Usage of non-commutative two-operand CET-operations in limited resources stream ciphers

V. Rudnytskyi [#1], N. Lada [#1], V. Larin [#1], O. Melnyk [#2], T. Stebetska[#3], T. Korotkyi[#4], D. Pidlasyi[#4]

[#1] State Scientific Research Institute of Armament and Military Equipment Testing and Certification, Cherkasy, Ukraine

[#2] State University of Civil Defence of Ukrain, Cherkasy, Ukraine

[#3] The Bohdan Khmelnytsky National University of Cherkasy, Cherkasy, Ukraine

[#4] Cherkasy State Technological University, Cherkasy, Ukraine

***Abstract*— Primary research task: to develop and evaluate different scenarios of utilizing the limited resources stream encryption based on noncommutative CET-operations; to develop and scientifically justify the structure of the devices applicable to the execution of encryption scenarios. The object of research: processes of data transformation in the limited resource stream ciphers based on CET-operations. We have used the hypothesis claiming that it is possible to improve the stream encryption systems if there is an option to permutate operands in both the commutative and non-commutative CET-operations as the foundation of our research. We have also defined and formalized the six groups of two-operand CET-operations allowing the permutation of operands to solve tasks according to the introduced classification. The attributes of the six groups mentioned above influence the data transformation process in stream ciphers. In addition, we have also defined the three possible scenarios of stream encryption, including their limitations, which consider the usage of the aforementioned groups of CET-operations. The provided examples combined with the results of modeling the processes of data transformation allow us to prove the operating efficiency of cryptographic systems in different encryption scenarios. The introduced and analyzed structures of devices based on the aforementioned stream encryption scenarios also reflect the attributes of their technical implementation. By examining the created models and the aforementioned devices for executing the stream encryption scenarios, we have defined the most prominent development paths for the limited resources stream encryption and created the foundation for their future improvement. The results of our analysis prove that all three scenarios have their own positive and negative aspects. For example, we suggest using the first and the third stream encryption scenarios in a single device during the practical implementation. This option suits the scenarios we have researched since it provides the best diversity of encryption algorithms with minimal complexity and a maximal number of substitution tables. We have defined mobile and stationary systems of limited resources cryptographic security of the confidential data as the most suitable branch for the application of the acquired results.**

***Keywords*— limited resources cryptography, post-quantum cryptography, cryptographic coding, stream encryption, CET-encryption, CET-operations, two-operand operations**

I. Introduction

Cybersecurity has become one of the necessary conditions to ensure the appropriate functioning of the digital data space. Thus, improving data security is a very important matter for both the average member of the society and the state as a whole. Much like in the past, today cryptographic security remains the most prominent way of data security [1]-[2]. For this reason, numerous scientific research papers related to cryptography are dedicated primarily to the issue of improving the existing means and methods of cryptographic protection, as well as developing new ones [3]-[6]. The most important research aspects for the post-quantum and limited resources cryptography are the improvement of cryptographic integrity, speed, and efficiency of cryptographic algorithms under conditions of reduced number of both the software and technical assets [7]-[10]. The analysis of the existing relevant literature, however, allows us to conclude that research papers related to post-quantum and limited resources cryptography are mostly focused on the improvements of block encryption, while the matter of limited resources stream encryption is underresearched [11]-[16].

II. Analysis of sources and target setting

Creation of an efficient limited resource cryptographic system is rather complex since it requires a proper balancing of different elements, such as cryptographic integrity and execution complexity, cryptographic integrity and execution speed, execution complexity and available resources, etc. Solving the aforementioned issue is possible, but only conceptually. One solution revolves around the utilization of CET-operations (CET – Cryptographic Encoding Theory) for cryptographic transformation of data. These operations are an integral part of CET-encryption.

The following articles [17], [18] highlight the results of research related to the influence of CET-encryption on the evaluation of the possible ways to improve the existing block ciphers. At the same time, they provide no data regarding the improvement of stream ciphers.

In general, two-operand operations are best suited for the utilization of stream ciphers. Article [19] highlights the creation process and utilization of operations related to the cryptographic modulo two addition to an accuracy of permutation in stream ciphers. Article [20] describes the technology for creating the symmetrical two-operand CET-operations according to the results of modeling. Article [21] describes the creation of the symmetrical two-operand CET-operations with random bitness. All of the aforementioned articles, however, describe only the symmetrical commutative operations, in which the results of cryptographic data transformation are unaffected by the permutation of operands.

Creation of non-symmetrical two-operand CET-operations with the maximal uncertainty of the encryption results is described in articles [22], [23]. Yet the aforementioned operations exclude the permutation of operands.

The monograph [24] describes the development principles related to the technology of stream encryption based on CET-operations. However, we believe that it does not properly highlight the utilization of the non-commutative CET-operations in stream ciphers.

To summarize, we are certain that all of the aforementioned research papers do not appropriately highlight both the symmetrical and commutative attributes of the two-operand CET-operations during creation of the models of stream ciphers.

III. PURPOSE AND OBJECTIVES OF RESEARCH

The purpose of this paper is to develop and evaluate different scenarios of utilizing the limited resources stream encryption based on non-commutative CET-operations, as well as the structures of the devices suitable for their execution. Fulfilling this purpose will allow us to define the most prominent development ways for the limited resources encryption and lay the foundation for its further evolution.

We have established the following objectives to fulfill the aforementioned purpose:

- to study and classify the CET-operations allowing the permutation of operands;
- to define the primary stream encryption scenarios and their existence constraints;
- to model the processes of data transformation corresponding with the aforementioned stream encryption scenarios;
- to develop the structures for the devices suitable for the execution of the proposed stream encryption scenarios;
- to evaluate the execution results of the proposed stream encryption scenarios, as well as their benefits and drawbacks.

IV. MATERIALS AND METHODS

The object of our research is the processes of data transformation in the limited resource stream ciphers based on CET-operations.

The primary hypothesis of this research is the possibility of permutating operands in both the commutative and non-commutative CET-operations. This allows for improvement in the systems of stream encryption.

We use the discrete models of CET-operations created as a result of the simulation experiment to prove this hypothesis. To identify the connection between the models of CET-operations and their mutual transformations, we use the methods of discrete mathematics, theory of sets and linear algebra during our research of encryption scenarios.

We have utilized only the two bit two-operand CET-operations for the first-hand objectives setting and presentation of the acquired results of our research related to the encryption scenarios. We explain this limitation of ours by the presence of a singular mathematical apparatus used to describe an entire set of CET-operations G_4, as well as by the purposes of the demonstration simplicity and the possibility for the mutual transformation of operations models.

