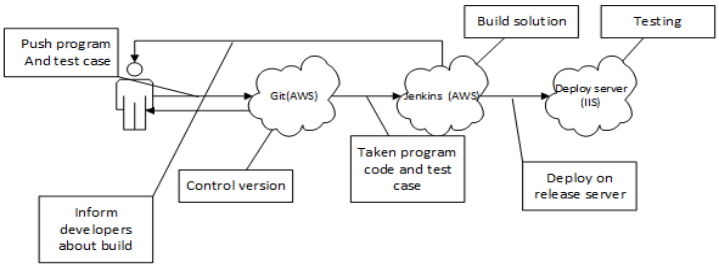


Figure 1. Develop application cycle

Public-key infrastructure consist of three level: web application, desktop application and hardware application (Fig. 2).

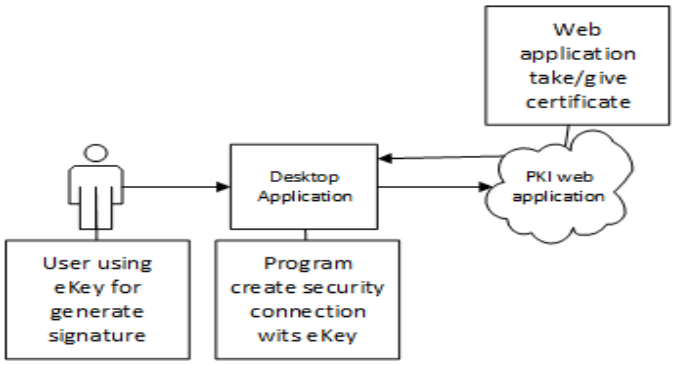


Figure 2. Architecture applications

The web application is being built using C# languages and ASP .NET MVC technology, in turn, desktop application is being built using C# as well and Windows Form technology, hardware application in eKey is being built using C language. Desktop application is create security channel with eKey and transfer data for generation key pair, singing and verification documents.

5. Conclusion

In this article, describe stateless algorithm GRAVITY [6]. These schemes rely on a limited number of assumptions that form collision-resistant, one-way, undetectable and pseudo-random function families. This means that their security is relatively well-understood, even against hypothetical adversaries with a quantum computer. Besides, the stateless property gives some “misuse-resistance” guarantees for signers that cannot reliably maintain a state over the lifetime of a key

pair. Another advantage of hash-based signatures is speed and simplicity of verification. On the signer side, we have seen that several trade-offs are available of signature size, computational resources, memory resources, as well as the planned number of signatures issued by a key pair. Development of group application show how to work public-key infrastructure with using stateless quantum-resistant algorithm.

References:

1. ETSI GR QSC 001 V.1.1.1 (2016-07). Quantum-Safe Cryptography (QSC); Quantum-safe algorithmic framework.
2. NIST PQC workshop: SAFEcrypto Project, M. O'Niell. 2015.
3. NIST Workshop on Cyber Security in a Post-Quantum World (2015). PQCrypto project, T Lange.
4. PQCrypto. Initial recommendation of Long-term secure post-quantum systems.
5. Endignoux G. Design and implementation of a post-quantum hash-based cryptographic signature scheme: mater's thesis:july 2017 /Guillaume Endignoux – Switzerland,2017. 42 p.
6. Gravity-Sphincs : Kudelski Security : 25 p./Jean-Philippe Aumasson, Guillaume Endignoux.
7. Becjer G. Merkle signature schemes, Merkle Trees and Their Cryptanalusis: 2013.
8. Gorbenko Yu.I., (2015), Construction and analysis of systems, protocols, and methods of cryptographic information protection. Pt. 1, Methods of construction and analysis, standardization and application of cryptographic systems, Kharkiv, Ukraine: Fort, (in Ukrainian).
9. Ellingwood. J. CI/CD tools comparison: Jenkins,Gitlab CI etc.[Internet resource]/ Justin Ellingwood. [Access mode]: https://www.digitalocean.com/community/tutorials/ci-cd-tools-comparison-jenkins-gitlab-ci-buildbot-drone-and-concourse. 02.02.2017.

Национальный аэрокосмический
университет имени Н.Е. Жуковского «ХАИ»

Поступила в редколлегию 27.03.2018

УДК 621.391

І.Д. ГОРБЕНКО, д-р техн. наук, О.А. ЗАМУЛА, д-р техн. наук

ДОСЛІДЖЕННЯ СТРУКТУРИ СПЕКТРІВ СИГНАЛОВ З ЛІНІЙНОЮ ЧАСТОТНОЮ МОДУЛЯЦІЄЮ

Вступ

Для ряду додатків інформаційно-комунікаційних систем (ІКС) пред'являються досить жорсткі вимоги щодо забезпечення ефективності їх функціонування в умовах складних зовнішніх і внутрішніх впливів, обумовлених природними і навмисними перешкодами; перешкодами від інших радіотехнічних систем, що функціонують на близьких частотах або в спільній ділянці діапазону частот; спробами проведення криптографічного аналізу та порушення цілісності (автентичності) даних користувачів і ін. До основних показників ефективності функціонування ІКС відносять: пропускну здатність мережі, завадозахищеність, продуктивність, інформаційну безпеку, скритність функціонування, живучість, своєчасність доставки повідомлень і ін. Численні дослідження показали, що поліпшення якісних показників, зокрема завадозахищеності та інформаційної безпеки ІКС і мереж, може бути досягнуто, в тому числі, шляхом розробки методів синтезу, формування і обробки складних дискретних сигналів – фізичних переносників даних з необхідними ансамблевими, структурними і кореляційними властивостями [1 – 6]. Ряд досліджень [7 – 9] свідчать, що подальше поліпшення основних якісних показників деяких додатків радіоканалів може бути досягнуто на основі використання сигналів з лінійною частотною модуляцією (ЛЧМ), ФМ ЛЧМ і в загальному випадку складових нерівномірних по тривалості ЛЧМ сигналів з внутрішньоімпульсною ФМ (СНЛЧМ-ФМ) сигналів. Особливий інтерес у зв'язку з малою дослідженністю спектральних, кореляційних, ансамблевих і структурних властивостей представляють ФМ ЛЧМ і СНЛЧМ сигнали. Також не досліджено властивості складових як рівномірних, так і нерівномірних ЛЧМ-ФМ сигналів. У зв'язку із зазначеним представляється актуальним дослідження спектральних, кореляційних, ансамблевих і структурних властивостей ФМ-ЛЧМ і СНЛЧМ-ФМ сигналів.

Основні результати досліджень

Аналітичне представлення СНЛЧМ-ФМ сигнали має вигляд

$$S^{(p)}(t) = S_0{}^{(p)} \sum_{n=1}^{N} \sum_{l=1}^{Q} V_e^{(p)} \operatorname{rect}\left(\frac{t - \sum_{r=0}^{n-1} T_r}{T_n} \right) \operatorname{rect}\left(\frac{t - (l-1)\tau_э}{\tau_э} \right) \times$$

$$\times \exp\left(j\left(\omega_n \left(t - \sum_{r=0}^{n-1} T_r \right) + \frac{\mu_n}{2} \left(t - \sum_{r=0}^{n-1} T_r \right)^2 + \varphi_n \right) \right), \qquad (1)$$

де $S_0{}^{(p)}$ – амплітуда огибаючої сигналу, N – число радіоімпульсів, що складають СНЛЧМ-ФМ сигнал; Q – число елементів двійкової маніпулюючої послідовності; $V_e^{(p)}$ – символ p-й маніпулюючої послідовності, причому $V_l^{(p)} \in \{1, -1\}$; $\operatorname{rect}(x)$ – є функція, що має вигляд $\operatorname{rect}(x) = \begin{cases} 1 \text{ при } 0 \le x \le 1, \\ 0 \text{ при } x < 0, x > 1 \end{cases}$; $\tau_э$ – тривалість елемента маніпулючої послідовності; ω_n, φ_n – несуча частота і початкова фаза n-го ЛЧМ радіоімпульсу; μ_n – коефіцієнт нахилу маніпулюючої характеристики n-го ЛЧМ радіоімпульсу, що пов'язаний з девіацією частоти ΔF_n і тривалістю T_n співвідношенням $\mu_n = \pm 2\pi \Delta F_n / T_n$.

Очевидно, що $\sum_{n=1}^{N} T_n = Q\tau_э$ і $\sum_{N=0}^{n-1} T_r = 0$ при n=1. Вираз для спектра $S(\omega)$ отримаємо, взявши перетворення Фур'є від (1).

Представимо комплексний спектр СНЛЧМ-ФМ сигналу у вигляді чотирьох компонент: амплітудного спектра, квадратичного, залишкового і додаткового фазового членів.
Амплітудний спектр має вигляд

$$\left|S(\omega)\right| = S_0^{(p)}\left(\sum_{n=1}^{N}\sqrt{\frac{I}{\mu_n}}\left(\sum_{l=a+2}^{b}\left(C\left(\chi_2^{(nl)}\right)-C\left(\chi_1^{(nl)}\right)\right)+V_{a+1}^{(p)}\left(C\left(\chi_1^{(nl)}\right)-C\left(\chi_3^{(nl)}\right)\right)-V_{b+1}^{(p)}\left(C\left(\chi_6^{(nl)}\right)-C\left(\chi_5^{(nl)}\right)\right)\right)\right)^2 +$$

$$+\left(\sum_{n=1}^{N}\sqrt{\frac{\pi}{\mu_n}}\left(\sum_{l=a+2}^{b}V_l^{(p)}\left(S\left(x_2^{(nl)}\right)-S\left(x_1^{(nl)}\right)\right)+V_{a+1}^{(p)}\left(S\left(x_4^{(nl)}\right)-S\left(x_3^{(nl)}\right)\right)=V_{b+1}^{(p)}\left(S\left(x_6^{(nl)}\right)-S\left(x_5^{(nl)}\right)\right)\right)\right)^2)^{\frac{1}{2}} \quad (2)$$

Квадратичний фазовий член:

$$\Phi_1(\omega) = \sum_{n=1}^{N}\frac{(\omega_n-\omega)^2}{2\mu_n} \quad (3)$$

Остаточний фазовий член:

$$\Phi_2(\omega) = \operatorname{arctg}\left(\sum_{n=1}^{N}\sqrt{\frac{\pi}{\mu_n}}\left(\sum_{l=a+2}^{b}V_l^{(p)}\left(S\left(x_2^{(nl)}\right)-S\left(x_1^{(nl)}\right)\right)+V_{a+1}^{(p)}\left(S\left(x_4^{(nl)}\right)-S\left(x_3^{(nl)}\right)\right)-V_{b+1}^{(p)}\left(S\left(x_6^{(nl)}\right)-S\left(x_5^{(nl)}\right)\right)\right)\right)/$$

$$/\left(\sum_{n=1}^{N}\sqrt{\frac{\pi}{\mu_n}}\left(\sum_{l=a+2}^{b}V_l^{(p)}\left(C\left(x_2^{(nl)}\right)-C\left(x_1^{(nl)}\right)\right)+V_{a+1}^{(p)}\left(C\left(x_4^{(nl)}\right)-C\left(x_3^{(nl)}\right)\right)+V_{b+1}^{(p)}\left(C\left(x_6^{(nl)}\right)-C\left(x_5^{(nl)}\right)\right)\right)\right) \quad (4)$$

