ENCRYPTION ALGORITHM BASED ON PROPERTIES OF COLUMN FUNCTIONS FOR APPLICATION IN THE INTERNET OF THINGS

Khasanov Kh.P., Akhmedova O.P., Nazarova M.Kh.
LLC “UNICON.UZ” - Scientific engineering and marketing research center

Abstract: Solving the issue of information security for the Internet of Things, along with the conditions for ensuring the security of other things, also requires a strict limitation of time and resources. This requires revising many cryptographic algorithms and proposing new algorithms.

This paper proposes a block encryption algorithm based on the use of column function properties for application in Internet of Things applications.

Keywords: Internet of things, encryption algorithm, column function, encryption algorithm parameters, decryption algorithm, complexity of algorithm, key tolerance.

Introduction. The Internet of Things is all devices that can be connected to the Internet and become part of the Internet of Things (IoT). The Internet of Things stores huge amounts of user data, due to the number and variety of "things" that are part of it. All this data can be stolen by cybercriminals, and devices can be hacked.

IoT security breaches can have catastrophic consequences. This is because the Internet of Things includes both virtual and physical systems. For example, cyberattacks on systems in healthcare that use the Internet of Medical Things, device hacking can lead to the disclosure of confidential patient data and even threaten patient safety. Hacking devices in smart homes could allow cybercriminals to control the very home people live in.

Designing a completely secure system in IoT is a very difficult task. First, because IoT systems are very heterogeneous, they consist of different devices that

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

have different operating systems, hardware, and use different protocols. Secondly, the systems are very large-scale, they can be both within one apartment and spread to cities and even countries. Thirdly, and very importantly, many IoTs have limited resources: memory, computing power and battery capacity, etc.

In this article, we will focus on one of the most important security methods - data encryption algorithms in IoT.

Review of literature on the subject. National encryption algorithms have been adopted in different countries, for example, in the USA since 2001, the AES (Advanced Encryption Standard) symmetric encryption algorithm has been used, where the key length is 128, 192 and 256 bits. In Russia, the GOST 34.12-2015 standard is used for a message block length of 64/128 bits and a key length of 256 bits. Also, GOST R 34.10-2012 is used for digital signature. These methods are used in data transfer protocols in IoT systems, as well as in smart devices themselves with reduced key lengths.

A comparative analysis of existing encryption algorithms showed that [1-5] the use of XOR, substitution, permutation, expansion-dense permutation, shifts and matrix multiplications, as well as one-way functions of modular arithmetic, as the main modifications, are customary in the practice of cryptographers. In the development of encryption algorithms, the use of large prime and content numbers in symmetric encryption algorithms based on the use of one-way discrete leveling functions reduces their performance and does not allow them to be widely used in encryption.

The problem of developing encryption algorithms using one-way promotion functions in small primitive modules has been an unsolved problem until now. The article describes the possibility of solving this problem using the example of an encryption algorithm based on the column function properties [6, 7].

1. *Features of the encryption algorithm based on the properties of the column function.* The cipher developed is a block cipher and is designed for data whose

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

length is a multiple of 8 bits. It uses the multi-parameter one-way column function properties of modulus arithmetic in the column group.

The input and output of the encryption algorithm consists of *b* blocks made up of columns with *m* elements: input blocks M_{sj} and output blocks Sh_{sj}. The input and output of the decoding algorithm also consist of *b* blocks made up of columns with *m* elements: input blocks Sh_{sj}, output blocks M_{sj}, where $s=1,2,\ldots,b$, $j=0,1,\ldots,m-1$. Division into blocks is performed depending on the encryption module and the number of steps. In order to explain the working principle of the encryption algorithm, from now on, we will mainly talk about converting the plain text of one block to the cipher text and recovering the plain text from the cipher text. This case is enough to fully express the electronic codebook protocol of encryption. The procedure for concatenation of cipher blocks is carried out according to the principle [1-5] using the initialization vector IV. As an initialization vector, a *m*-element column consisting of pseudo-random numbers different from *0* is used according to the encryption module.