V. Research results of the stream encryption process based on the non-commutative two-operand CET-operations

In [24], a two-operand CET-operation $C(x, y)$ is defined as a tuple of single-operand operations used to transform the first operand x depending on the value of the second operand y. A single-operand operation $C(x)$ is described as a discrete model of a substitution table, which is utilized as a foundation for the cryptographic transformation of an operand x.

Among the non-symmetrical two-operand CET-operations, CET-operations, which allow permutation of operands, are very important.

If, however, this permutation within a CET-operation does not change its execution results, then this operation will be considered commutative:

$$C(x, y) = C(y, x). \tag{1}$$

Correspondingly, if permutation of operands does alter the execution results, then a CET-operation will thus be considered non-commutative:

$$C(x, y) \neq C(y, x). \tag{2}$$

Only those CET-operations, that allow permutation of operands, can be either commutative or non-commutative.

If operations of both the direct and inverse cryptographic transformation are in line with each other, then a CET-operation will be considered symmetrical:

$$C(x, y) = C^{/}(x, y), \tag{3}$$

where $C^{/}(x, y)$ is a CET-operation of an inverse cryptographic transformation.

Correspondingly, if operations of both the direct and inverse cryptographic transformation are not in line with each other, then a CET-operation will be considered non-symmetrical: $C(x, y) \neq C^{/}(x, y)$.

We will use models of two-operand CET-operations created as a result of simulation experiments during our further research. Methods utilized to create such models are highlighted in [20], [23].

We will now analyze the models of CET-operations before and after the permutation of operands. Let us suppose that a cryptographic transformation is conducted based on a model of CET-operation:

$$C(x, y) = \begin{cases} \begin{bmatrix} x_1 \oplus 1 \\ x_2 \end{bmatrix}, & if\ \ y_1 = 0;\ y_2 = 0 \\ \begin{bmatrix} x_1 \oplus 1 \\ x_2 \oplus 1 \end{bmatrix}, & if\ \ y_1 = 0;\ y_2 = 1 \\ \begin{bmatrix} x_1 \\ x_2 \oplus 1 \end{bmatrix}, & if\ \ y_1 = 1;\ k_2 = 0 \\ \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, & if\ \ y_1 = 1;\ y_2 = 1 \end{cases} . \tag{4}$$

An important note is that a CET-operation (4) is symmetrical since the following equation $C(x) = C^{/}(x)$ is valid for all single-operand CET-operations within its tuple. As a result, the equation (1) will be valid as well. Thus, an equation $C^{/}(C(x, y), y) = x$ will be valid for the CET-operation (4).

By permutating the operands in a model of the non-symmetrical two-operand CET-operation (4), we acquire the following results:

$$C(y,x)=\begin{cases}\begin{bmatrix}y_1\oplus 1\\ y_2\end{bmatrix}, & if\ \ x_1=0;\ x_2=0\\ \begin{bmatrix}y_1\oplus 1\\ y_2\oplus 1\end{bmatrix}, & if\ \ x_1=0;\ x_2=1\\ \begin{bmatrix}y_1\\ y_2\oplus 1\end{bmatrix}, & if\ \ x_1=1;\ x_2=0\\ \begin{bmatrix}y_1\\ y_2\end{bmatrix}, & if\ \ x_1=1;\ x_2=1\end{cases}=\begin{cases}\begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\end{bmatrix}, & if\ \ y_1=0;\ y_2=0\\ \begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \ y_1=0;\ y_2=1\\ \begin{bmatrix}x_1\\ x_1\oplus x_2\end{bmatrix}, & if\ \ y_1=1;\ y_2=0\\ \begin{bmatrix}x_1\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \ y_1=1;\ y_2=1\end{cases}. \qquad (5)$$

This example shows that transformation of the incoming data x is conducted by different CET-operations as a result of the permutation of operands. However, this permutation of operands also results in a transformation of a symmetrical CET-operation into a non-symmetrical one. This is explained by the fact that the following inequation $C(x)\neq C^{/}(x)$ is valid for single-operand operations within a tuple of a two-operand operation. Thus, an inequation $C^{/}(C(x,y),y)\neq x$ will be valid for the CET-operation (5).

The CET-operation shown as an example is non-commutative since the following inequation $C(x,y)\neq C(y,x)$ is valid according to (2) and (3).

We can make the following assumptions according to this device:

Depending on conditions, a commutative CET-operation can be:

1. $\begin{cases}C(x,y)=C(y,x)\\ C^{/}(C(x,y),y)=x\\ C^{/}(C(y,x),x)=y\end{cases}$ – commutative symmetrical. In this case, CET-operations are symmetrical before and after the permutation of operands.

2. $\begin{cases}C(x,y)=C(y,x)\\ C^{/}(C(x,y),y)\neq x\\ C^{/}(C(y,x),x)\neq y\end{cases}$ – commutative non-symmetrical. In this case, CET-operations are non-symmetrical before and after the permutation of operands.

Depending on conditions, a non-commutative CET-operation can be:

3. $\begin{cases}C(x,y)\neq C(y,x)\\ C^{/}(C(x,y),y)=x\\ C^{/}(C(y,x),x)=y\end{cases}$ – non-commutative symmetrical. In this case, CET-operations are symmetrical before and after the permutation of operands.

4. $\begin{cases}C(x,y)\neq C(y,x)\\ C^{/}(C(x,y),y)\neq x\\ C^{/}(C(y,x),x)\neq y\end{cases}$ – non-commutative non-symmetrical. In this case, CET-operations are non-symmetrical before and after the permutation of operands.

5. $\begin{cases}C(x,y)\neq C(y,x)\\ C^{/}(C(x,y),y)=x\\ C^{/}(C(y,x),x)\neq y\end{cases}$ – non-commutative symmetrical/non-symmetrical. In this case, CET-operation is symmetrical before the permutation of operands but becomes non-symmetrical after the permutation of operands. Previously analyzed CET-operation (4) is an example.

6. $\begin{cases} C(x,y) \neq C(y,x) \\ C^{/}(C(x,y),y) \neq x \\ C^{/}(C(y,x),x) = y \end{cases}$ – non-commutative non-symmetrical/symmetrical. In this case, CET-operation is non-symmetrical before the permutation of operands but becomes symmetrical after the permutation of operands.

All of the aforementioned attributes of two-operand CET-operations, which allow permutation of operands, should be taken into account during the research of stream ciphers created on their basis.

We will now analyze the options of utilizing the two-operand CET-operations, which allow permutation of operands, in stream ciphers.

The cryptographic transformation of the incoming data in stream encryption is conducted by the addition of XOR cipher (γ). A pseudo-random sequence, which remains unchanged during both the encryption and decryption is used as the aforementioned XOR cipher.