Додатковий фазовий член:

$$\Phi_3(\omega) = \sum_{n=1}^{N}\left(\omega\sum_{r=0}^{n-1}T_r - \varphi_n\right), \quad (5)$$

де C(x) и S(x) – інтеграли Френеля;

$$x_1^{(nl)} = \frac{\mu_n\left((l-1)\tau_э - \sum_{r=0}^{n-1}T_r\right)+\omega_n-\omega}{\sqrt{\mu_n\pi}}; \qquad x_2^{(nl)} = \frac{\mu_n\left(l\tau_э - \sum_{r=0}^{n-1}T_r\right)+\omega_n-\omega}{\sqrt{\mu_n\pi}};$$

$$x_3^{(nl)} = \frac{\omega_n-\omega}{\sqrt{\mu_n\pi}}; \qquad x_4^{(nl)} = \frac{\mu_n\left((a+1)\tau_э - \sum_{r=0}^{n-1}T_r\right)+\omega_n-\omega}{\sqrt{\mu_n\pi}};$$

$$x_5^{(nl)} = \frac{\mu_n\left(b\tau_э - \sum_{r=0}^{n-1}T_r\right)+\omega_n-\omega}{\sqrt{\mu_n\pi}}; \qquad x_6^{(nl)} = \frac{\mu_n T_n+\omega_n-\omega}{\sqrt{\mu_n\pi}}.$$

В окремому випадку, при однаковій тривалості складних елементів ЛЧМ-ФМ сигналу, вирази (2) – (4) мають вигляд відповідно:

$$S(\omega) = \frac{S_0}{2\sqrt{NT}}\sum_{n=0}^{N-1}\sum_{l=0}^{Q-1}V_{Qn+l+1}^{(p)}\sqrt{\frac{\pi}{\mu_n}}\exp\left(j\left(\varphi_n-\omega_n T-\frac{(\omega_n-\omega)^2}{2\mu_n}\int_{x_1}^{x_2}\exp\left(j\left(\frac{\pi x^2}{2}\right)\right)dx\right)\right); \quad (6)$$

$$\Phi_1(\omega)=\sum_{n=0}^{N-1}\left(\varphi_n-\omega_n T-\frac{(\omega_n-\omega)^2}{2\mu}\right); \tag{7}$$

$$\Phi_2(\omega)=\operatorname{arctg}\sum_{n=0}^{N-1}\sum_{l=0}^{Q-1}\frac{V_{Q(n-1)+l}\left(S(x_2)-S(x_1)\right)}{V_{Q(n-1)+l}\left(C(x_2)-C(x_1)\right)}. \tag{8}$$

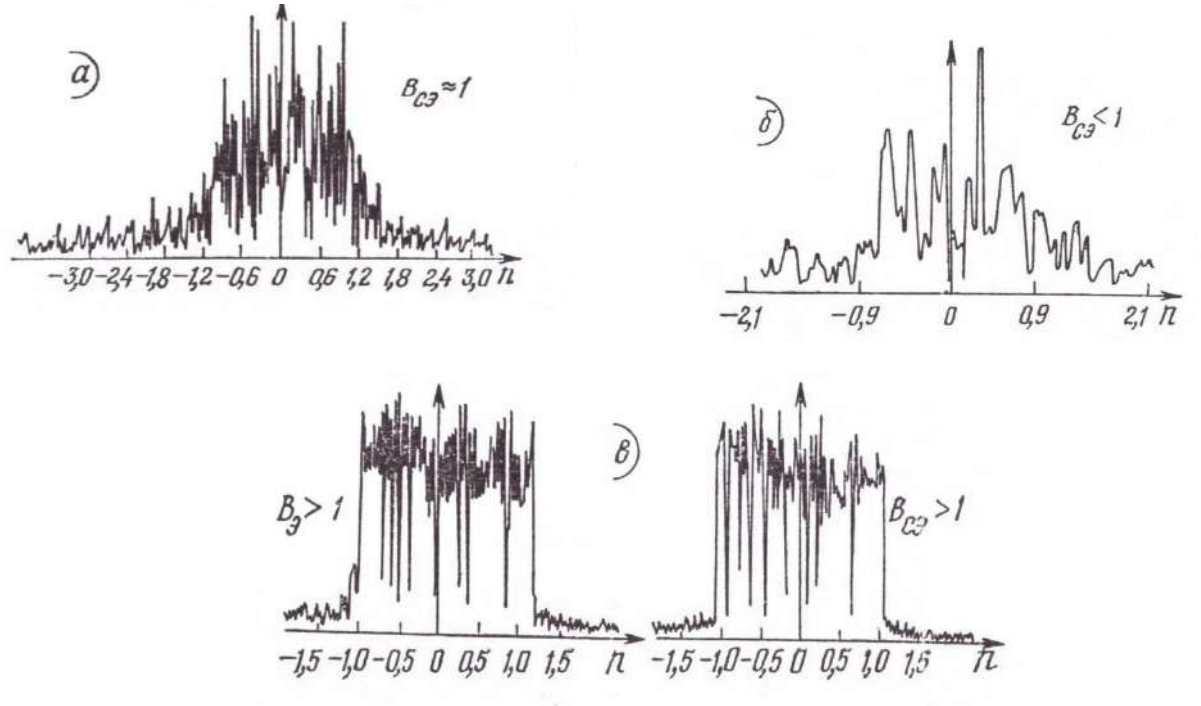
Розглянемо окремий випадок виразів (6) – (8) для ФМ ЛЧМ і ЛЧМ сигналу. Для ФМ ЛЧМ сигналу з (6) отримаємо

$$\left|S(\omega)\right|=\frac{S_0}{2\sqrt{T}}\sum_{l=0}^{Q-1}V_{l+1}{}^{p}\sqrt{\frac{\pi}{\mu}}\left(\left(-C(x_1)+C(x_2)\right)^2+\left(-S(x_1)+S(x_2)\right)^2\right), \tag{9}$$

що співпадає з виразом для амплітудного спектру ФМ ЛЧМ сигналу [10]. Аналогічно для ЛЧМ сигналу

$$\left|S(\omega)\right|=\frac{S_0}{2\sqrt{T}}\sqrt{\frac{\pi}{\mu}}\left(\left(-C(x_1)+C(x_2)\right)^2+\left(-S(x_1)+S(x_2)\right)^2\right), \tag{10}$$

що співпадає з виразом для ЛЧМ сигналу [10]. Аналогічно можна отримати вираз для квадратичного і залишкового фазових членів. З (2) – (4) випливає, що фазова модуляція ЛЧМ радіоімпульсу проявляється в зміні амплітудного спектра і залишкового фазового члена. Результати проведених розрахунків (2) – (4) дозволяють простежити, яким чином трансформується спектр ЛЧМ сигналу при кодуванні його внутрішньої фази за законом псевдовипадкових двійкових послідовностей. На рисунку показано амплітудні спектри ЛЧМ ФМ сигналів з різними співвідношеннями бази ЛЧМ радіоімпульсу і бази (числа елементів) кодуючої послідовності:



Як випливає з рисунку, при модуляції ЛЧМ радіоімпульсу по фазі кодуючою послідовністю спостерігається розширення спектра результуючого сигналу, причому спектр в межах ширини смуги сигналу набуває порізаний вид. У разі, коли база складового елементу сигналу більше одиниці ($B_{ce}>1$), в результуючому спектрі простежується структура кодуючої послідовності, тобто результуючий амплітудний спектр ЛЧМ-ФМ сигналу збігається

з сумою амплітудних спектрів складових його елементів. При цьому в «місцях зшивання» протифазних елементів спостерігаються «провали», що виникають в результаті накладання спектрів з протифазними компонентами і, таким чином (при $B_{ce}>1$), стає можливим визначити структуру сигналу по виду його амплітудного спектра. При малій базі складового елемента результуючий спектр стає нерегулярним, що є кращим для інформаційно-комунікаційних систем, в яких вимога скритності інформаційного обміну є пріоритетною. Аналіз виразу (4), який визначає залишковий фазовий член, показує, що в основній області частот він має сильно порізаний вид, при цьому, коли база складового елементу сигналу B_{ce} значно більше одиниці, в формі залишкового фазового члена проявляється структура кодуючої послідовності. Таким чином, при $B_{ce}>1$ структуру сигналу можна визначити не тільки за формою амплітудного спектра, а й за видом залишкового фазового члена. При малих значеннях бази складового елементу сигналу закон зміни залишкового фазового члена має нерегулярний характер.

Функцію невизначеності (ФН) СНЛЧМ ФМ знайдемо, використовуючи співвідношення [8]:

$$X^{(p,q)}(\tau_з,\omega_д)=\int_{-\infty}^{+\infty} S(t)S^*(t-\tau_з)e^{j\omega_д t}dt\,, \tag{11}$$

де $\tau_з$ – затримка сигналу за часом; $\omega_д$ – зміщення за частотою; * – символ комплексної спряженості.

Представимо затримку у вигляді

$$\tau_з=\sum_{z=0}^{\alpha}T_z+k\tau_э+\Theta\,, \tag{12}$$

де α – число, що показує на скільки ЛЧМ радіоімпульсів сигнал q затриманий щодо сигналу p; k – число, що показує, на скільки двійкових елементів в радіоімпульс сигнал q затриманий щодо сигналу p; Θ – затримка, величина якої задовольняє умові $0\le\Theta\le\tau_э$.

Отримаємо вирази для ФН СНЛЧМ-ФМ сигналу для випадку, коли $\mu_{n+\alpha}\ne\mu_n$. При такому співвідношенні коефіцієнтів нахилу маємо

$$X_{\text{I}}^{(p,q)}(\tau_з,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}\sum_{l=a+1}^{c}N_{l+k}{}^{q}V_l^{(p)}A\left(C_1+jR_1\right) \tag{13}$$

де $a=\text{entier}\left(\dfrac{\sum_{r=0}^{n}T_r+\tau_э}{\tau_э}\right)$, $c=\text{entier}\left(\dfrac{\sum_{r=0}^{n}T_r-\tau_з}{\tau_э}\right)$;

$$A=\sqrt{\frac{\pi}{\mu_n-\mu_{n+\alpha}}}\exp\Bigg(j\Bigg(\left(\omega_{n+\alpha}-\omega_n\right)\left(\sum_{r=0}^{n-1}T_r+\sum_{z=0}^{\alpha}T_z\right)+\frac{\mu_n}{2}\left(\sum_{r=0}^{n-1}T_r+\sum_{z=0}^{\alpha}T_z\right)^2+$$

$$+\omega_{n+\alpha}\left(k\tau_э+\Theta\right)-\mu_{n+\alpha}\left(\sum_{r=0}^{n-1}T_r+\sum_{z=0}^{\alpha}T_z+k\tau_э+\Theta\right)^2+\left(\varphi_n-\varphi_{n+\alpha}\right)-$$

$$-\frac{1}{2}\left(\omega_{n+\alpha}-\omega_n-\omega_д+\left(\mu_n-\mu_{n+\alpha}\right)\left(\sum_{r=0}^{n-1}T_r+\sum_{z=0}^{\alpha}T_z\right)-\mu_{n+\alpha}\left(kT_э+\Theta\right)\right)^2\Bigg)\Bigg)/\left(\mu_n-\mu_{n+\alpha}\right);$$

$$C_i=C\left(x_{2i}{}^{(nl)}\right)-C\left(x_{2i-1}{}^{(nl)}\right);\quad R_i=S\left(x_{2i}{}^{(nl)}\right)-S\left(x_{2i-1}{}^{(nl)}\right),$$

