Cryptographic Key Exchange Method for Data Factorial Coding

Emil Faure [1][0000-0002-2046-481X] and Anatoly Shcherba [1][0000-0002-3049-3497],
Yevhen Vasiliu [2][0000-0002-8582-285X] and Andriy Fesenko [3][0000-0001-5154-5324]

[1] Cherkasy State Technological University, Cherkasy, Ukraine
[2] O.S. Popov Odessa National Academy of Telecommunication, Odessa, Ukraine
[3] Taras Shevchenko National University of Kyiv, Kyiv, Ukraine
e.faure@chdtu.edu.ua

Abstract. The paper proposes a new cryptographic key exchange method. The basic idea of the proposed method is to use a permutation of a given set as a transformation object. The mathematical background of the method is the property of permutations to be decomposed into the product of disjoint cycles, the property of unique factorization of the product of disjoint cycles raised to the powers smaller than their order, and the complexity of factorization of the product of permutations whose cycles are noncommutative. The conditions to be met by the transformation parameters are defined. The concepts of cycle- and block-noncommutative permutations are introduced. These properties of two permutations known to all participants in information exchange are sufficient for the correct operation of the method. The key space cardinality of the values of cycles' exponents of two open permutations is investigated. It is shown that this cardinality is maximized if the disjoint cycles in the decomposition of open permutations are 3-cycles. The block diagram of a cryptographic system that implements the proposed method is investigated. Its work is described. The proposed method and system make it possible to generate a cryptographic key for information factorial coding without using a secure communication channel. They can also be used to form a non-permutation key.

Keywords: cryptography, method, key exchange, permutation, factorial coding.

1 Introduction

Several information security tasks, such as providing authentication, confidentiality, and integrity, are solved simultaneously when transmitting information in communication systems and systems of managing various technological processes. In particular, this is observed when transporting information arrays through noisy communication channels, including through tunneled protocols on computer networks. A separate solution of these tasks is associated with the use of different mathematical methods and algorithms. This leads to an increase in the load on means of information transforming and requirements for their productivity, to an increase in introduced redundancy, and, as a consequence, to a decrease of a relative transmission rate. These

 CybHyg-2019: International Workshop on Cyber Hygiene, Kyiv, Ukraine, November 30, 2019.

circumstances actualize the problem of providing information security during its storing and transmitting in telecommunication systems and networks by integrating channel coding and cryptographic security methods.

The factorial coding methodology [1] provides creation and study of methods for combining the following types of data protection in a single procedure:

- protection against errors caused by noise in communication channel;
- protection against unauthorized data modification;
- protection against unauthorized data access.

Separable factorial codes (FC) [2, 3] provide information integrity control and do not ensure its confidentiality. Code combinations of such codes contain separate information and control parts. The control part is a permutation π or its part. A permutation of a set of M elements is a bijection from a finite set X of cardinality M into itself. We denote the elements of a finite set X by nonnegative integers from 0 to $M-1$. Then $X=\{0,1,\ldots,M-1\}$ and the permutation π will be written as a sequence of elements of the set X, where each of the numbers $\{0,1,\ldots,M-1\}$ is applied only once (without gaps and repetitions).

Non-separable FCs [4-6] provide protection against unauthorized data access and are able to detect and correct a significant part of errors. Code combinations of such codes are formed by converting an information word $A(x)$ into a permutation π. They do not have separate information and control parts (as, for example, CRC codes have). In addition, non-separable FCs are self-synchronizing codes. A self-synchronization property of the code removes the problem of frame synchronization and eliminates the need for a delimiter in a data block header structure. Because of this, an amount of redundancy introduced during coding is reduced.

The key sequences for factorial coding methods are permutations. These key permutations are kept secret.

When encoding and decoding factorial codes, the same permutation is used. Therefore, factorial coding methods have the disadvantage of symmetric cryptographic transformation, such as the need to form a secret communication channel to transmit key data.

In [7], Whitfield Diffie and Martin Hellman proposed the first and the best-known key agreement method. This method and the corresponding cryptographic device is patented in the US with Ralph Merkle [8]. The proposed protocol (known as the Diffie-Hellman key exchange) allows two parties to form a shared secret key based on their own secret keys and each other's public keys. Cryptanalyst knows only public keys of two parties. He is unable to calculate their shared secret key within an acceptable amount of time and limited performance of computing facilities.

This method uses a vulnerable to eavesdropping channel. However, it does not provide user authentication. Therefore, it is vulnerable to the man-in-the-middle attack. To solve this problem, a number of modifications to the method have been proposed. In particular, they are outlined in [9, 10].