When operands are permutated, the same two-operand CET-operation executes two different models of the cryptographic data transformation (such as models (4) and (5)). We can thus assume that two-operand CET-operations, which allow permutation of operands, are capable of ensuring the simultaneous existence of several different scenarios of stream encryption. The operands within a two-operand CET-operation are not permutated in the first encryption scenario. They are, however, permutated in the second encryption scenario. We will now analyze this issue in detail.

5. 1. The first stream encryption scenario

The first encryption scenario exists if there is a CET-operation $C(x,\gamma)$, and a CET-operation $C^{/}(x,\gamma)$, so that $C^{/}(C(x,\gamma),\gamma) = x$.

Let us assume that no permutation of operands has occurred within a two-operand CET-operation.

If a two-operand CET-operation is utilized during stream encryption, then incoming (plaintext) data is encoded into the first operand. The XOR cipher used to control the transformation of the incoming data is encoded into the second operand. According to the analyzed example, the data encryption proceeds based on the following model:

$$C(x,\gamma) = \begin{cases} \begin{bmatrix} x_1 \oplus 1 \\ x_2 \end{bmatrix}, & if\ \gamma_1 = 0; \gamma_2 = 0 \\ \begin{bmatrix} x_1 \oplus 1 \\ x_2 \oplus 1 \end{bmatrix}, & if\ \gamma_1 = 0; \gamma_2 = 1 \\ \begin{bmatrix} x_1 \\ x_2 \oplus 1 \end{bmatrix}, & if\ \gamma_1 = 1; \gamma_2 = 0 \\ \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, & if\ \gamma_1 = 1; \gamma_2 = 1 \end{cases} = z, \tag{6}$$

where z is a result of the encryption of incoming data (x).

CET-operations used in cryptographic coding can be either symmetrical or non-symmetrical [21], [24].

For this reason, data decryption often requires an inverse CET-operation. Since all single-operand operations are symmetrical in CET-operation (4), then a two-operand operation is symmetrical as well [24]:

$$C^{/}(z,\gamma)=C(z,\gamma)=\begin{cases}\begin{bmatrix} z_1\oplus 1 \\ z_2 \end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=0 \\ \begin{bmatrix} z_1\oplus 1 \\ z_2\oplus 1 \end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=1 \\ \begin{bmatrix} z_1 \\ z_2\oplus 1 \end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=0 \\ \begin{bmatrix} z_1 \\ x_2 \end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=1 \end{cases}=x \tag{7}$$

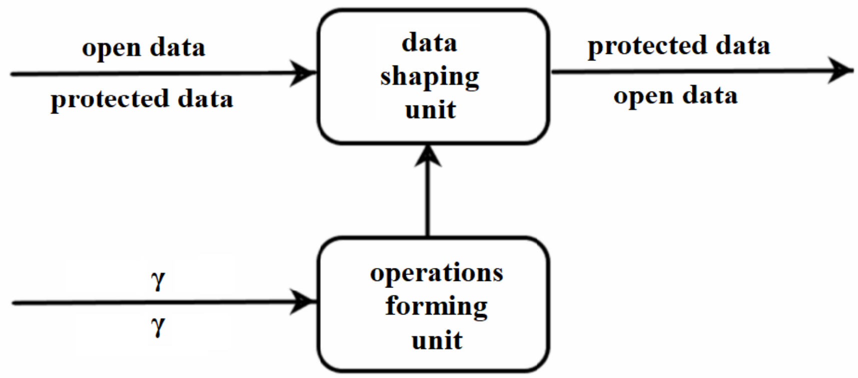
Under condition $x=z$, the following expression $C(x,\gamma)=C(z,\gamma)$ is valid for the models (6) and (7). In this example, both the encryption and decryption of data is conducted by the same CET-operation.

Thus, the model (6) describes the encryption of plaintext information by the symmetrical CET-operation, which allows permutation of operands. The encryption itself is conducted by a XOR cipher. Proper execution of the aforementioned model results in the acquisition of encrypted (protected) data. The model of an inverse CET-operation (7) conducts the decryption of such encrypted (protected) data. The models of both the direct and inverse CET-operations are identical. The decryption process too is conducted by a XOR cipher. Recurred execution of the model of CET-operation results in the acquisition of the decrypted (plaintext) data.

The following expression $C(x,\gamma)\neq C(z,\gamma)$ is valid under condition $x=z$ if a non-symmetrical CET-operation is utilized in this scenario. Therefore, it is necessary to use a model of an inverse CET-operation not identical to the model of CET-operation used for encryption to decrypt the data.

Structure of the device suitable for execution of the first stream encryption/decryption scenario is illustrated in Picture 1.

We will now explain the functionality of this device. At first, the plaintext information is transmitted to the data entry point of the data transformation unit. The XOR cipher is transmitted to the data entry point of the data shaping unit. Depending on the bit value of this XOR cipher within the data shaping unit, the sequential number of a single-operand CET-operation is then defined. The operation itself conducts the cryptographic transformation while its sequential number is transmitted to the data shaping unit. The aforementioned data shaping unit fulfills the functions of the decoder for the XOR cipher (γ). The transmitted plaintext information is then encrypted by the single-operand CET-operation from a tuple of a two-operand CET-operation according to its sequential number, which is acquired from the operations shaping unit. The whole process takes place within a data transformation unit. As a result, encrypted (protected) data is transmitted from the data output point of the data transformation unit.



Picture 1 Structure of the device suitable for the execution of the first stream encryption scenario

We will now explain the functionality of this device during the process of decryption of encrypted data. At first, the protected information is transmitted to the data entry point of the data transformation unit. The XOR cipher is transmitted to the data entry point of the data shaping unit. The bit value of the XOR cipher in the data shaping unit defines the sequential number of the inverse single-operand CET-operation. The acquired sequential number is then transmitted from the data output point to the data transformation unit. Since the XOR cipher remains unchanged during both encryption and decryption, the sequential number of the single-operand CET-operation transmitted to the data transformation unit will be identical to that of an inverse single-operand CET-operation. The transmitted protected information is then decrypted by the inverse single-operand CET-operation from a tuple of an inverse two-operand CET-operation according to its sequential number, which is acquired from the operations shaping unit. The whole process takes place within a data transformation unit. As a result, decrypted (plaintext) data is transmitted from the data output point of the data transformation unit.

Both the symmetrical and non-symmetrical operations can be used in the first encryption scenario.

Symmetrical CET-operations can be used in cryptographic systems with the device from Picture 1. In this case, both the encryption and decryption will require the utilization of the same symmetrical operation. However, usage of a non-symmetrical CET-operation during stream encryption will require the utilization of a direct operation during data encryption, and an inverse operation during data decryption. This condition exists due to $C^{/}(x,\gamma)\neq C(x,\gamma)$ [24].