In the given encryption algorithm, the main attention is paid to one block - column M_j with *m* elements, formed on the basis of the initial text *M* for the electronic code book, and column Sh_j with *m* elements as the result of encryption. The *0-*, *1-*,...*(m-2)* - elements of the *m*-element column M_j are the initial information (message) to be encrypted. The elements of *M* have a length multiple of the length of the module *p*, and the *(m-1)*-element is the encryption key (symmetric key). The first element of is equal to R_1. The *(m-1)* element of the ciphertext Sh_j is used as an message authentication code and allows to identify it on the receiving side of the ciphertext if changes have occurred during the transmission of the ciphertext.

A symmetric key is a string R^t with *e* elements, whose length is less than the prime modulus *p*. The number of key elements is equal to the number of steps (rounds). The encryption module *p* is a transparent parameter and can be taken as a common parameter for all users. The encryption algorithm, in addition to the symmetric key, uses the encryption and decryption key (K_{sh} , K_{dsh}) generated on the

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

basis of R^t and the step key that includes them. The number of stages in the encryption algorithm is an even number e, and the composition of the stage key is different for odd and even stages: the increase in the strength of the stage key in even stages corresponds to the law of arithmetic progression, in which the difference between the key number of the previous even stage and the key number of the current even stage is constant equal to 4.

The Pohlig-Hellman algorithm [2, 3] can be a prototype for this encryption algorithm. It is known that the Pohlig-Hellman algorithm is one-stage, in which the encryption module n (prime or composite number) as the shared secret and the pair of reciprocal secret integers (K, K^{-1}) is used, the length of the input and output blocks will not be less than the length of the module. The tolerance of the Pohlig-Hellman algorithm is determined by the complexity of the discrete logarithm problem. Therefore, considering that the 530-bit basic module cannot provide sufficient tolerance [8], it becomes clear that the Pohlig-Hellman algorithm can be used only in cases where the basic module $p>2^{530}$ or the component module $n>2^{2*530}$.

The cryptographically strong of the developed cryptosystem is not based on the complexity of the discrete logarithm problem, but on the complexity of finding the symmetric key R^t using brute force and the principle of multi-stepping. This is the principle difference of the cipher based on the use of column functions from the Pohlig-Hellman cipher.

2. Encryption algorithm parameters

The encryption algorithm uses the following parameters:

a) (e, p) is a pair of number of steps e and cipher module p; where e is an even number, p is a prime number, where $p\sim 2^c$, $c=8k$, $1\leq k\leq 20$; for p, the condition that $(p-1)/2$ or $(p+1)/2$ is a prime number must be satisfied; the upper limit of c is determined on the basis of the inequality $160\leq ce_{min}$, taking into account the smallest value of the number of encryption steps (rounds) e_{min} and the difference of the column parameter R bit length $c-|R|=2$ with c in a certain implementation of the cipher;

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

b) encryption key $\underline{R}^t = (R_1, \ldots, R_{i-1}, R_i, \ldots, R_{e-1}, R_e)$ is a symmetric key generated as a sequence of random numbers; where i is the number of the stage sequence;

c) $(k_i, \underline{k_i})$ – Euler's phi function $\varphi(p)$ is opposite to each other by the value of the pair of encryption k_i and decryption $\underline{k_i}$ keys for the i-th step, the secret key generated based on the symmetric key of the cipher, where $2^{c-6} < k_i < 2^c$; it is formed from keys (K_{sh}, K_{dsh}), for example, if the number of steps is $e=4$, then $K_{sh}=(k_1, k_2, k_3, k_4)$, $K_{dsh}=(\underline{k_1}, \underline{k_2}, \underline{k_3}, \underline{k_4})$.

The encryption key k_i and decryption key $\underline{k_i}$ pair are used in two functions in the encryption algorithm: the first function, which is the column function and the degree function in the parameter function; the second task is to use the m-element column elements formed with the participation of the column function and the parameter function at the end of each encryption step and at the beginning of each decryption step as the value of the push step. Here, during encryption, the column elements are moved periodically down the column with a step equal to $k_i \equiv k_i \bmod m$, during decoding, the column elements are periodically moved up the column with a step equal to $\underline{k_i}$. The push action of column elements is used in one of three ways. In the first option, the pushing operation is performed on all m elements of the column, in the second option, on the remaining elements of the column except for the last element. The third option does not use the push action. This article considers the second and third options.