де i – номер доданку;

$$X_{\text{II}}^{(p,q)}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}\sum^{c}V_c^{c(p)}A(G_2+jR_2);\tag{14}$$

$$X_{\text{III}}^{(p,q)}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}V_{c+k}^{(q)}V_{c+1}{}^{(p)}A(G_3+jR_3);\tag{15}$$

$$X_{\text{IV}}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}V_{c+k}^{q}V_{c+1}{}^{(p)}A(G_4+jR_4);\tag{16}$$

$$X_{\text{V}}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}V_{c+l+1}^{q}V_{c+1}{}^{p}A(G_5+jR_5);\tag{17}$$

$$X_{\text{VI}}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}\sum_{l=c+2}^{b}V_{l-k}^{(q)}V_l^{(p)}A(G_6+jR_6)\tag{18}$$

$$X_{\text{VII}}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}\sum_{l=c+2}^{b}V_{l+k+1}^{q}V_l^{(p)}A(G_7+jR_7)\tag{19}$$

$$X_{\text{VIII}}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}V_{l+k+1}^{q}V_{b+1}{}^{(p)}A(G_8+jR_8)\tag{20}$$

$$X_{\text{IX}}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}V_{l+1+1}^{q}V_{b+1}{}^{p}A(G_9+jR_9)\tag{21}$$

$$X_{\text{X}}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}\sum_{n=1}^{N-\alpha}V_{b+k+1}^{q}V_{b+1}{}^{(p)}A(G_{10}+jR_{10})\tag{22}$$

$$X_{\text{XI}}(\tau_э,\omega_д)=S_0^{(q)}S_0^{(p)}V_1^{(q)}V_{k+1}{}^{p}A(G_{11}+jR_{11}),\tag{23}$$

де $X_1=\left[\mu_1(l-1)\tau_э-\omega_1\right]/\mu_2;\ \mu_1=\mu-\mu_{n+\alpha};$

$$\omega_1=\left[\omega_{n+\alpha}-\omega_n-\omega_д+(\mu_n+\mu_{n+\alpha})\left(\sum_{r=0}^{n-1}T_r\sum_{z=0}^{\alpha}T_z\right)-\mu_{n+\alpha}(k\tau_д+\Theta)\right];\ \mu_n=\sqrt{\frac{2\pi}{\mu_r-\mu_{n+\alpha}}};$$

$$X_2^{(nl)}=X_3^{(nl)}=\left[\mu_1(l\tau_э-\Theta)-\omega_1\right]/\mu_2;\ \ X_4^{(nl)}=X_5^{(nl)}=(\mu_1 l\tau_э-\omega_1)/\mu_2;$$

$$X_6^{nl}=X_7^{nl}=\left[\mu_1\left(\sum_{r=0}^{n}T_r-\sum_{z=0}^{\alpha}T_z-k\tau_э-\Theta\right)-\omega_1\right]\Big/\mu_2;$$

$$X_8^{(nl)}=X_9^{(nl)}=\left\{\mu_1\left[(c+1)\tau_э-\Theta\right]-\omega_1\right\}/\mu_2;\qquad X_{10}^{nl}=\left[\mu_1(c+1)\tau_э-\omega_1\right]/\mu_2;$$

$$X_{11}^{nl}=\left[\mu_1(l-1)\tau_э-\omega_1\right]/\mu_2;\qquad X_{12}^{nl}=\left[\mu_1(l\tau_э-\Theta)-\omega_1\right]/\mu_2;$$

$$X_{13}^{nl}=X_{14}^{nl}=(\mu_1 b\tau_э-\omega_1)/\mu_2;\qquad X_{15}^{nl}=(\mu_1 b\tau_э-\omega_1)/\mu_2;$$

$$X_{16}^{nl}=X_{17}^{nl}=\left(\mu_1\sum_{r=0}^{n}T_r-\omega_1\right)\Big/\mu_2;\qquad X_{18}^{nl}=X_{19}^{nl}=\left\{\mu_1\left[(b+1)\tau_э-\Theta\right]-\omega_1\right\}/\mu_2;$$

$$X_{20}^{nl}=\left[\mu_1(b+1)\tau_э-\omega_1\right]/\mu_2;\qquad X_{21}=\omega_1/\mu_2;\ \ X_{22}=\left[\mu_1(\tau_э-\Theta)-\omega_1\right]/\mu_2.$$

В окремому випадку при однаковій тривалості ФМЛЧМ складних елементів, що становлять суми аперіодичних ФН рівномірних ЛЧМ – ФМ сигналів, доданки мають вигляд:

$$X_I(\tau_э,f_д)=\frac{S_0^2}{N^T}\sum_{m=0}^{N-1-\alpha}\sum_{S=0}^{Q-1-r}(V_{R(m+\alpha)+r+s+1}^{p}V_{Qm+s+1}^{q}\cos\left((\varphi_{m+\alpha}-\varphi_n)+2\pi f_m(r\tau_э+\Theta)-\pi k_m(r\tau_э+\Theta)^2+2\pi(m+\alpha)Tf_д+\right.$$

$$+2\pi\left(\mu_m\left(r\tau_э+\Theta\right)+f_д+f_{m+d}-f_m\right)\cdot\left(\left(r+s\right)\tau_э+\frac{\tau_э+\Theta}{2}\right)\cdot\frac{\sin\left(\pi\left(k_m\left(r\tau_э+\Theta\right)+f_д+f_{m+\alpha}-f_m\right)\left(\tau_э-\Theta\right)\right)}{\pi\left(\mu_m\left(\tau_э r+\Theta\right)+f_д+f_{m+\alpha}-f_m\right)}) \quad (24)$$

$$X_{II}\left(\tau,f_д\right)=\frac{S_0^2}{NT}\sum_{m=0}^{N-1-\alpha}\sum_{S=0}^{Q-r-2}(V^p_{Q(m+\alpha)+r+s+2}V^q_{Qm+s+1}\cos\left(\varphi_{m+\alpha}-\varphi_m\right)+2\pi f_m\left(r\tau_э+\Theta\right)-\pi\mu_m\left(r\tau_э+\Theta\right)^2-$$

$$-2\pi\left(m+\alpha\right)Tf_д+2\pi\left(\mu_m\left(r\tau_э+\Theta-T\right)+f_д+f_{m+\alpha}-f_m\right)\cdot\left(\left(r+s+1\right)\tau_э+\frac{\Theta}{2}\right)\cdot\frac{\sin\left(\pi\left(\mu_N\left(r\tau_э+\Theta\right)+f_д+f_{m+\alpha}-f_m\right)\Theta\right)}{\pi\left(\mu_m\left(\tau_э r+\Theta\right)+f_д+f_{m-\alpha}-f_m\right)}) \quad (25)$$

$$X_{III}\left(\tau,f_д\right)=\frac{S_0^2}{N^T}\sum_{m=0}^{N-2-\alpha}\sum_{S=Q-r}^{Q-1}(V^p_{Q(m+\alpha)+r+s+1}V^q_{Qm+s+1}\cos((\varphi_{m+\alpha-1}-\varphi_m)+2\pi f_m\left(\Theta+2\tau-T\right)-\pi\mu_m\left(r\tau_э+\Theta-T\right)^2+$$

$$+2\pi\left(m+\alpha+1\right)Tf_э+2\pi\left(\mu_m\left(r\tau_э+\Theta-T\right)+f_д+f_{m+\alpha+1}-f_m\right)\cdot\left(\left(r+s-\Theta\right)\tau_э+\frac{\tau_э+\Theta}{2}\right)\cdot\frac{\sin\left(\pi\left(\mu_m\left(r\tau_э+\Theta-T\right)+f_д+f_{m+\alpha+1}-f_m\right)\left(\tau_э-\Theta\right)\right)}{\pi\left(\mu_m\left(\tau_э r+\Theta-T\right)+f_д+f_{m+\alpha+1}-f_m\right)}) \quad (26)$$

$$X_{IV}\left(\tau,f_э\right)=\frac{S_0^2}{N^T}\sum_{m=0}^{N-2-\alpha}\sum_{S=\Theta-r-1}^{Q-1}(V^p_{Q(m+\alpha)+r+s+2}V^q_{Qm+s+1}\cos((\varphi_{m+\alpha-1}-\varphi_m)+2\pi f_m\left(r\tau_э+\Theta-T\right)-\pi\mu_m\left(r\tau_э-\Theta-T\right)^2+$$

$$+2\pi\left(m+\alpha+1\right)Tf_э+2\pi\left(\mu_m\left(r\tau_э+\Theta-T\right)+f_д-f_{m+\alpha+1}-f_m\right)\cdot\left(\left(r+s+1-\Theta\right)\tau_э+\frac{Q}{2}\right)\cdot\frac{\sin\left(\pi\left(\mu_m\left(r\tau_э+\Theta-T\right)+f_д+f_{m+\alpha+1}-f_m\right)\Theta\right)}{\pi\left(\mu_m\left(\tau_э r+\Theta-T\right)+f_д+f_{m+\alpha+1}-f_m\right)}).(27)$$

На наш погляд, потребують більш детальних досліджень взаємна та автокореляційна функції кореляції ФМ-ЛЧМ і СНЛЧМ-ФМ сигналів, які є відповідними зрізами функцій невизначеності зазначених сигналів. Аналіз (2) – (27) показує, що обумовлені ними амплітудні і фазові спектри можуть бути обчислені тільки з використанням комп'ютерних систем. Функції $C(x)$ і $S(x)$, які входять до виразів (2) – (27), мають слабку збіжність, тому витрати машинного часу стають практично нереалізованими навіть при невеликих значеннях N. Істотне прискорення вирішення завдань спектрального і кореляційного аналізу може бути досягнуто при використанні алгоритмів швидких перетворень в різних базисах: Фур'є, теоретико-числових перетворень, Фур'є – Винограда та ін. Аналіз отриманих з використанням (2) амплітудних спектрів складових ЛЧМ сигналів і ЛЧМ-ФМ сигналів показує, що спектр таких сигналів у порівнянні зі спектром ЛЧМ сигналів спотворюється, причому зі збільшенням різниці $T_{\max}-T_{\min}$ (де $T_{\max}$ і $T_{\min}$ – відповідно максимальні і мінімальні періоди ЛЧМ радіоімпульсів, що утворюють складений сигнал) збільшується пік-фактор спектральної щільності сигналу. Крім того, спектр зазначених сигналів при відношенні $\frac{\sum_{n=1}^{N}\left|\Delta F_n\right|T_n}{Q}\geq 1$ порізаний, проте в ньому чітко помітні переходи як від одного ЛЧМ радіоімпульсу до іншого, так і переходи модулюючого двійкового сигналу (1, 1) або (1, 1). При відношенні $\frac{\sum_{n=1}^{N}\left|\Delta F_n\right|T_n}{Q}<1$ спектр сигналу наближається до спектру шумоподібного сигналу. При цьому смуга частот, яку займає сигнал (($F_{\min}-F_{\max}$), де $F_{\max}$ і $F_{\min}$ – відповідно максимальна і мінімальна частота ЛЧМ сигналу), розширюється. Рівень бічних складових амплітудного спектра зростає зі збільшенням числа елементів двійкової маніпулючої послідовності. Порівняльний аналіз великого числа розрахунків показує, що, з точки зору ефективності використання смуги пропускання, бажано в якості маніпулюючих використовувати послідовності з одно- або дворівневою функцією автокореляції, тому що при маніпуляції фази СНЛЧМ сигнали є ортогональними сигналами і рівень бічних складових амплітудного

спектра на 7 – 10 % вище, ніж у разі маніпуляції того ж СНЛЧМ сигналу іншими класами сигналів. Крім того, у складових ЛЧМ и ЛЧМ-ФМ сигналів відбувається подавлення декількох частотних складових за рахунок розширення спектру частот окремих ЛЧМ, ЛЧМ-ФМ сигналів та інтерференції між ними.