5. 2. The second stream encryption scenario

The second encryption scenario exists if there is a CET-operation $C(x,\gamma)$, and a CET-operation $C(\gamma,x)$, so that $C(x,\gamma)\neq C(\gamma,x))$.

The operands are permutated in the second scenario. In addition, a single-operand operation of cryptographic transformation is conducted on the XOR cipher (the first operand). This operation is defined by the plaintext information. Thus, an encrypted XOR cipher is transmitted into the data transmission channel instead of the encrypted data. The data is then decrypted by a cryptographic transformation of the XOR cipher by the single-operand operations. These operations are defined by the XOR cipher itself. Depending on how the operands are permutated within the CET-operation used as the foundation for the data decryption, an additional third encryption scenario can be derived from the second scenario.

We will now analyze the encryption of the XOR cipher by using the cryptographic system with CET-operation (4) as an example. Cryptographic system transforms the incoming data under control of the XOR cipher based on the model (6). The whole process occurs as a part of the first encryption scenario. Thus, the first encryption scenario results in the acquisition of the following cryptogram z.

Execution of the second encryption scenario in CET-operation (6) requires the permutation of operands.

We will now modify the encryption model (6) by permutating the operands similarly to (5).

$$C(\gamma,x)=\begin{cases}\begin{bmatrix}\gamma_1\oplus 1\\ \gamma_2\end{bmatrix}, & if\ x_1=0;\ x_2=0\\ \begin{bmatrix}\gamma_1\oplus 1\\ \gamma_2\oplus 1\end{bmatrix}, & if\ x_1=0;\ x_2=1\\ \begin{bmatrix}\gamma_1\\ \gamma_2\oplus 1\end{bmatrix}, & if\ x_1=1;\ x_2=0\\ \begin{bmatrix}\gamma_1\\ \gamma_2\end{bmatrix}, & if\ x_1=1;\ x_2=1\end{cases}=\begin{cases}\begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\end{bmatrix}, & if\ \ \gamma_1=0;\ \gamma_2=0\\ \begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=0;\ \gamma_2=1\\ \begin{bmatrix}x_1\\ x_1\oplus x_2\end{bmatrix}, & if\ \ \gamma_1=1;\ \gamma_2=0\\ \begin{bmatrix}x_1\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=1;\ \gamma_2=1\end{cases}. \tag{8}$$

According to (8), the following model conducts the data encryption following the second encryption scenario after permutation of operands:

$$C(x,\gamma)=\begin{cases}\begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\end{bmatrix}, & if\ \ \gamma_1=0;\gamma_2=0\\ \begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=0;\gamma_2=1\\ \begin{bmatrix}x_1\\ x_1\oplus x_2\end{bmatrix}, & if\ \ \gamma_1=1;\gamma_2=0\\ \begin{bmatrix}x_1\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=1;\gamma_2=1\end{cases}=\varphi\cdot \tag{9}$$

where φ is the result of the XOR cipher encryption (γ).

Since the encryption models in the first (6) and the second (9) encryption scenarios are different, the cryptograms z and φ are different as well.

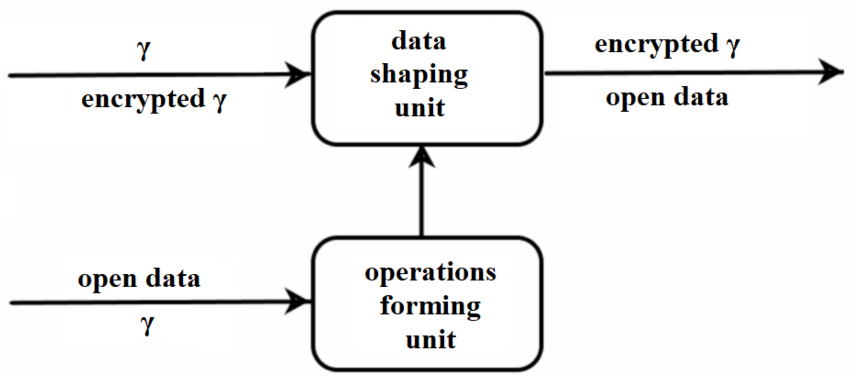
To decrypt the cryptogram φ, which was acquired as a result of the second encryption scenario, we will use an inverse CET-operation. For the following single-operand CET-operation $C(x)=\begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\end{bmatrix}$ the following operation $C^{/}(x)=\begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\oplus 1\end{bmatrix}$ is inverse, for the $C(x)=\begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\oplus 1\end{bmatrix}$ the following operation $C^{/}(x)=\begin{bmatrix}x_1\oplus 1\\ x_1\oplus x_2\end{bmatrix}$ is inverse, for the $C(x)=\begin{bmatrix}x_1\\ x_1\oplus x_2\end{bmatrix}$ the following operation $C^{/}(x)=\begin{bmatrix}x_1\\ x_1\oplus x_2\end{bmatrix}$ is inverse, and for $C(x)=\begin{bmatrix}x_1\\ x_1\oplus x_2\oplus 1\end{bmatrix}$ the following operation $C^{/}(x)=\begin{bmatrix}x_1\\ x_1\oplus x_2\oplus 1\end{bmatrix}$ is inverse. Thus, we can acquire the following results:

$$C^{/}(\varphi,\gamma)=\begin{cases}\begin{bmatrix}\varphi_1\oplus 1\\ \varphi_1\oplus\varphi_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=0;\gamma_2=0\\ \begin{bmatrix}\varphi_1\oplus 1\\ \varphi_1\oplus\varphi_2\end{bmatrix}, & if\ \ \gamma_1=0;\gamma_2=1\\ \begin{bmatrix}\varphi_1\\ \varphi_1\oplus\varphi_2\end{bmatrix}, & if\ \ \gamma_1=1;\gamma_2=0\\ \begin{bmatrix}\varphi_1\\ \varphi_1\oplus\varphi_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=1;\gamma_2=1\end{cases}=x\cdot \tag{10}$$

The encrypted XOR cipher is transmitted into the first operand. A normal XOR cipher is transmitted into the second operand. This process occurs to decrypt the encrypted XOR cipher according to the specified model of a CET-operation.

Since models for data decryption (7) and (10) are different, then it is necessary to include one model for data encryption and two models for data decryption into a cryptographic system that executes the first and the second encryption scenarios.

Structure of the device suitable for the execution of the second stream encryption scenario is illustrated in Picture. 2.