A cryptographic strength of the Diffie-Hellman protocol and its modifications, as well as the El Gamal algorithm [11], is based on the complexity of the discrete logarithm problem. At the same time, quantum computers using the Shore algorithm [12]

will easily solve the problems of discrete logarithmization and factorization of integers.

To date, a number of post-quantum key exchange protocols have been developed, including Supersingular isogeny Diffie–Hellman key exchange [13], NTRU [14], Ring Learning with Errors Key Exchange [15].

One of the most promising modern research areas in the field of cryptographic key exchange is their quantum distribution [16-18]. However, to date, this direction has limitations on transmission distance and communication network structure. In addition, all of the above key exchange methods are not adapted for data factorial coding that uses permutations.

The purpose of this work is to provide the ability to generate cryptographic keys for data factorial coding without using a secret communication channel. The key exchange method should have the prerequisites for use in post-quantum cryptography [19-25].

The work is organized as follows. In Section 2, we describe our cryptographic key exchange method. In Section 3, we analyze a key space cardinality and conditions to its maximization. A cryptographic system and a description of its work are given in Section 4. In the last section, we present the conclusion.

2 Construction of Cryptographic Key Exchange Method

The essence of the proposed approach is the following.

- Each exchange party generates a shared key by converting permutations received from the other party. In this case, the direct transformations of permutations on each party of information exchange are easy to implement, and the inverse transformations are almost impossible to implement.
- The generated key is used for data factorial coding during forward and reverse transformations of messages transmitted by an insecure communication channel.

The method for factorial coding key exchange over an open channel involves the following procedures.

- Two parties Alice and Bob know two permutations, α and β, and their decompositions into the products of disjoint cycles: $\alpha = \prod_{i=1}^{n(\alpha)} \alpha_i$, $\beta = \prod_{j=1}^{n(\beta)} \beta_j$.
- Alice generates a random secret key in the form of two vectors $\overline{k} = \left(k_1, k_2, \ldots, k_{n(\alpha)}\right)$ and $\overline{m} = \left(m_1, m_2, \ldots, m_{n(\beta)}\right)$ of dimensions $n(\alpha)$ and $n(\beta)$, moreover $0 \le k_i \le l(\alpha_i) - 1$, $0 \le m_j \le l(\beta_j) - 1$, where $l(\alpha_i)$, $l(\beta_j)$ are the orders of cycles α_i and β_j, respectively.

▪ Bob generates another random secret key in the form of two vectors $\bar{t} = \left(t_1, t_2, \ldots, t_{n(\alpha)}\right)$ and $\bar{s} = \left(s_1, s_2, \ldots, s_{n(\beta)}\right)$ of dimensions $n(\alpha)$ and $n(\beta)$, and $0 \le t_i \le l(\alpha_i) - 1,\ 0 \le s_j \le l(\beta_j) - 1$.

▪ Alice forms a permutation $Y_1 = \sigma_1 \cdot \omega_1$, where $\sigma_1 = \prod_{i=1}^{n(\alpha)} \alpha_i^{k_i}$, $\omega_1 = \prod_{j=1}^{n(\beta)} \beta_j^{m_j}$. Sends Y_1 to Bob.

▪ Bob forms a permutation $Y_2 = \sigma_2 \cdot \omega_2$, where $\sigma_2 = \prod_{i=1}^{n(\alpha)} \alpha_i^{t_i}$, $\omega_2 = \prod_{j=1}^{n(\beta)} \beta_j^{s_j}$. Sends Y_2 to Alice.

▪ Receiving Y_1, Bob computes the shared key $K = \sigma_2 \cdot Y_1 \cdot \omega_2$. Once getting Y_2, Alice obtains $K = \sigma_1 \cdot Y_2 \cdot \omega_1$.

The above operations and the procedure for their implementation ensure the achievement of the claimed technical result. It consists in the possibility of data factorial coding with its secure transmission over an insecure communication channel without prearrangement of a cryptographic key over a secret channel. This is possible because each party in the exchange of information independently forms a shared key on its side.

The proposed method is suitable for practical implementation.

Correctness. We now show that if Alice and Bob do the above steps, they will share an identical key K.