d) round key $(\{\underline{kr}(i)\}, \{\underline{kr}((i\geq 1)_1\}, \{\underline{r}(i\geq 4)_2\})$;

if the round order number is an odd number, then $\underline{kr}(i)=(k_i / \underline{k_i}, R_i)$, k_i is used for encryption, and $\underline{k_i}$ is used for decryption;

if the round sequence number is an even number, then the round key consists of two parts $\underline{kr}\ (i\geq 1)_1$, $\underline{r}(i\geq 4)_2$

$\underline{kr}(i\geq 1)_1 = (k_i / \underline{k_i}, R_i, \sum_1^i R_i \bmod p)$;

if $i\geq 4$, then $\underline{r}(i\geq 4)_2$ elements $r_{ij} \in \underline{r}(i\geq 4)_2$,

$j \in \{1, \ldots, \#\underline{r}\}$ (бу ерда $\#\underline{r} = \#\underline{r}(i\geq 4)_2$).

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

For example, *for* $i=4$ $\underline{r}(i)2 = (r_{i1}, r_{i2}, r_{i3}, r_{i4}, r_{i5})$, where $r_{i1} = R_1 + R_3 \bmod p$, $r_{i2} = R_2 + R_3 \bmod p$, $r_{i3} = R_1 + R_4 \bmod p$, $r_{i4} = R_2 + R_4 \bmod p$, $r_{i5} = R_3 + R_4 \bmod p$

3. *The sequence of steps of the encryption algorithm based on the properties of the column function*

Partitioning into blocks depends on the encryption module and the number of steps in the *e*-round as follows.

Table 1 shows the power (number of elements) $\#\underline{r}(i)$ values of the step-dependent part $\underline{r}(i)_2$ of the round key used in even rounds.

Table 1. of $\underline{r}(i)_2$ is the number of elements

i	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32
$\#\underline{r}(i)$	0	5	9	13	17	21	25	29	33	37	41	45	49	53	57	61

The principle of dividing the initial data into blocks of $(m-1)$ elements is the condition that the inequality $m \leq \#\underline{r}(e)+2$ is satisfied. Here $m \leq \#\underline{r}(e)+2$ it is noted that the complexity of restoring the original text with the given symmetric key is at the highest level when the parity is strong.

If the number of elements of the last block is less than $m-1$, then the missing elements are given a value of 1 when using the electronic code book rule, otherwise, a value of 0 is given.

Then $M_{m-1}=R_1$ is added to each block (column of elements) as $(m-1)$-element, and a one-element extended access block M_{rj} is formed, where $j \in \{0,1,\ldots,m-1\}$. This element acts as a message authentication code.

The processes of converting plaintext to ciphertext and ciphertext to plaintext (encryption) consisting of one block are carried out in the following order.

The encryption process consists of periodically repeating the specified sequence of operations for odd and even rounds.

Below is the sequence of encryption algorithm steps.

M_j , $j \in \{0,1,\ldots,m-1\}$, is entered into the cipher cryptomodule of side A.

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

On the A side, the following sequence of actions is performed:

For i =1 step 2 to e

1. The column parameter for the input column

$R=1+S^{i-1}M_0+ S^{i-1}M_1+\dots S^{i-1}M_{m-2}+ S^{i-1}M_{m-2} +S^{2i-2}M_{m-1} mod\ p$ is calculated;

2. With column parameter $s =1^{\backslash ki}\ mod\ p$ is calculated;

3. *For* $j \in \{0,\dots m-2\}$, $S^iM_j =s *(S^{i-1} M_j)\ mod\ p$ is calculated*;*

4. For $j=m-1$, $S^{2i-1}M_{m-1} = s *(S^{2i-2}M_{m-1})\ mod\ p$ is calculated*;*

5. For $j=m-1$, $S^{2i}M_{m-1} = (S^{2i-1}M_{m-1})^{\backslash\ ki}\ mod\ p$ is calculated with the parameter R_i and the degree indicator k_i;

6. The column elements formed during the step are periodically pushed down along the column with a step equal to k_i and $i=i+1$ is taken;

7. *For* $j= 0$, the parameter $\rho_i=\sum_1^i R_i\ mod\ p$ and with degree index k_i is $S^iM_0=(S^{i-1}M_0)^{\backslash ki}\ mod\ p$; if $i\neq 2$, then it goes to 10;

8. If $i=2$, then $\underline{r}(i)_2=0$ and the parameter for $j \in \{1,\dots,m-2\}$ is $S^iM_j= (S^{i-1}M_j)^{\backslash ki}\ mod\ p$ with S^iM_{j-1}