Picture 2 Structure of the device suitable for the execution of the second stream encryption scenario

We will now explain the functionality of this device. At first, the XOR cipher is transmitted to the data entry point of the data transformation unit. The plaintext information is transmitted to the data entry point of the data shaping unit. Depending on the bit value of this plaintext information within the data shaping unit, the sequential number of a single-operand CET-operation is defined. This sequential number is then transmitted to the data shaping unit. The operations shaping unit can be described as a decoder of the plaintext data code.

The transmitted XOR cipher is encrypted by the single-operand CET-operation from a tuple of a two-operand CET-operation according to its sequential number, which is acquired from the operations shaping unit. The whole process takes place within a data transformation unit. As a result, an encrypted (protected) XOR cipher is transmitted from the data output point of the data transformation unit to the data output point of the device.

We will now explain the functionality of this device during the process of decryption of encrypted data. At first, the encrypted (protected) XOR cipher is transmitted to the data entry point of the data transformation unit. Then a regular XOR cipher is transmitted to the data entry point of the data shaping unit. Depending on the bit value of this XOR cipher within the data shaping unit, the sequential number of an inverse single-operand CET-operation is then defined. The operation itself is utilized to decrypt the encrypted XOR cipher and then transmit it to the data shaping unit. The transmitted protected XOR cipher is then decrypted by the inverse single-operand CET-operation from a tuple of an inverse two-operand CET-operation according to its sequential number. The whole process takes place within a data transformation unit. Afterward, plaintext information is transmitted from the data output point of the data transformation unit to the data output point of the device.

We will now define the conditions, under which a CET-operation can be considered suitable for the execution of the second encryption scenario.

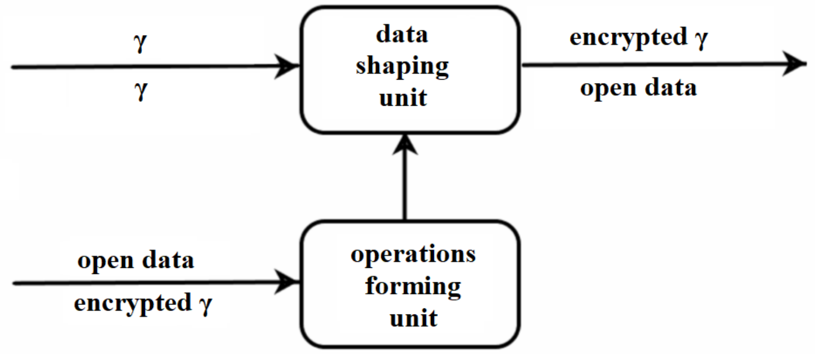
If a CET-operation is commutative and symmetrical (condition 1), or non-symmetrical (condition 2), then the CET-operation utilized for the data decryption and the results of the said encryption will remain unaltered by the permutation of operands.

If a CET-operation complies to conditions 3 – 6, then the following expression $C(x, y) \neq C(y, x)$ is valid, which results in $C^{/}(x, y) \neq C^{/}(y, x)$. Non-commutative CET-operations will alter the CET-operation utilized for the data decryption and the results of the said encryption when the permutation of operands occurs.

5. 3. The third stream encryption scenario

The third encryption scenario exists if there is a CET-operation $C(x, \gamma)$, and a CET-operation $C(\gamma, x)$, so that the following condition $C^{/}(\gamma, \varphi) \neq C^{/}(\varphi, \gamma)$ is valid for its inverse operations $C^{/}(\gamma, \varphi)$ and $C^{/}(\varphi, \gamma)$.

The difference between the third and second scenarios lies in the fact that the operands are permutated during decryption. Structure of the device suitable for the execution of the third stream encryption scenario is illustrated in Picture. 3.



Picture 3 Structure of the device suitable for the execution of the third stream encryption scenario

The functions of this device are very similar to those utilized for the execution of the second stream encryption scenario. The only difference lies in certain attributes of data decryption, whereas a regular XOR cipher is transmitted to the data entry point of the data transformation unit instead of an encrypted one. The same occurs regarding the data entry point of the operations forming unit.

We will now analyze the examples of a proper implementation of the third encryption scenario.

Let us assume that the data is encrypted according to the first encryption scenario based on the model of CET-operation (6).

We can encrypt the aforementioned data according to the second and the third scenarios only after the permutation of operands by the model of CET-operation (9).

To comply with the third encryption scenario, we must first permutate the operands prior to data decryption.

$$C(\gamma,x)=\begin{cases}\begin{bmatrix}\gamma_1\oplus 1\\ \gamma_1\oplus\gamma_2\end{bmatrix}, & if\ \ x_1=0;\ x_2=0\\ \begin{bmatrix}\gamma_1\oplus 1\\ \gamma_1\oplus\gamma_2\oplus 1\end{bmatrix}, & if\ \ x_1=0;\ x_2=1\\ \begin{bmatrix}\gamma_1\\ \gamma_1\oplus\gamma_2\end{bmatrix}, & if\ \ x_1=1;\ x_2=0\\ \begin{bmatrix}\gamma_1\\ \gamma_1\oplus\gamma_2\oplus 1\end{bmatrix}, & if\ \ x_1=1;\ x_2=1\end{cases}=\begin{cases}\begin{bmatrix}x_1\oplus 1\\ x_2\end{bmatrix}, & if\ \ \gamma_1=0;\ \gamma_2=0\\ \begin{bmatrix}x_1\oplus 1\\ x_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=0;\ \gamma_2=1\\ \begin{bmatrix}x_1\\ x_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=1;\ \gamma_2=0\\ \begin{bmatrix}x_1\\ x_2\end{bmatrix}, & if\ \ \gamma_1=1;\ \gamma_2=1\end{cases}$$

This permutation results in the acquisition of a model of modified CET-operation, the execution results of which are required to be decrypted. This process is illustrated by the following expression:

$$C(x,\gamma)=\begin{cases}\begin{bmatrix}x_1\oplus 1\\ x_2\end{bmatrix}, & if\ \ \gamma_1=0;\ \gamma_2=0\\ \begin{bmatrix}x_1\oplus 1\\ x_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=0;\ \gamma_2=1\\ \begin{bmatrix}x_1\\ x_2\oplus 1\end{bmatrix}, & if\ \ \gamma_1=1;\ \gamma_2=0\\ \begin{bmatrix}x_1\\ x_2\end{bmatrix}, & if\ \ \gamma_1=1;\ \gamma_2=1\end{cases}=z \tag{11}$$

Because the encryption model (6) is identical to the model used to transform the incoming data before its decryption according to the third scenario (11), both scenarios will deal with the decryption of the same cryptogram z. Since CET-operations (6) and (11) are symmetrical, then it is possible to decrypt the data by using models (7) and (6).

By analyzing this example, we can conclude that the permutation of operands during decryption in the third scenario provides an opportunity to utilize only one inverse CET-operation instead of two in the second encryption scenario.