Alice computes $K_1 = \sigma_1 \cdot Y_2 \cdot \omega_1 = \sigma_1 \cdot (\sigma_2 \cdot \omega_2) \cdot \omega_1$ and Bob computes $K_2 = \sigma_2 \cdot Y_1 \cdot \omega_2 = \sigma_2 \cdot (\sigma_1 \cdot \omega_1) \cdot \omega_2$. Because of the property of associativity of reflections, $K_1 = \sigma_1 \cdot (\sigma_2 \cdot \omega_2) \cdot \omega_1 = (\sigma_1 \cdot \sigma_2) \cdot (\omega_2 \cdot \omega_1)$ and $K_2 = \sigma_2 \cdot (\sigma_1 \cdot \omega_1) \cdot \omega_2 = (\sigma_2 \cdot \sigma_1) \cdot (\omega_1 \cdot \omega_2)$. Since $\sigma_1 \cdot \sigma_2 = \sigma_2 \cdot \sigma_1 = \prod_{i=1}^{n(\alpha)} \alpha_i^{k_i + t_i}$ and $\omega_1 \cdot \omega_2 = \omega_2 \cdot \omega_1 = \prod_{j=1}^{n(\beta)} \beta_j^{m_j + s_j}$, $K_1 = K_2 = K$.

Necessary conditions. We now determine the conditions that the conversion parameters must satisfy.

First, a cryptanalyst should not be able to easily calculate the values of $\bar{k}$, $\bar{m}$, $\bar{t}$, and $\bar{s}$ by the known values of Y_1 and Y_2. Second, the pairs of values $\left(\bar{k}; \bar{m}\right)$, $\left(\bar{t}; \bar{s}\right)$ must mutually uniquely determine Y_1 and Y_2, respectively, that is $\left\{\left(\bar{k}; \bar{m}\right)\right\} \leftrightarrow \{Y_1\}$ and $\left\{\left(\bar{t}; \bar{s}\right)\right\} \leftrightarrow \{Y_2\}$.

We introduce the following definitions.

Definition 1. Permutations α and β are called cycle-noncommutative if the decompositions into the product of disjoint cycles $\alpha = \prod_{i=1}^{n(\alpha)} \alpha_i$, $\beta = \prod_{j=1}^{n(\beta)} \beta_j$ have the property $\alpha_i \beta_j \neq \beta_j \alpha_i$ for $\forall i, j$.

The property of cycle-noncommutativity of permutations α and β is sufficient to provide a high degree of mixing elements in the products $\sigma_1 \cdot \omega_1$ and $\sigma_2 \cdot \omega_2$. This complicates the cryptanalyst work.

Unique factorization property (condition of bijections $\left\{\left(\bar{k};\bar{m}\right)\right\} \leftrightarrow \{Y_1\}$ and $\left\{\left(\bar{t};\bar{s}\right)\right\} \leftrightarrow \{Y_2\}$).

Let permutations $\alpha = \prod_{i=1}^{n(\alpha)} \alpha_i$ and $\beta = \prod_{j=1}^{n(\beta)} \beta_j$ be cycle-noncommutative. Then the permutation $Y_1 = \sigma_1 \cdot \omega_1 = \prod_{i=1}^{n(\alpha)} \alpha_i^{k_i} \cdot \prod_{j=1}^{n(\beta)} \beta_j^{m_j}$ (or $Y_2 = \sigma_2 \cdot \omega_2 = \prod_{i=1}^{n(\alpha)} \alpha_i^{t_i} \cdot \prod_{j=1}^{n(\beta)} \beta_j^{s_j}$) is mutually uniquely determined by specifying the powers $\bar{k} = \left(k_1, k_2, \ldots, k_{n(\alpha)}\right)$ and $\bar{m} = \left(m_1, m_2, \ldots, m_{n(\beta)}\right)$ (or $\bar{t} = \left(t_1, t_2, \ldots, t_{n(\alpha)}\right)$ and $\bar{s} = \left(s_1, s_2, \ldots, s_{n(\beta)}\right)$) if:

1. $l(\alpha_i) \leq 2n(\beta) - 1$, $l(\beta_j) \leq 2n(\alpha) - 1$ for $\forall i, j$;
2. $0 \leq k_i \leq l(\alpha_i) - 1$, $0 \leq m_j \leq l(\beta_j) - 1$ for $\forall i, j$.

These conditions are sufficient.