Висновки

Дослідження показують, що кореляційні властивості СНЛЧМ сигналів при $f_g = 0$ практично збігаються з кореляційними властивостями ЛЧМ сигналів, однак при неузгодженості по частоті відбувається роздвоєння основного викиду ФН СНЛЧМ сигналів і зі збільшенням частоти неузгодженості швидкість роздвоєння збільшується, при цьому спостерігається незначне зменшення амплітуди основного викиду. При збільшенні значення $M_{\max} - M_{\min}$ (де $M_{\max}$ і $M_{\min}$ – максимальний і мінімальний коефіцієнти нахилу модуляційної характеристики ЛЧМ радіоімпульсів) спостерігається збільшення різниці амплітуд роздвоєного основного викиду. Рівень бічних пелюсток ФН ЛЧМ-ФМ сигналів при $\frac{\sum_{n=1}^{N} \Delta F_n T_n}{Q} \geq 1$ порівняно з ЛЧМ сигналами збільшується і не перевищує 14,2 дБ основного викиду в області початку координат. Таким чином, знання тонкої структури спектрів ЛЧМ, ФМ ЛЧМ, складових нерівномірних по тривалості ЛЧМ сигналів з внутрішньоімпульсною ФМ дозволяє ефективно використовувати зазначені класи сигналів в різних інформаційно-комунікаційних системах, в тому числі в системах, в яких пред'являються підвищені вимоги до скритності їх функціонування.

Список літератури:

1. *Gorbenko I.D., Zamula A.A., Semenko Ye.A.* Ensemble and correlation properties of cryptographic signals for telecommunication system and network applications // Telecommunications and Radio Engineering. 2016. Vol. 75, Issue 2. P. 169-178.

2. *Gorbenko I.D., Zamula A.A.* Cryptographic signals: requirements, methods of synthesis, properties, application in telecommunication systems // Telecommunications and Radio Engineering. 2017. Vol. 76, Issue 12. P. 1079-1100.

3. *Gorbenko I.D., Zamula A.A, Semenko A. E., Morozov V.L.* Method for synthesis of performed signals systems based on cryptographic discrete sequences of symbols // Telecommunications and Radio Engineering. 2017. Vol. 76, Issue 17. P. 1523-1533.

4. *Gorbenko I.D., Zamula A.A, Semenko A. E., Morozov V.L* Method for complex improvement of characteristics of orthogonal ensembles based on multiplicative combining of signals of different classes // Telecommunications and Radio Engineering. 2017. Vol. 76, Issue 18. P. 1581-1594 .

5. *Gorbenko I.D., Zamula A.A, Morozov V.L* Information security and noise immunity of telecommunication systems under conditions of various internal and external impacts // Telecommunications and Radio Engineering. 2017. Vol. 76, Issue 19. P. 1705-1717.

6. *Горбенко І.Д., Замула О.А.* Моделі та методи синтезу криптографічних сигналів та їх оптимізація за критерієм часової складності // Математичне та комп'ютерне моделювання. Серія: Фізико-математичні науки : зб. наук. праць / Ін-т кібернетики імені В.М. Глушкова Національної академії наук України. 2017. Вип. 15. 272 с.

7. *Скляр Б.* Цифровая связь. Теоретические основы и практическое применение ; пер. с англ. Москва : Изд. дом «Вильямс», 2003. 1104 с.

8. *Прокис Джон.* Цифровая связь ; пер. с англ. ; под ред. Д.Д. Кловского. Москва : Радио и связь. 200с.

9. *Rohling H.* OFDM Concepts for Future Communication System. Springer – Verlage Berlin Heidelberg, 2011.

10. *Кук Ч., Бернфельд М.* Радиолокационные сигналы. Теория и применение ; пер. с англ. ; под ред. В.С. Кельзона. Москва : Сов. радио, 1971. 567с.

Харківський національний
університет імені В.Н. Каразіна

Надійшла до редколегії 12.02.2018

РЕФЕРАТЫ РЕФЕРАТИ ABSTRACTS

МЕТОДЫ, МЕХАНИЗМЫ И АЛГОРИТМЫ КРИПТОГРАФИЧЕСКИХ ПЕРСПЕКТИВНЫХ ПРЕОБРАЗОВАНИЙ

МЕТОДИ, МЕХАНІЗМИ ТА АЛГОРИТМИ КРИПТОГРАФІЧНИХ ПЕРСПЕКТИВНИХ ПЕРЕТВОРЕНЬ

METHODS, MECHANISMSC OF CRYPTOGRAPHIC PERSPECTIVE TRANSFORMATIONS

УДК 004.056.55

Общие положения и анализ алгоритма направленного шифрования NTRU Prime IIT Ukraine / *И.Д. Горбенко, О.Г. Качко, М.В. Есина* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 5 – 16.

Рассмотрена NTRU-подобная криптосистема NTRU Prime IIT Ukraine, которая создана на основе существующих криптопреобразований типа «направленное шифрование». Приведены особенности ее реализации, сравнение основных характеристик и показателей, а также определение отличий от существующих сегодня NTRU-подобных криптоалгоритмов. Сделаны выводы и приведены рекомендации.

Ключевые слова: кольцо, направленное шифрование, поле, полином, фактор кольцо.

Табл. 8. Библиогр.: 9 назв.

УДК 004.056.55

Загальні положення та аналіз алгоритму направленого шифрування NTRU Prime IIT Ukraine / *І.Д. Горбенко, О.Г. Качко, М.В. Єсіна* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 5 – 16.

Розглянуто NTRU-подібну криптосистему NTRU Prime IIT Ukraine, що створена на основі існуючих криптоперетворень типу направлене шифрування. Також наведено особливості її реалізації, порівняння основних характеристик та показників, а також визначення відмінностей від існуючих сьогодні NTRU-подібних криптоалгоритмів. Зроблено висновки та наведено рекомендації.

Ключові слова: кільце, направлене шифрування, поле, поліном, фактор кільце.

Табл. 8. Бібліогр.: 9 назв.

UDC 004.056.55

General statements and analysis of the end-to-end encryption algorithm NTRU Prime IIT Ukraine / *I.D. Gorbenko, O.G. Kachko, M.V. Yesina* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 5 – 16.

The NTRU-like cryptosystem NTRU Prime IIT Ukraine, created on the basis of existing cryptographic transformations end-to-end encryption type, is considered. The description of this cryptosystem is given and its analysis is carried out. Also, features of its implementation, comparison of the main characteristics and indicators, as well as the definition of differences from existing NTRU-like cryptographic algorithms are presented. Conclusions are made and recommendations are given.

Key words: ring, end-to-end encryption, field, polinomial, quotient ring.

8 tab. Ref.: 9 items.

УДК 004.056.55

Математическая структура потокового шифра Струмок / *А.А. Кузнецов, И.Д. Горбенко, Ю.И. Горбенко, А.Н. Алексейчук, В.А. Тимченко* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 17 – 27.

Излагаются основные результаты по разработке нового генератора ключевого потока Струмок, который предлагается в качестве кандидата на национальный стандарт симметричного шифрования в Украине. Струмок построен по SNOW2.0-подобной схеме суммирующего генератора, увеличены длины секретного ключа и вектора инициализации, что позволяет надежно применять потоковый шифр даже с учетом квантовых методов криптографического анализа. Приводятся основные преобразования и отдельные результаты по исследованию быстродействия шифра, в частности показано, что генератор способен формировать ключевой поток со скоростью более 10 Гбит/с.

Ключевые слова: потоковый шифр, генератор ключевого потока, симметричный стандарт шифрования, псевдослучайные последовательности

Табл. 2. Ил. 3. Библиогр.: 20 назв.

УДК 004.056.55

Математична структура потокового шифру Струмок / *О.О. Кузнецов, І.Д. Горбенко, Ю.І.Горбенко, А.М. Олексійчук, В.А. Тимченко* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 17 – 27.

Викладено основні результати з розробки нового генератору ключового потоку«Струмок», який пропонується як кандидат на національний стандарт симетричного шифрування в Україні. Струмок побудовано за SNOW-2.0-подібною схемою підсумовуючого генератора, збільшені довжини секретного ключа та вектору

ініціалізації дозволяють надійно застосовувати потоковий шифр навіть з із врахуванням квантових методів криптографічного аналізу. Наведено основні перетворення та окремі результати з дослідження швидкодії шифру, зокрема показано, що генератор здатний формувати ключовий потік із швидкістю понад 10 Гбіт/с.

Ключові слова: потоковий шифр, генератор ключового потоку, симетричний стандарт шифрування, псевдовипадкові послідовності

Табл. 2. Іл. 3. Бібліогр.: 20 назв.

UDC 004.056.55

Mathematical structure of the Strumok stream cipher / *O.O. Kuznetsov, I.D. Gorbenko, Y.I. Gorbenko, A.M. Alekseychuk, V.A. Tymchenko* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 17 – 27.

The main development results of a new keystream generator, named "Strumok", and offered as a candidate for the national symmetric encryption standard of Ukraine, are presented. Strumok is built according to the SNOW2.0-like schema of the summation generator; increased secret key length and the initialization vector allow using reliably the stream cipher even taking into account quantum cryptographic analysis methods. The basic transformations and individual results from the cipher performance research are given, moreover, here it is shown the generator, capable of forming a keystream at speed exceeding 10 Gbit per sec.

Key words: stream cipher, keystream generator, symmetric encryption standard, pseudorandom sequences

2 tab. 3 fig. Ref.: 20 items.

УДК 621.391:519.2

Алгоритмы оценивания стойкости SNOW 2.0-подобных потоковых шифров над кольцами вычетов относительно корреляционных атак / *А.Н. Алексейчук, С.М. Игнатенко* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 28 – 34.

Исследуется класс потоковых шифров, которые получаются путем замены в схеме генератора гаммы шифра SNOW 2.0 поразрядного булевого сложения арифметическим сложением по модулю степени числа 2. Разработаны алгоритмы оценивания стойкости таких шифров относительно корреляционных атак, аналогичных известным атакам на SNOW 2.0. Показано, что при определенных условиях указанная замена приводит к существенному повышению стойкости шифра относительно известных корреляционных атак.

Ключевые слова: поточный шифр, корреляционная атака, система линейных уравнений с искаженными правыми частями, кольцо вычетов, алгоритм BKW, SNOW 2.0.

Табл. 1. Ил. 1. Библиогр.: 18 назв.