This conclusion is based on our research of the cryptographic systems models created based on the non-commutative symmetrical CET-operation. We will now attempt to prove its validity in terms of its application to the cryptographic systems with the non-commutative non-symmetrical CET-operation.

To do so, we will analyze another example. Let us assume that the data encryption is conducted based on a non-commutative non-symmetrical CET-operation.

$$C(x,\gamma)=\begin{cases}\begin{bmatrix}x_1\oplus x_2\\ x_1\oplus 1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=0\\ \begin{bmatrix}x_1\oplus x_2\oplus 1\\ x_1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=1\\ \begin{bmatrix}x_1\oplus x_2\\ x_1\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=0\\ \begin{bmatrix}x_1\oplus x_2\oplus 1\\ x_1\oplus 1\end{bmatrix}, & f\ \gamma_1=1;\ \gamma_2=1\end{cases}=z\cdot \tag{12}$$

According to the first encryption scenario, the data decryption is conducted by utilizing the model below.

$$C^{/}(z,\gamma)=\begin{cases}\begin{bmatrix}z_1\oplus z_2\oplus 1\\ z_2\oplus 1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=0\\ \begin{bmatrix}z_2\\ z_1\oplus z_2\oplus 1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=1\\ \begin{bmatrix}z_2\\ z_1\oplus z_2\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=0\\ \begin{bmatrix}z_2\oplus 1\\ z_1\oplus z_2\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=1\end{cases}=x \tag{13}$$

We will now permutate the operands within the CET-operation used for encryption in the first scenario.

$$C(\gamma,x)=\begin{cases}\begin{bmatrix}\gamma_1\oplus\gamma_2\\ x_1\oplus 1\end{bmatrix}, & if\ x_1=0;\ x_2=0\\ \begin{bmatrix}\gamma_1\oplus\gamma_2\oplus 1\\ \gamma_1\end{bmatrix}, & if\ x_1=0;\ x_2=1\\ \begin{bmatrix}\gamma_1\oplus\gamma_2\\ \gamma_1\end{bmatrix}, & if\ x_1=1;\ x_2=0\\ \begin{bmatrix}\gamma_1\oplus\gamma_2\oplus 1\\ \gamma_1\oplus 1\end{bmatrix}, & if\ x_1=1;\ x_2=1\end{cases}=\begin{cases}\begin{bmatrix}x_2\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=0\\ \begin{bmatrix}x_2\oplus 1\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=1\\ \begin{bmatrix}x_2\oplus 1\\ x_1\oplus x_2\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=0\\ \begin{bmatrix}x_2\\ x_1\oplus x_2\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=1\end{cases}$$

This results in the second and third encryption scenarios using a non-commutative non-symmetrical CET-operation for data encryption. This is illustrated by the expression below.

$$C(x,\gamma)=\begin{cases}\begin{bmatrix}x_2\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=0\\ \begin{bmatrix}x_2\oplus 1\\ x_1\oplus x_2\oplus 1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=1\\ \begin{bmatrix}x_2\oplus 1\\ x_1\oplus x_2\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=0\\ \begin{bmatrix}x_2\\ x_1\oplus x_2\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=1\end{cases}=\varphi \qquad (14)$$

We will now permutate the operands before the data decryption within a model used for encryption (14) in the third scenario.

$$C(\gamma,x)=\begin{cases}\begin{bmatrix}\gamma_2\\ \gamma_1\oplus\gamma_2\oplus 1\end{bmatrix}, & if\ x_1=0;\ x_2=0\\ \begin{bmatrix}\gamma_2\oplus 1\\ \gamma_1\oplus\gamma_2\oplus 1\end{bmatrix}, & if\ x_1=0;\ x_2=1\\ \begin{bmatrix}\gamma_2\oplus 1\\ \gamma_1\oplus\gamma_2\end{bmatrix}, & if\ x_1=1;\ x_2=0\\ \begin{bmatrix}\gamma_2\\ \gamma_1\oplus\gamma_2\end{bmatrix}, & if\ x_1=1;\ x_2=1\end{cases}=\begin{cases}\begin{bmatrix}x_1\oplus x_2\\ x_1\oplus 1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=0\\ \begin{bmatrix}x_1\oplus x_2\oplus 1\\ x_1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=1\\ \begin{bmatrix}x_1\oplus x_2\\ x_1\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=0\\ \begin{bmatrix}x_1\oplus x_2\oplus 1\\ x_1\oplus 1\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=1\end{cases}$$

The aforementioned permutation of operands results in the acquisition of a modified encryption model. This model will then be decrypted according to the third scenario. This process is illustrated by the expression below.

$$C(x,\gamma)=\begin{cases}\begin{bmatrix}x_1\oplus x_2\\ x_1\oplus 1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=0\\ \begin{bmatrix}x_1\oplus x_2\oplus 1\\ x_1\end{bmatrix}, & if\ \gamma_1=0;\ \gamma_2=1\\ \begin{bmatrix}x_1\oplus x_2\\ x_1\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=0\\ \begin{bmatrix}x_1\oplus x_2\oplus 1\\ x_1\oplus 1\end{bmatrix}, & if\ \gamma_1=1;\ \gamma_2=1\end{cases}=z \qquad (15)$$

Since models of CET-operations (15) and (12) are identical, it confirms our hypothesis regarding the first and the third scenarios, whereas it is enough to implement one model of data encryption and one model of data decryption. The model (13) confirms this. Permutation of operands before the encryption and decryption alters the final cryptogram, which is transmitted through the open channel. Recurring permutation of operands before the decryption negates the results of the previous permutation of operands before the encryption.

VI. DISCUSSION OF THE MODELING RESULTS OF THE PROCESSES RELATED TO THE DATA TRANSFORMATION IN DIFFERENT STREAM ENCRYPTION SCENARIOS

The first stream encryption scenario does not allow permutation of operands within a CET-operation. It can operate with any CET-operation that has a corresponding inverse CET-operation. If the same CET-operation is utilized for the direct and inverse data transformation in a cryptographic system, then this CET-operation must be symmetrical regardless of the possibility (or the lack thereof) to permute the operands. The CET-operations that allow permutation of operands must correspond with the following conditions: these operations should be either commutative symmetrical (condition 1); non-commutative symmetrical (condition 3); non-commutative symmetrical/non-symmetrical (condition 5). The operations for the direct and inverse cryptographic transformation will not be identical if the CET-operations, which allow permutation of operands but correspond with the other conditions (2, 4, and 6), are used in the first scenario. In this case, different CET-operations are used by the cryptographic system for data encryption and decryption.