Definition 2. Let permutations α and β be represented as the products of disjoint cycles: $\alpha = \alpha_1 \cdot \ldots \cdot \alpha_{i_1} \cdot \alpha_{i_1+1} \cdot \ldots \cdot \alpha_{i_2} \cdot \ldots \cdot \alpha_{i_{L-1}+1} \cdot \ldots \cdot \alpha_{i_L} = A_1 \cdot A_2 \cdot \ldots \cdot A_L$, $\beta = \beta_1 \cdot \ldots \cdot \beta_{j_1} \cdot \beta_{j_1+1} \cdot \ldots \cdot \beta_{j_2} \cdot \ldots \cdot \beta_{j_{L-1}+1} \cdot \ldots \cdot \beta_{j_L} = B_1 \cdot B_2 \cdot \ldots \cdot B_L$, where $A_l = \alpha_{i_{l-1}+1} \cdot \ldots \cdot \alpha_{i_l}$, $B_l = \beta_{j_{l-1}+1} \cdot \ldots \cdot \beta_{j_l}$, $1 \leq l \leq L$, $i_0 = 0$, $j_0 = 0$. Then the permutations α and β are called block-noncommutative if inequalities $\alpha_i \beta_j \neq \beta_j \alpha_i$, where $i_{l-1} + 1 \leq i \leq i_l$, $j_{l-1} + 1 \leq j \leq j_l$, are true for each pair of blocks $\{A_l; B_l\}$, $1 \leq l \leq L$.

The condition of block-noncommutativity of permutations α and β is more lenient than the condition of cycle-noncommutativity. At the same time, it implies cycle-noncommutativity between the corresponding blocks of permutations α and β and allows the elements within the blocks to be mixed in the results of the products $\sigma_1 \cdot \omega_1$ and $\sigma_2 \cdot \omega_2$.

Remark. Block-noncommutativity of permutations α and β retains the property of unique factorization of permutations of the type $Y=\sigma\cdot\omega=\prod_{i=1}^{n(\alpha)}\alpha_i^{t_i}\cdot\prod_{j=1}^{n(\beta)}\beta_j^{s_j}$ if each block $\{A_l;B_l\}$, $1\le l\le L$, is uniquely factorized.

3 Key Space Cardinality

We now evaluate a cardinality of key space for the proposed key exchange method. A secret key to a cryptosystem is a tuple $\left(\bar{k},\bar{m},\bar{t},\bar{s}\right)$ consisting of two pairs of vectors $\begin{cases}\bar{k}=\left(k_1,k_2,\ldots,k_{n(\alpha)}\right)\\ \bar{m}=\left(m_1,m_2,\ldots,m_{n(\beta)}\right)\end{cases}$ and $\begin{cases}\bar{t}=\left(t_1,t_2,\ldots,t_{n(\alpha)}\right)\\ \bar{s}=\left(s_1,s_2,\ldots,s_{n(\beta)}\right)\end{cases}$ that are Alice's and Bob's secret keys. The following conditions must be met: $0\le k_i,t_i\le l\left(\alpha_i\right)-1$, $0\le m_j,s_j\le l\left(\beta_j\right)-1$, where $l\left(\alpha_i\right)$, $l\left(\beta_j\right)$ are the orders of cycles α_i and β_j, respectively. Then the key space cardinality is equal to $\mu=\prod_{i=1}^{n(\alpha)}\left(l\left(\alpha_i\right)\right)^2\cdot\prod_{j=1}^{n(\beta)}\left(l\left(\beta_j\right)\right)^2=\left(\prod_{i=1}^{n(\alpha)}l\left(\alpha_i\right)\cdot\prod_{j=1}^{n(\beta)}l\left(\beta_j\right)\right)^2$.

We define the conditions under which the value μ will be maximized. Since the order of cyclic permutation is equal to its length, the following equalities are true: $\sum_{i=1}^{n(\alpha)}l\left(\alpha_i\right)=M\left(\alpha\right)$ and $\sum_{j=1}^{n(\beta)}l\left(\beta_j\right)=M\left(\beta\right)$, where $M\left(\alpha\right)$ and $M\left(\beta\right)$ are the lengths of permutations α and β, respectively. Then $\prod_{i=1}^{n(\alpha)}l\left(\alpha_i\right)\to\max$ and $\prod_{j=1}^{n(\beta)}l\left(\beta_j\right)\to\max$, when $l\left(\alpha_i\right)\to M\left(\alpha\right)/n\left(\alpha\right)$ and $l\left(\beta_j\right)\to M\left(\beta\right)/n\left(\beta\right)$ for $\forall i,j$, as well $n\left(\alpha\right)\to M\left(\alpha\right)/e$ and $n\left(\beta\right)\to M\left(\beta\right)/e$. Because $M\left(\alpha\right),M\left(\beta\right),n\left(\alpha\right),n\left(\beta\right),l\left(\alpha\right),l\left(\beta\right)\in Z$, then $\mu=\left(\prod_{i=1}^{n(\alpha)}l\left(\alpha_i\right)\cdot\prod_{j=1}^{n(\beta)}l\left(\beta_j\right)\right)^2\to\max$, when $M\left(\alpha\right)\bmod 3=0$, $M\left(\beta\right)\bmod 3=0$, $n\left(\alpha\right)=M\left(\alpha\right)/3$, $n\left(\beta\right)=M\left(\beta\right)/3$, and $l\left(\alpha_i\right)=l\left(\beta_j\right)=3$ for $\forall i,j$. Then $\mu=3^{2\left(n(\alpha)+n(\beta)\right)}=3^{2\left(M(\alpha)+M(\beta)\right)/3}$.