9. *For* $j=m-1$, parameter $S^{i-1}M_{m-2}$ and $S^{2i-1}M_{m-1}=(S^{2i-2}M_{m-1})^{\backslash ki}\ mod\ p$ is calculated*;*

10. *For* $j= m-1$, the parameter R_i and the degree index k_i are calculated as $S^{2i}M_{m-1} = (S^{2i-1}M_{m-1})^{\backslash ki}\ mod\ p$ and go to step 15;

11. *If* $i\geq 4$, then for $j \in \{1,\dots,\#\ \underline{r}\}$ j (column element) encryption is $S^iM_j= (S^{i-1}M_j)^{\backslash ki}\ mod\ p$ with parameter r_{ij};

12. *For* $j \in \{\#\underline{r}+1,\dots m-2\}$, the parameter $S^{i-1}M_{j-1}$ and $S^iM_j= (S^{i-1}M_j)^{\backslash ki}\ mod\ p$ are calculated;

13. *For* $j= m-1$ parameter $S^{i-1}M_{m-2}$ and $S^{2i-1}M_{m-1}= (S^{2i-2}M_{m-1})^{\backslash\ ki}\ mod\ p$ are calculated;

14. *For* $j= m-1$ $S^{2i}M_{m-1} = (S^{2i-1}M_{m-1})^{\backslash\ ki}\ mod\ p$ is calculated with parameter R_i and order k_i

15. the column elements formed during the round are moved periodically down the column with a step equal to k_i

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

16. *If i<e, then i=i+1* is accepted (goes to the odd stage), otherwise *(i=e)* the calculations are stopped and the formed column is given to the output

end for

Note: Assume that the block corresponding to the ciphertext received on the B side serves as the starting column in the cryptomodule S^iM_j, $j\in\{0,\dots\ m\text{-}2\}$, $S^{2i}M_{m-1}$ in the form.

On the B side, the following sequence of actions is performed:

for i =e step -2 to 0

1. The elements of the initial column included in the decryption cryptomodule are moved periodically down the column with a step equal to k_i;

2. *For j=m-1*, $S^{2i}M_{m-1} = (S^{2i-1}M_{m-1})^{\backslash\ ki}\ mod\ p$ is calculated with the parameter R_i and the degree indicator k_i;

3. *For j= 0*, the parameter $\rho_i=\sum_1^i R_i\ mod\ p$ and with degree index k_i is $S^iM_0=(S^{i-1}M_0)^{\backslash ki}\ mod\ p$, if *i=2*, then it goes to 6;

4. *If i≥4, then for* $j\in\{1,\dots,\#\underline{r(i)}\}$ *the j*-element encryption with r_{ij} parameter is $S^{i-1}M_j= (S^iM_j)^{\backslash ki}\ mod\ p$ is calculated*;*

5. *For* $j\in\{\#\underline{r(i)}+1,\dots\ m\text{-}2\}$ parameter S^iM_{j-1} and $S^{i-1}M_j = (S^iM_j)^{\backslash ki}\ mod\ p$ is calculated*;*

6. For *j=m-1*, the parameter S^iM_{m-2} and $S^{2i-2}M_{m-1} = (S^{2i-1}M_{m-1})^{\backslash\underline{ki}}\ mod\ p$ is calculated and go to step 9;

7. If *i=2*, then $\underline{r(i)}_2 =0$ and the parameter S^iM_{j-1} for $j\in\{1,\dots\ m\text{-}2\}$ and $SM_j = (S^iM_j)^{\backslash ki}\ mod\ p$ is calculated;

8. For *j=m-1*, the parameter S^iM_{m-2} and $S^{2i-2}M_{m-1} = (S^{2i-1}M_{m-1})^{\backslash\underline{ki}}\ mod\ p$ is calculated;

9. The elements of the column formed during the round are periodically pushed up along the column with a step equal to $\underline{k_i}$ and set *i=i-1* (pass to the odd step);

10. *For j= m-1*, $S^{2i}M_{m-1} = (S^{2i-1}M_{m-1})^{\backslash ki}\ mod\ p$ is calculated with the parameter R_i and the degree index $\underline{k_i}$;