However, we believe that it is unnecessary to limit the cryptographic system functioning according to the first scenario to either the commutative or non-commutative operations. The aforementioned limitation drastically reduces the possible sets of CET-operations. This results in a decrease in the variability of the cryptographic system and its cryptographic integrity.

The second stream encryption scenario allows permutation of operands within a CET operation. In this case, the permutation of operands modifies the CET operation, which results in the acquisition of the two different cryptograms. Decryption of these cryptograms requires different inverse operations.

Only the commutative and non-commutative CET-operations allow permutation of operands.

Permutation of operands will not alter the cryptogram if a CET-operation is commutative. Thus, the second encryption scenario will partially replicate the first encryption scenario, with the difference being the fact that it is limited to operating only with the CET-operations, which allow the permutation of operands. Encryption and decryption require only one CET-operation under condition 1 (usage of commutative symmetrical operation). Under condition 2 (usage of commutative non-symmetrical operation), however, encryption requires a direct CET operation, while decryption requires an inverse CET-operation.

Permutation of operands will alter the cryptogram if a CET-operation is non-commutative. Symmetry or non-symmetry of CET-operations used for the direct and inverse cryptographic transformation is not considered a defining factor for using two CET-operations for the data decryption.

In comparison to the first scenario, the second scenario limits the variability of the cryptographic system by forcing it to operate only with the non-commutative CET-operations. However, permutation of operands doubles the amount of single-operand CET-operations (discrete models of substitution tables), which serve as the foundation for pseudo-random data encryption and decryption. Therefore, an increase in the number of substitution tables randomly chosen within the cryptographic system will increase the complexity of the cryptographic analysis.

The main disadvantage of the second stream encryption scenario revolves around the complexity of creating a model of CET-operation after the permutation of operands followed by an increase in bitness. For instance, there is no unified mathematical apparatus suitable for the description of all possible permutations for three-bit single-operand CET-operations . Tuples of single-operand operations with the defined attributes, which can be described by the unified model, may be chosen to create a two-operand CET-operation. In this case, these tuples of single-operand operations will be affected by the permutation of operands and will be considered pseudo-random after such permutation occurs. Thus, identifying the operation of an inverse transformation for the operation with the permutated operands is a complex task.

The third stream encryption scenario is a modified version of the second stream encryption scenario. In the third scenario, permutations of operands occur during both the encryption and decryption. Conditions required for the existence and implementation of both the second and the third scenarios are identical, This results in alteration of a cryptogram when permutation of operands occurs in the third scenario. Even if the operands are permutated twice in a row within the model of a two-operand CET operation, the model remains unaltered. A recurring permutation of operands before the decryption modifies the model used for the encryption of an operation in a special way, so that this model becomes identical to the encryption operation before the permutation of operands. Since models for data transformation before decryption are identical in the first and the third scenarios, models of operations for the inverse data transformation are identical as well. If permutation of operands is not taken into account, models of CET-operations for encryption and decryption will also be identical. We believe that it is necessary to divide the encryption and decryption in the first and the third scenarios into two stages. The first stage revolves around the implementation of a CET-operation necessary for permutation of operands. The mode of operation of the aforementioned operation depends on the encryption scenario. The second stage implements the data transformation operation.

Compared to the second, the third scenario requires only one CET-operation of an inverse cryptographic transformation for its implementation. Utilization of one operation of an inverse cryptographic transformation allows negating the primary downside of the second stream encryption scenario. The aforementioned downside is the complexity of creating a model of CET-operation for data decryption with the operands permutated during encryption.

Despite utilizing the same CET-operations for the encryption and decryption, the difference between the first and the third stream encryption scenarios is the final cryptogram transmitted through the open channel.

During creation of cryptographic systems, two-operand CET-operations are synthesized from the single-operand operations, which comply with the specified conditions for the quality of data transformation [17], [24]. However, the encryption results acquired from the permutation of operands are unable to ensure the guaranteed quality of a cryptographic transformation due to a pseudo-random tuple of the single-operand CET-operations. We believe this to be the primary downside of this research. One of its main achievements, however, is the establishment of correlations between CET-operations acquired as a result of the permutation of operands. These findings will prove beneficial in regards to generating the two-operand CET-operations to an accuracy of permutation of the second operand [22], which are then utilized in the first and the third scenarios.

VII. Conclusions

1. The research of CET-operations, which allow permutation of operands, allowed us to categorize these operations as commutative and non-commutative, symmetrical and non-symmetrical. We have defined six groups of CET-operations in total. The attributes of these groups affect the process of the data transformation in stream ciphers.

2. We have defined the three possible scenarios of stream encryption, including their limitations, which consider the usage of the categorized groups of CET-operations. We have then researched these stream encryption scenarios in detail.

3. The results of modeling the processes of data transformation allow us to prove the operating efficiency and the correct operation of cryptographic systems in different encryption scenarios. In addition, these results have proven the hypothesis about the possibility of improving the stream encryption systems with CET-operation that allow permutation of operands.

4. The introduced and analyzed structures of the devices based on the aforementioned stream encryption scenarios also reflect the attributes of their technical implementation. They can affect the practical choice of the multibit two-operand CET-operations, which will then be used as the foundation for creating the improved limited resources stream encryption systems.

5. The conducted analysis of the defined stream encryption scenarios and their execution attributes allows us to make numerous conclusions.

- The first stream encryption scenario is the most versatile in terms of choosing a CET-operation during creation of a cryptographic system. The aforementioned scenario is also the simplest one in terms of its technical execution;
- Compared to the first, the second stream encryption scenario requires an additional permutation of operands before encryption, as well as usage of two CET-operations for decryption. It doubles the number of substitution tables, but is also substantially more complex due to the necessity to create the second CET-operation for data decryption;
- The third stream encryption scenario is based on the permutation of operands before encryption and decryption. Similarly to the second scenario, it also doubles the amount of the substitution tables. However, the third scenario does not require an additional CET-operation for data decryption similar to the first scenario.

We believe that practical implementation of the aforementioned information is best done by combining the first and the third encryption scenarios during the creation of the limited resources cryptographic systems.

References

[1] Zheng, Z.,Tian, K. & Liu, F. (2023). Modern Cryptography Volume 2. A Classical Introduction to Informational and Mathematical Principle. Springer: Singapore. https://doi.org/10.1007/978-981-19-7644-5

[2] Semenov, S., Davydov, V., Kuchuk, N. & Petrovskaya, I. (2021). Software security threat research 31st International Scientific Symposium Metrology and Metrology Assurance, MMA 2021. https://doi.org/10.1109/MMA52675.2021.9610877

[3] Kumar, C., Prajapati, S. & Verma, R. (2022). A Survey of Various Lightweight Cryptography Block ciphers for IoT devices. IEEE International Conference on Current Development in Engineering and Technology (CCET), Bhopal, India, 2022, 1-6. https://doi.org/10.1109/CCET56606.2022.10080556.