УДК 621.391:519.2

Алгоритми оцінювання стійкості SNOW 2.0-подібних потокових шифрів над кільцями лишків відносно кореляційних атак / *А.М. Олексійчук, С.М. Ігнатенко* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 28 – 34.

Досліджується клас потокових шифрів, які отримуються шляхом заміни в схемі генератора гами шифру SNOW 2.0 порозрядного булевого додавання арифметичним додаванням за модулем степеня числа 2. Розроблено алгоритми оцінювання стійкості таких шифрів відносно кореляційних атак, аналогічних відомим атакам на SNOW 2.0. Показано, що за певних умов зазначена вище заміна приводить до суттєвого підвищення стійкості шифрів відносно відомих кореляційних атак.

Ключові слова: потоковий шифр, кореляційна атака, система лінійних рівнянь зі спотвореними правими частинами, кільце лишків, алгоритм BKW, SNOW 2.0.

Табл. 1. Іл. 1. Бібліогр.: 18 назв.

UDC 621.391:519.2

Algorithms for evaluation of the SNOW 2.0-like stream ciphers security over residue rings against correlation attacks / *A.N. Alekseychuk, S.M. Ignatenko* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 28 – 34.

The class of stream ciphers obtained by replacing the SNOW 2.0 cipher in the generator scheme with a bitwise addition by an arithmetic addition modulo a power of 2 is investigated. Algorithms for evaluation of such ciphers security against correlation attacks, analogous to the known attacks on SNOW 2.0, are developed. It is shown that under certain conditions the above replacement results in a significant increasing the security of the cipher against known correlation attacks.

Key words: stream cipher, correlation attack, system of linear equations corrupted by noise, BKW algorithm, SNOW 2.0.

1 tab. 1 fig. Ref: 18 items.

УДК 004.056

Анализ атак специального типа для NTRU-подобного алгоритма / *Е.Г. Качко, Ю.И. Горбенко, О.С. Акользина* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 35 – 40.

Приведены результаты анализа NTRU-подобного алгоритма направленного шифрования по стойкости против специальных атак. Сформированы предложения по противодействию этим атакам. Показано насколько

уменьшится пространство ключей при ограничениях пространства ключей для обеспечения свойств случайных последовательностей ключевых данных.

Табл. 5. Ил. 1. Библиогр.: 17 назв.

УДК 004.056

Аналіз атак спеціального типу щодо NTRU-подібного алгоритму / *О.Г. Качко, Ю.І. Горбенко, О.С. Акользіна* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 35 – 40.

Наведено результати аналізу NTRU-подібного алгоритму направленого шифрування щодо стійкості від атак спеціального типу. Сформовано пропозиції щодо протидії цим атакам. Оцінено наскільки зменшується простір ключів, коли відбувається обмеження простору ключів для забезпечення властивостей випадковості ключових послідовностей.

Табл. 5. Іл. 1. Бібліогр.: 17 назв.

UDC 004.056

Side-channel attacks analisys against NTRU-similar algorithm / *O.G. Kachko, Yu. I. Gorbenko, O.S. Akolzina* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 35 – 40.

The results of NTRU-similar algorithm analysis with respect to resistance against side-channel attacks are given. Proposals of defending these attacks are given. The evaluations of how many keys will be reduced with key space limitations of key data (which are needed to ensure the properties of random sequences) are made.

5 tab. 1 fig. Ref: 17 items.

УДК 004.056.55

Первичный анализ и исследование кодовых схем электронной цифровой подписи и направленного шифрования с NIST PQC / *А.С. Киян, М.С. Луценко, А.А. Кузнецов* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 41 – 52.

Исследуются кодовые схемы электронной цифровой подписи и направленного шифрования, представленные на конкурс постквантовых криптоалгоритмов NIST PQC. Приводится общая характеристика алгоритмов, оцениваются основные свойства и параметры. Проводится сравнительный анализ схем электронной цифровой подписи и направленного шифрования по критериям быстродействия и основных криптографических показателей.

Ключевые слова: постквантовая криптография, кодовые подписи, криптографические параметры

Табл. 4. Ил. 11. Библиогр.: 8 назв.

УДК 004.056.55

Первинний аналіз та дослідження кодових схем електронного цифрового підпису та направленого шифрування з NIST PQC / *А.С. Кіян, М.С. Луценко, О.О. Кузнецов* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 41 – 52.

Досліджуються кодові схеми електронного цифрового підпису та направленого шифрування, представлені на конкурс постквантових криптоалгоритмів NIST PQC. Наводиться загальна характеристика алгоритмів, оцінюються основні властивості та параметри. Проводиться порівняльний аналіз схем електронного цифрового підпису та направленого шифрування за критеріями швидкодії і основних криптографічних показників.

Ключові слова: постквантова криптографія, кодові підписи, криптографічні параметри

Табл. 4. Іл. 11. Бібліогр.: 8 назв.

UDC 004.056.55

Primary analysis and research on code-based schemes of electronic digital signature and public-key cryptosystems from NIST PQC / *A.S. Kiian, M.S. Lutsenko, A.A. Kuznetsov* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 41 – 52.

The code-based schemes of electronic digital signature and public-key cryptosystems, submitted to the contest of post-quantum cryptoalgorithms NIST PQC are considered. The general characteristics of the algorithms are give and basic properties and parameters are estimated. The comparative analysis of the electronic digital schemes and public-key cryptosystems is carried out according to the criteria of speed and length of the main cryptographic parameters.

Key words: post-quantum cryptography, code-based signatures, cryptographic parameters

4 tab. 11 fig. Ref.: 8 items.

УДК 004.056.55

Анализ и сравнительные исследования кодовых схем инкапсуляции ключей, представленных на конкурс NIST PQC / *М.С. Луценко, А.С. Киян, Т.Ю. Кузнецова, А.А. Кузнецов* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 53 – 66.

Исследуются кодовые схемы инкапсуляции ключей, представленные на конкурсе NIST PQC. Приводятся результаты первичного сравнительного анализа криптографической стойкости и показателей быстродействия схем инкапсуляции ключей.

Ключевые слова: криптосистемы с открытым ключом на основе кодов, ключевые механизмы инкапсуляции, постквантовая стандартизация

Табл. 2. Ил. 7. Библиогр.: 15 назв.

УДК 004.056.55

Аналіз та порівняльні дослідження кодових схем інкапсуляції ключів, що представлені на конкурсі NIST PQC / *М.С. Луценко, А.С. Кіян, Т.Ю. Кузнецова, О.О. Кузнецов* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 53 – 66.

Досліджуються кодові схеми інкапсуляції ключів, які були представлені на конкурсі NIST PQC. Наводяться результати первинного порівняльного аналізу криптографічної стійкості і показників швидкодії схем інкапсуляції ключів.

Ключові слова: криптосистеми з відкритим ключем на основі кодів, ключові механізми інкапсуляції, постквантовая стандартизація

Табл. 2. Іл. 7. Бібліогр.: 15 назв.

UDC 004.056.55

Analysis and comparative studies of code-based key encapsulation mechanisms submitted to the NIST PQC competition / *M.S. Lutsenko, A.S. Kiian, T.Y. Kuznetsova, A.A. Kuznetsov* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 53 – 66.

The error correcting code-based key encapsulation mechanisms, presented to the NIST PQC competition, are investigated. The results of the primary comparative analysis of the cryptographic properties and the performance indicators of the key encapsulation schemes are presented.

Keywords: code-based public-key cryptosystems, key encapsulation mechanisms, post-quantum standardization

2 tab. 7 fig. Ref.: 15 items.

УДК 004.056.55

Исследование регистров сдвига с нелинейными обратными связями в качестве комбинирующих и фильтрующих функций / *Н.А. Полуяненко, А.В. Потий* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 67 – 74.

Криптографическая стойкость потокового шифра определяется, помимо прочего, способностью сгенерированной псевдослучайной последовательности (ПСП) противостоять аналитическим атакам. Одними из основных составляющих алгоритма формирования ПВП потокового шифра являются комбинирующая и фильтрующая булевы функции. Рассматривается возможность применения регистров сдвига с нелинейными обратными связями второго порядка, формируется последовательность максимального периода в качестве комбинирующих или фильтрующих функций. Исследуются основные показатели криптографической стойкости таких функций: сбалансированность, наличие запретов, корреляционная иммунность и нелинейность. Исследованы и приведены экспериментальные указания корреляционной имунности и нелинейности для всех М-РЗНЗЗ второго порядка размерностью до девяти ячеек включительно.

Ключевые слова: регистры сдвига с нелинейными обратными связями, РЗНЗЗ, NLFSR, фильтрующая функция, комбинирующая функция, потоковые шифры, генераторы псевдослучайных последовательностей.

Табл. 3. Ил. 1. Библиогр.: 11 назв.

УДК 004.056.55

Дослідження регістрів зсуву з нелінійними зворотними зв'язками в якості комбінуючих та фільтруючих функцій / *М.О. Полуяненко, О.В. Потій* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 67 – 74.

Криптографічна стійкість потокового шифру визначається, крім іншого, спроможністю згенерованої псевдовипадкової послідовності (ПВП) протистояти аналітичним атакам. Одними з основних складових алгоритму формування ПВП потокового шифру є комбінуючі та фільтруючі булеві функції. Розглядається можливість застосування регістрів зсуву з нелінійними зворотними зв'язками другого порядку, що формують послідовність максимального періоду в якості комбінуючих або фільтруючих функцій. Досліджується основні показники криптографічної стійкості таких функцій: збалансованість, наявність заборон, кореляційна імунність та нелінійність. Досліджено та приведено експериментальні зазначення кореляційної імунності та нелінійності для всіх М-РЗНЗЗ другого порядку розмірністю до дев'яти чарунок включно.

Ключові слова: регістрів зсуву з нелінійними зворотними зв'язками, РЗНЗЗ, NLFSR, фільтруюча функція, комбінуюча функція, потокові шифри, генератори псевдовипадкових послідовностей.

Табл. 3. Іл. 1. Бібліогр .: 11 назв.

UDC 004.056.55

Investigation of shift registers with nonlinear feedbacks as combining and filtering functions / *N. Poluyanenko, O. Potii* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 67 – 74.

The cryptographic resistance of the stream cipher is determined, among other things, by the ability of the generated pseudo-random sequence (PRS) to resist analytic attacks. One of the main components of the PRS streaming cipher generation algorithm is Boolean functions combining and filtering. The possibility of using shift registers with nonlinear feedbacks of the second order is considered, a sequence of the maximum period as a combining or filtering

function is formed. The main indicators of cryptographic stability of such functions are investigated, namely: balance, presence of inhibitions, correlation immunity and nonlinearity. Experimental indications of correlation immunity and nonlinearity for all M-РЗН33 of the second order with a dimension of up to 9 cells inclusive are investigated and presented.

Key words: shift registers with nonlinear feedbacks, РЗН33, NLFSR, filtering function, combining function, stream ciphers, pseudo-random sequence generators.

3 tab. 1 fig. Ref.: 11 items.

УДК 681.3.06

Анализ сложности реализации криптосистемы на группе Судзуки / *Г.З. Халимов, Е.В. Котух, Ю.А. Сергийчук, А.С. Марухненко* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 75 – 81.