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

11. The column parameter $R=1+S^iM_0+S^iM_1+\dots S^iM_{m-2}+S^iM_{m-2}+S^{2i}M_{m-1} mod\ p$ is calculated;

12. $\underline{s}=1^{\backslash ki} mod\ p$ is calculated with the column parameter;

13. . *For* $j\in\{0,\dots m-2\}$, $S^{i-1}M_j=\underline{s}*(S^iM_j)\ mod\ p$ is calculated;

14. *For* $j=m-1$, $S^{2i-2}M_{m-1}=s*(S^{2i-1}M_{m-1})^{\backslash ki}\ mod\ p$ is calculated and set $i=i-1$ (pass to the even step).

end for

Note: $S^{i-1}M_j$, $j\in\{0,\dots m-2\}$, $S^{2i-2}M_{m-1}$ serve as the input data (column) for decoding for the next step, and this process is repeated periodically and ends in 1 round. At the output of the encryption cryptomodule, the recovered text is in the form $M'_j, j\in\{0,\dots m-1\}$ and if the condition $M'_{m-1}=R_1$ is satisfied, then the original text M'_j, $i\in\{0,\dots m-2\}$ is valid (the ciphertext is not modified), otherwise it is concluded that the ciphertext is modified.

The encryption algorithm presented above is a sequential scheme of pairs of rounds *(1-2,3-4,5-6,….(e-1)-e).* Such a scheme is characterized by the fact that as the number of rounds increases, the tolerance of the algorithm is 2^{4r} times higher than the complexity of the previous pair of round cryptanalysis (where r is the average value of the length of the rank parameter). In the implementation of the encryption algorithm, it is worth noting the use of other schemes of pairs of rounds, for example, the use of a spiral scheme in which the average complexity of the stage cryptanalysis is almost equally distributed. In a spiral scheme, when $0=e\ mod\ 4$ and $0=e\ mod\ 2$, the scheme of pairs of rounds, correspondingly, looks like this:

(1-2, (e-1)-e, 3-4, (e-3)-(e-2), 5-6, (e-5)- (e-4),… , (e/2-1)- e/2),

(1-2, (e-1)-e, 3-4, (e-3)-(e-2), 5-6, (e-5)- (e-4),… , e/2).

In the spiral scheme, the mixing index of the column elements is high in two rounds of the algorithm.

In the figure below example is given for the case where, the module $p=227$, $e=4$ and the encryption key $\underline{R}^t=(201,153,191,72)$, $K_{sh}=(141,53,117,153)$,

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

K_{dsh}=(109,145,85,65), 127= R1+ R2 mod p, 163=R1+R2+R3+R4 mod p, $\underline{r}(4)_2$=(165,117,46,225,36).

Here, in order to make it easier to observe the mutual compatibility of the elements of the round, the encryption algorithm does not use the operation of "elements pushing".

Table 2. The encryption and decryption process of the encryption algorithm

Encryption									
1-round						2-round			
M_j		R	$s=1^{\backslash k1}$	SM=s*M			R_1+R_2	Sh^2M	
20		24	220	87			127	157	
151				78				217	
37				195				2	
215				84				120	
200				189				109	
107				159	s^2			155	sh^4
201				182	162			11	101
3- round						4- round			
Sh^2M		R	$s=1^{\backslash k3}$	$Sh^3M=s*Sh^2M$			$R_1+R_2+R_3+R_4$	Sh^4M	
157		181	75	198			163	53	
217				158			165	108	
2				150			117	9	
120				147			46	13	
109				3			225	148	
155				48	sh^6		36	196	sh^8
101				84	189			23	141
Decryption									
3- round				4- round					

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

$Sh^2M=\underline{s}*SM$		R	$s=\underline{1^{\backslash k3}}$	$R_1+R_2+R_3+R_4$		Sh^3M		Sh^4M	
157		108	112	163		198		53	
217				165		158		108	
2				117		150		9	
120				46		147		13	
109				225		3		148	
155				36		48	s^5	196	s^7
101	11					85	84	141	23
1- round				2- round					
M_j =s*SM		R	$s=\underline{1^{\backslash k1}}$	R_1+R_2		Sh M		Sh^2M	
20		67	162	127		87		157	
151						78		217	
37						195		2	
215						84		120	
200						189		109	
107						159		155	s^3
201					145	162	182	11	101

The analysis of the encryption algorithm given as an example showed that changing one element of M_j by 1 bit at the input causes all elements to change in the second round. If one of the two elements of M_j changes by +1 bit and the other by -1 bit, then this change causes all the elements of the current column to change in the 3rd round. Consequently, the encryption algorithm based on column function properties meets the requirements specific to modern symmetric encryption algorithms.