[4] Khaled Salah Mohamed (2020). New Frontiers in Cryptography. Quantum, Blockchain, Lightweight, Chaotic and DNA. Springer International Publishing. Springer, Cham, 104. https://doi.org/10.1007/978-3-030-58996-7

[5] Thomas Xuan Meng, W. (2020). Buchanan. Lightweight Cryptographic Algorithms on Resource-Constrained Devices. Preprints. https://doi.org/10.20944/PREPRINTS202009.0302.V1

[6] Zakaria, A., Azni, A., Ridzuan, F., Zakaria, N. & Maslina, H. (2023). Daud Systematic literature review: Trend analysis on the design of lightweight block cipher. IEEE Journal of King Saud University - Computer and Information Sciences. 35, Issue 5, May, 101550. https://doi.org/10.1016/j.jksuci.2023.04.003

[7] Yalamuri, G., Honnavalli, P. & Eswaran, S. (2022). A Review of the Present Cryptographic Arsenal to Deal with Post-Quantum Threats. Procedia Comput. Sci. 215, 834–845. https://doi.org/10.1016/j.procs.2022.12.086

[8] Zakaria, A., et al. (2023). Systematic literature review: Trend analysis on the design of lightweight block cipher. Journal of King Saud University - Computer and Information Sciences, 35(5), 101550. https://doi.org/10.1016/j.jksuci.2023.04.003

[9] Khudoykulov, Z. (2024). A Comparison of Lightweight Cryptographic Algorithms. In: Aliev, R.A., et al. 12th World Conference "Intelligent System for Industrial Automation" (WCIS-2022). WCIS 2022. Lecture Notes in Networks and Systems, vol 912. Springer, Cham. https://doi.org/10.1007/978-3-031-53488-1_36

[10] Yasmin, N. & Gupta, R. (2023). Modified lightweight cryptography scheme and its applications in IoT environment. Int. j. inf. tecnol. 15, 4403–4414 https://doi.org/10.1007/s41870-023-01486-2

[11] Lohachab, A., Lohachab, A. & Jangra A. (2020). A comprehensive survey of prominent cryptographic aspects for securing communication in post-quantum IoT networks Internet of Things, March 2020, 9, 100174. https://doi.org/10.1016/j.iot.2020.100174

[12] Haythem, P., Zorkta, Y., Allawi, D. & Al-Nakkar, M. (2020). Improved lightweight encryption algorithm (ILEA) International Conference for Emerging Technology, IEEE 2020, 1-4. https://doi.org/10.1109/INCET49848.2020.9154170

[13] Sabani, M., et al. (2023). Evaluation and Comparison of Lattice-Based Cryptosystems for a Secure Quantum Computing Era. Electronics, 12(12), 2643. https://doi.org/10.3390/electronics12122643

[14] Zhang, Y., Li, P. & Huang, R. (2019). Lightweight Quantum Encryption for Secure Transmission of Power Data in Smart Grid. In IEEE 2019, Access, 7, 36285-36293. DOI: 10.1109/ACCESS.2019.2893056

[15] Gasser, L. (2023). Post-quantum Cryptography. In: Mulder, V., Mermoud, A., Lenders, V., Tellenbach, B. (eds) Trends in Data Protection and Encryption Technologies. Springer, Cham. https://doi.org/10.1007/978-3-031-33386-6_10

[16] Jancarczyk, D., Rudnytskyi, V., Breus, R., Pustovit, M., Veselska, O, & Ziubina, R. (2020). Two-Operand Operations of Strict Stable Cryptographic Coding with Different Operands' Bits. (2020) IEEE 5th International Symposium on Smart and Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS). IEEE; 2020 Sep 17; 247-254.

[17] Rudnytskyi, V., Korchenko, O., Lada, N., Ziubina, R., Wieclaw, L. & Hamera, L. (2022). Cryptographic encoding in modern symmetric and asymmetric encryption. 2022 IEEE 26th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2022) Procedia Computer Science 207, 54–63. https://doi.org/10.1016/j.procs.2022.09.037

[18] Rudnytskyi, V., Lada, N., Pochebut, M., Melnyk, O. & Tarasenko, Ya. (2023). Increasing the cryptographic strength of CETencryption by ensuring the transformation quality of the information block. 13th International Conference on Dependable Systems, Services and Technologies (DESSERT), Oct. 13-15, 2023, Athens, Greece. 1-6. https://doi.org/10.1109/DESSERT61349.2023.10416546

[19] Lada, N. & Kozlovska, S. (2018). Applying cryptographic addition operations by module two with accuracy of permutation in stream ciphers. Control, Navigation and Communication Systems. Academic Journal. Poltava: National University "Yuri Kondratyuk Poltava Polytechnic", 1 (47), 127-130. https://doi.org/10.26906/SUNZ.2018.1.127

[20] Rudnytskyi, V., Lada, N. & Kozlovska, S. (2018). Technology of two operand operations construction of information cryptographic transformation by modeling results. Advanced Information Systems, 2 (4), C.26-30. http://ais.khpi.edu.ua/article/viewFile/2522-9052.2018.4.04/151747

[21] Rudnytskyi, V., Babenko, V., Lada, N., Tarasenko, Ya. & Rudnytska, Yu. (2022). Constructing symmetric operations of cryptographic information encoding. Workshop on Cybersecurity Providing in Information and Telecommunication Systems (CPITS II 2021), Oct. 26, 2021. Kyiv, Ukraine: CEUR Workshop Proceedings, 2022, 182–194. ISSN 1613-0073

[22] Breus, R. (2019). Synthesising the two-bit two-operand operations of strict stable cryptographic coding by the second operand's conversion. Control, Navigation and Communication Systems. Academic Journal. – Poltava: National University "Yuri Kondratyuk Poltava Polytechnic", 5 (57), 29–32. https://doi.org/10.26906/SUNZ.2019.5.029

[23] Rudnitsky, V., Berdibayev, R., Breus, R., Lada, N. & Pustovit, M. (2019). Synthesis of reverse two-bit dual-operated strictly straight cryptographic coding on the basis of another operation. Advanced Information Systems, Kharkiv: NTU "KhPI", 3 (4), 109–114. http://doi.org/10.20998/2522-9052.2019.4.16

[24] Rudnytskyi, V., Lada N., Kushuk, G. & Pidlasyi, D. (2024). Architecture of CET-operations and stream encryption technologies: Monograph. Cherkasy: publisher Ponomarenko R.V., 374 p. ISBN 978- 978-966-2554-81. https://dndivsovt.com/index.php/monograph/issue/view/22/22