Рассмотрены реализации для криптосистем конечных группах на основе логарифмической подписи и накрытия. Рассмотрена логарифмическая подпись на примере группы перестановки с алгоритмами асимметричного шифрования и расшифрования. Дано описание усовершенствованной криптосистемы MST3 на Судзуки 2-группе с порядком группы q^2. Использование Судзуки 2-группы имеет существенное преимущество в реализации за счет большого центра и простой групповой операции. Получены оценки затрат на шифрование, расшифрование и сравнение с RSA алгоритмом.

Ключевые слова: криптосистема MST3, группа Судзуки, логарифмическая подпись

Табл. 4. Библиогр.: 6 назв.

УДК 681.3.06

Аналіз складності реалізації криптосистеми на групі Судзукі / *Г.З. Халімов, Є.В. Котух, Ю.О. Сергійчук, О.С. Марухненко* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 75 – 81.

Розглянуто реалізації для криптосистем кінцевих груп на основі логарифмічного підпису та накриття. Розглянуто логарифмічний підпис на прикладі групи перестановки з алгоритмами асиметричного шифрування та розшифрування. Надано опис удосконаленої криптосистеми MST3 на Судзукі 2-групі з порядком q^2. Використання Судзукі 2-групи має суттєву перевагу в реалізації, за рахунок великого центру та простої групової операції. Отримано оцінки витрат на шифрування, розшифрування та порівняння з RSA алгоритмом.

Ключові слова: криптосистема MST3, група Судзукі, логарифмічний підпис

Табл. 4. Бібліогр.: 6 назв.

UDC 681.3.06

Analysis of the implementation complexity of the cryptosystem on the Suzuki group / *G.Z. Khalimov, Y.V. Kotukh, Yu.A. Sergiychuk, A.S. Marukhnenko* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 75 – 81.

Implementations for cryptosystems of finite groups based on logarithmic signature and covering are considered. A logarithmic signature is exemplified by a permutation group with the asymmetry of encryption and decryption algorithms. Description of the improved cryptosystem MST3 in Suzuki 2-group with the order of the group q^2 is given. The Suzuki 2-group use has a significant advantage in implementation, due to the large center and simple group operation. Cost estimates for encryption, decryption and comparison with the RSA algorithm are obtained.

Keywords: cryptosystem MST3, Suzuki group, logarithmic signature

4 tab. Ref.: 6 items.

УДК 004.056

Принципы построения децентрализованной инфраструктуры открытых ключей / *Е.В. Исирова, А.В. Потий, В.В. Семенец* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 82 – 93.

Приведены основные принципы построения существующей инфраструктуры открытых ключей. Освещены проблемные вопросы, связанные с функционированием такой системы. Предложена новая концепция построения инфраструктуры открытых ключей с использованием технологии blockchain, которая позволяет избежать слабостей и недостатков иерархической архитектуры.

Ключевые слова: инфраструктура открытых ключей, модель доверия вокруг пользователя, технология blockchain, постквантовые подписи.

Табл. 1. Ил. 6. Библиогр.: 4 назв.

УДК 004.056

Принципи побудови децентралізованої інфраструктури відкритих ключів / *К.В. Ісірова, О.В. Потій, В.В. Семенець* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 82 – 93.

Наведено основні принципи побудови існуючої інфраструктури відкритих ключів. Висвітлено проблемні питання, пов'язані з функціонуванням такої системи. Запропоновано новуа концепцію побудови інфраструктури відкритих ключів з використанням технології blockchain, яка дозволяє уникнути слабкостей і недоліків ієрархічної архітектури.

Ключові слова: інфраструктура відкритих ключів, модель довіри навколо користувача, технологія blockchain, постквантові підписи.

Табл. 1. Іл. 6. Бібліогр .: 4 назв.

UDC 004.056

Principles of decentralized public key infrastructure building / *K.V. Isirova, O.V. Potii, V.V. Semenez* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 82 – 93.

The main principles of building the existing public key infrastructure are given. Problems related to the functioning of such system are described. New concept for public key infrastructure building based on blockchain technology is proposed. Such concept avoids the weaknesses and shortcomings of the hierarchical architecture.

Key words: public key infrastructure, trust model around the user, blockchain technology, post-quantum signatures.

1 tab. 6 fig. Ref.: 4 items.

УДК 004.056.55

Исследование кросплатформенных реализаций потоковых симметричных шифров / *А.А. Кузнецов, В.О. Фроленко, Е.С. Ерёмин, Д.В. Иваненко* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 94 – 106.

Исследуются современные потоковые симметричные шифры Enocoro, Decim, Grain, HC, MUGI, Mickey, Rabbit, RC-4, Salsa20, SNOW2.0, Sosemanuk, Strumok, Trivium, а также алгоритм блочного шифрования AES, который может применяться в потоковых режимах шифрования. Излагаются основные результаты тестирования скоростей формирования ключевого потока при кроссплатформенной реализации алгоритмов шифрования на языке программирования Java на операционных системах Windows 10 (x64), Debian (Kali) и Android (x64). Тестирование проводилось на переносном персональном компьютере (Intel Pentium 3550m 2.3ГГц, оперативна память 4Гб (1600МГц)) и на мобильном устройстве, в частности на смартфоне(Samsung galaxy S7).

Ключевые слова: потоковые шифры, криптография, псевдослучайные последовательности, тесты быстродействия, криптографические преобразования

Табл. 16. Библиогр.: 12 назв.

УДК 004.056.55

Дослідження кросплатформних реалізацій потокових симетричних шифрів / *О.О. Кузнецов, В.О. Фроленко, Є.С. Єрьомін, Д.В. Іваненко* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 94 – 106.

Досліджуються сучасні потокові симетричні шифри Enocoro, Decim, Grain, HC, MUGI, Mickey, Rabbit, RC-4, Salsa20, SNOW2.0, Sosemanuk, Strumok, Trivium, а також алгоритм блокового шифрування AES, який може застосовуватися у потокових режимах шифрування. Викладаються основні результати з тестування швидкості формування ключового потоку при кросплатформовій реалізації алгоритмів шифрування мовою програмування Java на операційних системах Windows 10 (x64), Debian (Kali) та Android (x64). Тестування проводилися на переносному персональному комп'ютері (Intel Pentium 3550m 2.3ГГц, оперативна пам'ять 4Гб (1600МГц)) і на мобільних гаджетах, зокрема на смартфоні (Samsung galaxy S7).

Ключові слова: потокові шифри, криптографія, псевдовипадкові послідовності, тести швидкодії, криптографічні перетворення.

Табл. 16. Бібліогр.: 12 назв.

UDC 004.056.55

Investigation of cross-platform realizations of stream symmetric ciphers / *AA Kuznetsov, V.O. Frolenko, E.S. Eremin, D.V. Ivanenko* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 94 – 106.

Modern symmetric stream ciphers Enocoro, Decim, Grain, HC, MUGI, Mickey, Rabbit, RC-4, Salsa20, SNOW2.0, Sosemanuk, Strumok, Trivium, and a block cipher AES, which can be used in streaming encryption modes, are being researched. The main results of testing the rate of formation of a key stream with cross-platform implementation of encryption algorithms in the Java programming language on operating systems Windows 10 (x64), Debian (Kali) and Android (x64) are presented. Testing was conducted on the portable personal computer (Intel Pentium 3550m 2.3GHz, RAM 4GB (1600MHz)) and on the mobile device, in particular, on the smartphone (Samsung galaxy S7).

Key words: stream ciphers, cryptography, pseudorandom sequences, speedtest, cryptographic transformations

16 tab. Ref.: 12 items.

УДК 621.37:621.391

Вероятностная модель дактилоскопических образов компьютерной биометрической аутентификации / *В.Н. Шлокин, С.Г. Рассомахин* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 107 – 117.

Проведен анализ методов компьютерной биометрической аутентификации. Исследованы особенности процесса идентификации характерных признаков отпечатков пальцев. Предложена вероятностная модель дактилоскопических образов. Обоснована природа и источники ошибок при обработке реализаций портретов отпечатков в условиях искажающих воздействий. Получены описания и характеристики распределений ошибок.

Ключевые слова: биометрическая дактилоскопическая аутентификация, папиллярные линии, характеристики минуций, вероятностная модель дактилоскопических портретов, виды и числовые характеристики распределений ошибок реализаций отпечатков пальцев.

Табл. 1. Ил. 9. Библиогр. 6 назв.

УДК 621.37:621.391

Імовірнісна модель дактилоскопічних образів комп'ютерної біометричної автентифікації / *В.М. Шлокін, С.Г. Рассомахін* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 107 – 117.

Проведено аналіз методів комп'ютерної біометричної автентифікації. Досліджено особливості процесу ідентифікації характерних ознак відбитків пальців. Запропоновано імовірнісну модель дактилоскопічних образів. Обгрунтовано природу і джерела помилок при обробці реалізацій портретів відбитків в умовах зовнішніх впливів. Отримано опис і характеристики розподілів помилок.

Ключові слова: біометрична дактилоскопічна автентифікація, папілярні лінії, характеристики мінуцій, імовірнісна модель дактилоскопічних портретів, види і числові характеристики розподілів помилок реалізацій відбитків пальців.

Табл. 1. Іл. 9. Бібліогр.: 6 назв.

UDC 621.37:621.391

Probabilistic model of fingerprint images of computer biometric authentication / *V.M. Shlokin, S.G. Rasomakhin* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 107 – 117.

Methods of computer biometric authentication are analyzed. Peculiarities of the identification process of fingerprints characteristic features are investigated. A probabilistic model of fingerprint images is proposed. The nature and sources of errors in the processing of imaging portraits under the conditions of distorting influences are grounded. The descriptions and characteristics of error distributions are obtained.

Key words: biometric fingerprint authentication, papillary lines, characteristics of mines, probabilistic model of fingerprinting portraits, types and numerical characteristics of error distributions of fingerprint implementations.

1 tab. 9 fig. Ref.: 6 items.

МЕТОДЫ ПОСТРОЕНИЯ ЗАЩИЩЕННЫХ ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

МЕТОДИ БУДУВАННЯ ЗАХИЩЕНИХ ТЕЛЕКОМУНІКАЦІЙ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

METHODS FOR CONSTRUCTION OF PROTECTED TELECOMMUNICATIONS AND INFORMATION TECHNOLOGIES

УДК 681.3.07 (3.06)

Метод измерения квантового фазового шума и ширины линии рабочего перехода радиооптической системы генератора случайных чисел / *А.П. Нарежний, В.В. Семенец, Т.А. Гриненко* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 118 – 132.

Приведены результаты комплексных теоретических и экспериментальных исследований по созданию прототипа квантового генератора случайных чисел (КГВЧ) на основе реализации метода двойного радиооптического резонанса в парах изотопа рубидия. Разработан метод аналитического и численного решения уравнения связанных мод КГВЧ, который описывает режим взаимодействия квантовых генераторов в процессе измерения их параметров. Особенностью предложенного метода является возможность исследовать квантовые фазовые шумы и ширину линии квантовых дискриминаторов на парах щелочных металлов при наличии погрешности от взаимодействия.

Ключевые слова: квантовый генератор случайных чисел, метод двойного радиооптического резонанса, квантовый фазовый шум, квантовая мера частоты.

Табл. 1. Ил. 3. Библиогр.: 32 назв.

УДК 681.3.07 (3.06)

Метод вимірювання квантового фазового шуму та ширини лінії робочого переходу радіооптичної системи генератора випадкових чисел / *О.П. Нарєжній, В.В. Семенець, Т.О. Гріненко* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 118 – 132.