The level of complexity of this algorithm = $2^{e9,5+4\Sigma\#\underline{r}\,(i)}$ is determined by the complexity of performing actions. If the encryption algorithm uses periodic shifting of column elements, it is natural that the level of complexity of the algorithm will be higher.

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

For $e_{min}=20$, the level of complexity $=2^{e9,5+4\Sigma\# \underline{r}\,(i)} =2^{954}$ is determined by the complexity of performing the operation according to the basic module $p<256$.

Table 3. Complexities are bit lengths

e	$\underline{R}^t$	$e9,5$	$\underline{r}$	My	$\underline{r}$	My	$\underline{r}$	My	$\underline{r}$	My	$\underline{r}$	My
32	224	304									503	2316
30	210	285							441	2049	441	2049
28	196	266					383	1798	383	1798	383	1798
26	182	247					329	1563	329	1563	329	1563
24	168	228			279	1344	279	1344	279	1344	279	1344
22	154	209			233	1141	233	1141	233	1141	233	1141
20	140	190	191	954	191	954	191	954	191	954	191	954
18	126	171	153	783	153	783	153	783	153	783	153	783
16	112	152	119	628	119	628	119	628	119	628	119	628
14	98	133	90	493	90	493	90	493	90	493	90	493
12	84	114	65	374	65	374	65	374	65	374	65	374
10	70	95	44	271	44	271	44	271	44	271	44	271
8	56	76	27	184	27	184	27	184	27	184	27	184
6	42	57	14	113	14	113	14	113	14	113	14	113
4	28	38	5	58	5	58	5	58	5	58	5	58
2	14	19	0	19	0	19	0	19	0	19	0	19

Summary.

Encryption key tolerance $\underline{R}^t$ consists of 7-bit elements with 2^{140} operations when $e_{=}e_{min}$, 2^{168}, 2^{196}, 2^{210}, 2^{224}, 2^{238}, 2^{252} operations when the number of rounds is $e=24, 28, 30, 32, 34, 36$ respectively determined by demand. If the encryption algorithm uses periodic shifting of the column elements, then the brute force attack object $(\sum_1^i R_i \bmod p \sum_1^i, R_i, r_{ij}, M_j)$ for each stage is unknown to the cryptanalyst, which complicates the cryptanalyst's work. Such a relationship between the levels of

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/

complexity of recovering the original text and finding the encryption key has not been achieved in algorithms developed in most foreign countries.

References

1. Wenbo Mao. Modern Cryptography: Theory and Practice, 1/e (Hewlett-Packard Professional Books) – Prentice Hall; Reprint edition, 2003. – 707 p.

2. Menezes A., van Oorschot P., Vanstone S. Handbook of Applied Cryptography // CRC Press, 1996. – 780 p.

3. Bruce Schneier Applied Cryptography Protocols, Algorithms, and Source Code in C. - 20th Anniversary Hardcover, 2015 – 784 p.

4. Молдовян Н., Молдовян А. Криптография. От примитивов к синтезу алгоритмов. Изд.:BHV-СПб, 2004.

5. Акбаров Д.Е. «Ахборот хавфсизлигини таъминлашнинг криптографик усуллари ва уларнинг қўлланилиши» - Т.: «Ўзбекистон маркаси», 2009, 424 б.

6. Хасанов Х.П. Диаматрица устунлар ва параметрлар алгебраси // Кибернетика масалалари. – Тошкент, 2006, №173. – 93-100 б.

7. Хасанов Х.П. Такомиллашган диаматрицалар алгебралари ва параметрли алгебра асосида криптотизимлар яратиш усуллари ва алгоритмлари. Тошкент, ФТМТМ, 2008. 208 б.

8. Немецкие ученые успешно решили проблему дискретного логарифмирования по модулю 530-битного (160 десятичных знаков) простого числа р.– http://www.securitylab.ru.

ISSN 2521-3261 (Online)/ ISSN 2521-3253 (Print)
https://journalofresearch.eu/