Наведено результати комплексних теоретичних та експериментальних досліджень зі створення прототипу квантового генератора випадкових чисел (КГВЧ) на основі реалізації методу подвійного радіооптичного резонансу в парах ізотопа рубідію. Розроблено метод аналітичного та чисельного вирішення рівняння зв’язаних мод КГВЧ, який описує режим взаємодії квантових генераторів у процесі вимірювання їх параметрів. Особливістю запропонованого методу є можливість дослідження квантових фазових шумів і ширини лінії квантових дискримінаторів на парах лужних металів при наявності похибки від взаємодії.

Ключові слова: квантовий генератор випадкових чисел, метод подвійного радіооптичного резонансу, квантовий фазовий шум, квантова міра частоти

Табл. 1. Іл. 3. Бібліогр.: 32 назви.

UDC 681.3.07 (3.06)

Method for measuring quantum phase noise and working transition line width of radio-optical system of random number generator / *O.P. Nariezhnii, V.V. Semenets, T.O. Grinenko* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 118 – 132.

Results of comprehensive theoretical and experimental studies on creation of a Quantum Random Number Generator (QRNG) prototype based on the implementation of double radio optical resonance in the vapor of the rubidium isotope are presented. A method is developed for the analytical and numerical solution of the coupled-mode QRNG equation, which describes the mode of interaction of quantum generators in the process of measuring their parameters. A feature of the proposed method is the possibility of studying quantum phase noise and the width of the quantum discriminators line on alkali metal vapors in the presence of an interaction error.

Key words: quantum random number generator, double radio optical resonance method, quantum phase noise, quantum measure of frequency

1 tab. 3 fig. Ref.: 32 items.

УДК 004.652

Инвариантная к предметным областям схема базы данных и ее отличительные особенности / *В.И. Есин* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 133 – 142.

Стремление избежать излишних затрат, свойственных традиционной методологии создания реляционных баз данных, не только на этапах концептуального и логического проектирования, но и на этапе физического проектирования актуализировало задачу разработки инвариантной к предметным областям схемы базы данных. Основными разработанными и отличающимися элементами данной схемы являются: состав, структура базовых отношений; реализации ограничений целостности; средства, обеспечивающие безопасность БД.

Ключевые слова: база данных, реляционная база данных, схема базы данных, модель данных с универсальным базисом отношений.

Ил. 6. Библиогр.: 14 назв.

УДК 004.652

Інваріантна до предметних областей схема бази даних і її відмінні особливості / *В.І. Єсін* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 133 – 142.

Прагнення уникнути зайвих витрат, властивих традиційній методології створення реляційних баз даних, не тільки на етапах концептуального і логічного проектування, а й на етапі фізичного проектування, актуалізувало задачу розробки інваріантної до предметних областей схеми бази даних. Основними розробленими елементами даної схеми і такими, що відрізняються, є: склад, структура базових відношень; реалізації обмежень цілісності; засоби, що забезпечують безпеку БД.

Ключові слова: база даних, реляційна база даних, схема бази даних, модель даних з універсальним базисом відношень.

Іл. 6. Бібліогр.: 14 назв.

UDC 004.652

Database schema invariant to subject domains and its distinctive features / *V.I. Yesin* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 133 – 142.

The desire to avoid unnecessary costs inherent in the traditional methodology of creating relational databases, not only at the stages of conceptual and logical design, but also at the stage of physical design, has actualized the task of developing a database schema that is invariant to subject domains. The main developed and distinguishing elements of this scheme are as follows: composition, structure of the basic relations; implementation of integrity constraints; means ensuring the database security.

Key words: database, relational database, database schema, data model with an universal basis of relations.

6 fig. Ref.: 14 items.

УДК 681.142

Примеры определения ранга числа, представленного в непозиционной системе счисления остаточных классов / *В.А. Краснобаев, А.А. Замула, А.С. Янко* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 143 – 151.

Рассмотрены возможные методы определения позиционного признака непозиционного кода (ППНК) системы счисления в остаточных классах (СОК) ранга числа. Представлены два метода определения ППНК ранга числа в СОК. Основное внимание уделено методу определения ранга числа, основанному на использовании результата доказательства теоремы о ранге суммы двух чисел. Приведены примеры конкретного определения ранга числа.

Ключевые слова: ранг, вычет, система счисления, остаток, переполнение, признак кода, быстродействие, арифметическая операция.

Табл. 2. Библиогр.: 6 назв.

УДК 681.142

Приклади визначення рангу числа, наданого в непозиційній системі числення залишкових класів / *В.А. Каснобаєв, О.А. Замула, А.С. Янко* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 143 – 151.

Розглянуто можливі методи визначення позиційного ознаки непозиційних коду (ПОНК) системи числення в залишкових класах (СЗК) – рангу числа. Представлено два методи визначення ПОНК рангу числа в СЗК. Основна увага приділяється методу визначення рангу числа, заснованого на використанні результату доведення теореми про ранг суми двох чисел. Наведено приклади конкретного визначення рангу числа.

Ключові слова: ранг, відрахування, система числення, лишок, переповнення, ознака коду, швидкодія, арифметична операція.

Табл. 2. Бібліогр.: 6 назв.

UDC 681.142

Examples of determining the rank of a number represented in the non-position system of residual classes / *V.A. Krasnobayev, A.A. Zamula, A.S. Yanko* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 143 – 151.

Possible methods are considered for determining the positional feature of the non-position code (PFNC) of the number system in residual classes (SRC) of the rank of a number. Two methods for determining the rank of a number in a SRC are presented. The main attention is paid to the method of determining the rank of a number based on using the result of the proof of the theorem on the rank of the sum of two numbers. Examples of a specific definition of the rank of a number are given.

Key words: rank, deduction, number system, remainder, overflow, code tag, speed, arithmetic operation.

2 tab. Ref.: 6 items.

УДК 681.3.06:519.248.681

Технологии формирования OFDM сигналов в современных информационно-коммуникационных системах / *А.А. Замула* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 152 – 158.

Рассмотрены технологии формирования сигналов, используемых в системах связи и телекоммуникаций, а также приводится анализ перспективных технологий, которые могут найти применение в беспроводных системах связи широкополосного доступа. Показано, что широко используемая схема модуляции OFDM обладает рядом недостатков, которые могут привести к снижению показателей эффективности систем. Представлены альтернативные технологии формирования сигналов, в частности технология формирования сигналов, основанная на оконной обработке сигналов (W-OFDM), позволяющие устранить недостатки технологии OFDM.

Ключевые слова: множественный доступ, сотовая связь, частотное разделение, помехоустойчивость, интерференция, оконная обработка, пик-фактор, модуляция.

Ил. 1. Библиогр.: 19 назв.

УДК 681.3.06:519.248.681

Технології формування OFDM сигналів в сучасних інформаційно-комунікаційних системах / *О.А. Замула* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 152 – 158.

Розглянуто технології формування сигналів, що використовуються в системах зв'язку і телекомунікацій, а також наводено аналіз перспективних технологій, які можуть знайти застосування в бездротових системах зв'язку широкосмугового доступу. Показано, що широко використовувана схема модуляції OFDM має низку недоліків, які можуть призвести до зниження показників ефективності систем. Представлено альтернативні технології формування сигналів, зокрема технологія формування сигналів, яка заснована на віконній обробці сигналів (W-OFDM), що усувають недоліки технології OFDM.

Ключові слова: множинний доступ, стільниковий зв'язок, частотне розділення, завадостійкість, інтерференція, віконна обробка, пік-фактор, модуляція.

Іл. 1. Бібліогр.: 19 назв.

UDC 681.3.06:519.248.681

Technologies of forming OFDM signals in modern information and communication systems / *A.A. Zamula* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 152 – 158.

Technologies of forming signals used in communication and telecommunications systems are considered, and the analysis of promising technologies that can be used in wireless broadband access communication systems is given. It is shown that a widely used OFDM modulation scheme has a number of shortcomings that can lead to a decrease in the system efficiency indicators. Alternative signal generation technologies are presented, in particular, signal-based signal processing based on window signal processing (W-OFDM), which eliminate the shortcomings of the OFDM technology.

Key words: multiple access, cellular communication, frequency separation, noise immunity, interference, window processing, peak factor, modulation.

1 fig. Ref.: 19 items.

УДК 004.056.53:621.39

Целесообразное распределение затрат на внедрение мер защиты от технических средств разведки / *В.И. Заболотный, А.В. Ермолович* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 159 – 163.

Сформирована математическая модель и предложено решение задачи обоснования целесообразности затрат на защиту информации на предприятии, что обеспечивает максимальный доход от деятельности предприятия. Разработана модель, которая отображает вероятность создания успешной системы мер защиты при некоторых общих затратах на защиту от технических разведок и техническую защиту информации.

Ключевые слова: средства технических разведок; техническая защита информации; защита от технических разведок; вероятность защиты информации

Ил. 5. Библиогр.: 3 назв.

УДК 004.056.53:621.39

Доцільний розподіл витрат на впровадження заходів захисту від технічних засобів розвідки / *В.І. Заболотний, А.В. Єрмолович* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 159 – 163.

Сформульовано математичну модель і запропоновано рішення задачі обґрунтування доцільності витрат на захист інформації на підприємстві, що забезпечують максимальний прибуток загальної діяльності підприємства. Розроблено модель, що відображає імовірність впровадження успішної системи заходів захисту при деяких загальних витратах на захист від технічних розвідок та технічний захист інформації.

Ключові слова: засоби технічних розвідок; технічний захист інформації; захист від технічних розвідок; імовірність захисту інформації.

Іл. 5. Бібліогр.: 3 назви.

UDC 004.056.53:621.39

Expedient allocation of costs for implementation of protection measures against technical reconnaissance means / *V.I. Zabolotniy, A.V. Yermolovych* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 159 – 163.

A mathematical model is developed and a solution of the problem of justification of the expediency of expenses for information protection at the enterprise is proposed, which provides the maximum income from the enterprise's activity. A model is developed that reflects the probability of creating a successful system of protection measures, with some overall costs of protection against technical reconnaissance and of information technical protection.

Key words: means of technical intelligence; technical protection of information; protection against technical intelligence; probability of information protection

5 fig. Ref.: 3 items.

УДК 004.56:004.353.2

Особенности моделирования параметров видеоимпульса для исследования спектров побочных электромагнитных излучений / *В.И. Заболотный, В.И. Перепадя* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 164 – 168.

На основании метода статистических испытаний Монте-Карло описаны особенности моделирования параметров видеоимпульса для исследования спектров побочных электромагнитных излучений. Установлено, что ключевыми параметрами, с точки зрения возможности утечки информации, являются: длина плавного перехода сигнала между линейными частями и длина аппроксимации линейной составляющей фронта импульса, которые и определяют форму сигнала. Используя вероятностный подход, авторы описали метод определения плотности распределения вероятности напряженности электрического поля на границе контролируемой зоны для оценки показателей технической защиты информации.

Ключевые слова: побочные електромагнитные излучения; имитационное моделирование; метод Монте-Карло; видеотракт; видеоимпульс; фронт видеоимульса; спектр видеоимпульса.

Ил. 4. Библиогр.: 3 назви.

УДК 004.56:004.353.2

Особливості моделювання параметрів відеоімпульсу для дослідження спектрів побічних електромагнітних випромінювань / *В.І. Заболотний, В.І. Перепадя* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 164 – 168.

На основі методу Монте-Карло описано особливості моделювання параметрів відеоімпульсу для дослідження спектрів побічних електромагнітних випромінювань. Встановлено, що ключовими параметрами, з точки зору можливості витоку інформації, є: довжина плавного переходу сигналу між лінійними частинами і довжина апроксимації лінійної складової фронту імпульсу, які і визначають форму сигналу. Використовуючи імовірнісний підхід, автори описали спосіб визначення густини розподілу імовірності напруженості електричного поля на межі контрольованої зони для оцінки показників технічного захисту інформації.

Ключові слова: побічні електромагнітні випромінювання; імітаційне моделювання; метод Монте-Карло; відеотракт; відео імпульс; фронт відеоімульса, спектр відеоімпульса.

Іл. 4. Бібліогр.: 3 назви.

UDC 004.56:004.353.2

Features of video-pulse parameters simulation for studying spectra of secondary electromagnetic radiation / *V.I. Zabolotny, V.I. Perepadia* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 164 – 168.

Features of video-pulse parameters simulation for studying spectra of secondary electromagnetic radiation are described on the basis of the Monte-Carlo statistical test method. It is established that the key parameters in terms of the possibility of information leakage are: the length of the smooth transition of the signal between the linear parts and the length of the approximation of the linear component of the pulse front, which determine the shape of the signal. Using a probabilistic approach, a method for determining the density distribution of the probability of electric field intensity at the boundary of the controled zone is described to assess the technical protection of information.

Key words: secondary electromagnetic radiation; simulation modeling; the Monte-Carlo method; video tour; video impulse; front of video emulsion, spectrum of video pulse.

4 fig. Ref.: 3 items.

УДК 004.056.55

Сокрытие данных в структуру файловой системы семейства FAT / *К.Ю. Шеханин, А.А. Колгатин, Е.Е. Деменко, А.А. Кузнецов* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 169 – 178.

Исследуются методы технической стеганографии, а именно – основанные на сокрытии информационных сообщений в структуру файловой системы семейства FAT. Рассмотрен базовый метод, основанный на изменении нумерации отдельных кластеров покрывающих файлов, исследованы его преимущества и недостатки. Предложен усовершенствованный метод, который позволяет значительно увеличить объем сокрытых данных. Приводятся результаты сравнительного анализа методов сокрытия данных в структуру файловой системы семейства FAT.

Ключевые слова: сокрытие информации, стеганография, файловая система, покрывающие файлы

Табл. 6. Ил. 4. Библиогр.: 14 назв.

УДК 004.056.55

Приховування даних у структуру файлової системи сімейства FAT / *К.Ю. Шеханін, А.О. Колгатін, Є.Є. Деменко, О.О. Кузнецов* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 169 – 178.

Досліджуються методи технічної стеганографії, зокрема засновані на приховуванні інформаційних повідомлень в структуру файлової системи сімейства FAT. Розглянуто базовий метод, заснований на зміні нумерації окремих кластерів покрівельних файлів, досліджено його переваги та недоліки. Запропоновано удосконалений метод, який дозволяє значно підвищити обсяг прихованих даних. Наведено результати порівняльного аналізу методів приховування даних у структуру файлової системи сімейства FAT.

Ключові слова: приховування інформації, стеганографія, файлова система, покрівельні файли

Табл. 6. Іл. 4. Бібліогр.: 14 назв.

UDC 004.056.55

Data hiding in the FAT family file system structure / *K.Yu. Shekhanin, A.O. Kolhatin, E.E. Demenko, A.A. Kuznetsov* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 169 – 178.

Methods of technical steganography, namely, those ones based on information messages hiding in the structure of the FAT family file system are studied. The basic method based on changing the numbering of cover files clusters is considered, its advantages and disadvantages are researched. An improved method is proposed that makes it possible to increase significantly the hidden data size. The results of the comparative analysis of hiding data in the structure of the FAT family file system are given.

Key words: information hiding, steganography, file system, cover file

6 tab. 4 fig. Ref.: 14 items.

УДК 004.056.55

Особенности реализации EDELIVERY в контексте электронных доверительных услуг. Опыт Евросоюза / *Е.В. Брошеван, А.В. Потий* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 179 – 185.

Рассматриваются составные части Connecting Europe Facility и услуга eDelivery частности. Анализируются организационные и архитектурные особенности реализации регистрируемой доставки документов и ее функции. Описывается процесс развертывания инфраструктуры для услуги и его проектирования. Рассматривается опыт применения услуги в странах ЕС.

Ключевые слова: электронная идентификация, электронные доверительные услуги, регистрируемая доставка документов, точка доступа, SML, SMP.

Ил. 4. Библиогр.: 6 назв.

УДК 004.056.55

Особливості реалізації EDELIVERY в контексті електронних довірчих послуг. Досвід Євросоюзу / *Є.В. Брошеван, О.В. Потій* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 179 – 185.

Розглядаються складові частини Connecting Europe Facility і послуга eDelivery зокрема. Аналізуються організаційні та архітектурні особливості реалізації реєстрованої доставки документів і її функції. Описується процес розгортання інфраструктури для послуги та його проектування. Розглядається досвід застосування послуги в країнах ЄС.

Ключові слова: Електронна ідентифікація, електронні довірчі послуги, реєстрована доставка документів, точка доступу, SML, SMP.

Іл. 4. Бібліогр.: 6 назв.

UDC 004.056.55

Features of the EDELIVERY implementation in the context of electronic trust services. The experience of the Euro-Union / *E.V. Brochevan, A.V. Potii* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 179 – 185.

The components of Connecting Europe Facility and the eDelivery service are considered. The organizational and architectural features of the implementation of the registered documents delivery and its functions are analyzed. Describes The process of deploying the infrastructure for the service and its design is described. The experience of using the service in the EU countries is considered.

Key words: Electronic identification, electronic trust services, registered delivery of documents, access point, SML, SMP.

4 fig. Ref.: 6 items.

Реализация механизма контроля целостности программного обеспечения в постквантовый период / *А.В. Потий, А.С. Карпенко* // Радиотехника : Всеукр. межвед. науч.-техн. сб. 2018. Вып. 193. С. 186 – 191.

Статья посвящена описанию реализации инфраструктуры открытого ключа с использованием алгоритма цифровой подписи без учета состояния стойкого к атакам с использованием квантового компьютера. Описывается цикл разработки и реализации схемы с использованием современных фреймворков и национального стандарта ДСТУ 7564:2014.

Ключевые слова: инфраструктура открытых ключей, хеш функция, квантовая криптография, схемы без учета состояния, жизненный цикл, контроль версий, веб-сервис.

Табл. 2. Ил. 2. Библиогр.: 9 назв.

УДК 004.056

Реалізація механізму контроля цілісності програмного забезпечення у постквантовий період / *О.В. Потій, А.С. Карпенко* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. 2018. Вип. 193. С. 186 – 191.

Стаття описує реалізацію інфраструктури відкритого ключа з використанням алгоритму цифрового підпису без урахування стану стійкого до атак з використанням квантового комп'ютера. Також описується цикл розробки та реалізації схеми з використанням сучасних фреймворків і національного стандарту ДСТУ 7564:2014.

Ключові слова: інфраструктура відкритих ключів, геш функція, квантова криптографія, схема без урахування стану, схема одноразового підпису, життєвий цикл, контроль версій, вебсервіс.

Табл. 2. Іл. 2. Бібліогр.: 9 назв.

UDC 004.056

Realization of the mechanism of control software integrity in post quantum period / *O. Potii, A. Karpenko* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. 2018. №193. P. 186 – 191.

This article is devoted to description the realization of the public-key infrastructure using the stateless algorithm of digital signature with quantum-resistance. This article is detail describe the development cycle and implementation of the stateless signature scheme using the hash function of DSTU 7564:2014.

Key words: public key infrastructure, hash functions, quantum cryptography, stateless signature, lifecycle, source control, web service.

Tab. 2. Fig. 2. Ref.: 9 items.

УДК 621.391

Исследование структуры спектров сигналов с линейной частотной модуляцией / *И.Д. Горбенко, А.А. Замула* // Радиотехника : Всеукр. межвед. науч.-техн. сб. – 2018. – Вып. 193. – С. 192 – 198.

Получены аналитические выражения для спектра и функции неопределенности составных неравномерных по длительности ЛЧМ сигналов с внутриимпульсной ФМ (СНЛЧМ-ФМ сигналов). Излагаются методика и результаты исследований спектральных, корреляционных и структурных свойств СНЛЧМ-ФМ сигналов при манипуляции фазы различными классами двоичных дискретных последовательностей.

Ключевые слова: спектр, модуляция, функция неопределенности, функция корреляции, составной элемент, информационная безопасность, скрытность.

Ил. 1. Библиогр.: 10 назв.

УДК 621.391

Дослідження структури спектрів сигналів з лінійною частотною модуляцією / *І.Д. Горбенко, О.А. Замула* // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. – 2018. – Вип. 193. – С. 192 – 198.

Отримано аналітичні вирази для спектра і функції невизначеності складних нерівномірних по тривалості ЛЧМ сигналів з внутрішньоімпульсною ФМ (СНЛЧМ-ФМ сигналів). Викладаються методика і результати досліджень спектральних, кореляційних і структурних властивостей СНЛЧМ-ФМ сигналів при маніпуляції фази різними класами двійкових дискретних послідовностей.

Іл. 1. Бібліогр.: 10 назв.

Ключові слова: спектр, модуляція, функція невизначеності, функція кореляції, складний елемент, інформаційна безпека, скритність.

UDC 621.391

Investigation into the structure of spectra of signals with linear frequency modulation / *I.D. Gorbenko, A.A. Zamula* // Radiotekhnika : All-Ukr. Sci. Interdep. Mag. – 2018. – №193. – P. 00 – 00.

Analytical expressions for the spectrum and the uncertainty function for composite non-uniform signals with FM pulse width (SNLFM-FM signals) are obtained. The technique and results of studies of the SNLFM - FM signals spectral, correlation and structural properties are presented in the manipulation of the phase by various classes of binary discrete sequences.

Key words: spectrum, modulation, uncertainty function, correlation function, composite element, information security, stealth.

1 fig. Ref.: 10 items.

ЗБІРНИК НАУКОВИХ ПРАЦЬ
РАДІОТЕХНІКА
Випуск 193
Російською, українською та англійською мовами

СБОРНИК НАУЧНЫХ ТРУДОВ
РАДИОТЕХНИКА
Выпуск 193
На русском, украинском и английском языках

Коректор Л.І. Сащенко

Підп. до друку 15.05.2018. Формат 60x90/8. Папір офсет. Гарнітура Таймс. Друк. ризограф.
Ум. друк. арк**.** 10,6. Обл.-вид. арк. 11,48. Тираж 300 прим. Зам. № 252. Ціна договір.

Харківський національний університет радіоелектроніки (ХНУРЕ)
Просп. Науки, 14, Харків, 61166.

Оригінал-макет підготовлено і збірник надруковано у ПФ „Колегіум”, тел. (057) 703-53-74.
Свідоцтво про внесення суб’єкта видавничої діяльності до Державного реєстру видавців.
Сер. ДК №1722 від 23.03.2004.