Thus, to achieve the maximum of key space cardinality, it is necessary to form permutations α and β as the product of disjoint 3-cycles. Under these conditions, the cardinality of the space of possible permutation α values is equal to

$\mu(\alpha) = 3^{-n(\alpha)} \cdot (M(\alpha))!$. The cardinality of the space of possible permutation β values depends on a method for its forming and requires additional research.

Consider a few examples.

Example 1. A permutation β cycle-noncommutative to a permutation α can be formed for $M(\alpha) = M(\beta)$ as follows. Each of the cycles β_j, $1 \le j \le n(\beta)$, contains one randomly selected element from each of the cycles α_i, $1 \le i \le n(\alpha)$. If $l(\alpha_i) = 3$, then $n(\beta) = 3$ and $l(\beta_j) = n(\alpha)$ for $\forall j$. Then the cardinality of the space of possible permutation β values is equal to $\mu(\beta) = (3!)^{n(\alpha)}$. Note that $l(\beta_j) = 3$ for $\forall j$ is possible only when $n(\alpha) = n(\beta) = 3$.

Example 2. A method of forming a permutation β block-noncommutative to a permutation α may be as follows. Disjoint cycles are grouped into blocks of three: $(\alpha_1, \alpha_2, \alpha_3), (\alpha_4, \alpha_5, \alpha_6), \ldots, (\alpha_i, \alpha_{i+1}, \alpha_{i+2}), \ldots, (\alpha_{n(\alpha)-2}, \alpha_{n(\alpha)-1}, \alpha_{n(\alpha)})$. The first element of a cycle β_i is selected randomly from the elements of a cycle α_i, the second element – from the elements of a cycle α_{i+1}, and the third – from a cycle α_{i+2}. The first element of a cycle β_{i+1} is randomly selected from the rest of the elements of a cycle α_i, the second element – from the rest elements of a cycle α_{i+1}, and the third – from a cycle α_{i+2}. The elements of a cycle β_{i+2} are uniquely defined. Note that this method of forming β requires $|n(\alpha)|_3 = 0$. Then the cardinality of the space of possible permutation β values is equal to $\mu(\beta) = \left((3!)^3\right)^{n(\alpha)/3} = (3!)^{n(\alpha)}$.

4 Cryptographic System Description

The method for factorial coding key exchange can be implemented in a cryptographic system, a block diagram of which is shown in Fig. 1.

Two-way communication between Alice 1 and Bob 2 is exchanged on an open insecure duplex (half-duplex) communication channel 7 using transceivers 6 and 8. Alice and Bob have factorial codecs 3 and 9, shared key generators 4 and 10, as well as their own secret key generators 5 and 11, respectively. Let Alice creates a plaintext S_o. Then codecs 3 and 9 perform transformations $FC_K(S_o)$ and FC_K^{-1} (the opposite of transformation FC_K), respectively. These transformations depend on the shared key, permutation K. Factorial codec 5 generates a codeword CW, which is transmitted by transceiver 6 through the open channel 7 to the Bob's transceiver 8. The codeword CW is decoded in the codec 9.

Own key generators 5 and 11 are independent generators based on random or pseudorandom processes. Alice's key generator 5 creates four signals: $\alpha, \beta, \bar{k}, \bar{m}$. The requirements for these variables are described above. Two permutations α and β

are sent over the insecure communication channel 7 to the Bob's shared key generator 10. Vectors $\bar{k}, \bar{m}$ are transmitted to the shared key generator 4 and are kept secret by Alice. Bob's own key generator 11 generates two signals $\bar{t}, \bar{s}$ and transmits them to the shared key generator 10. Vectors $\bar{t}$ and $\bar{s}$ are keep secret by Bob.

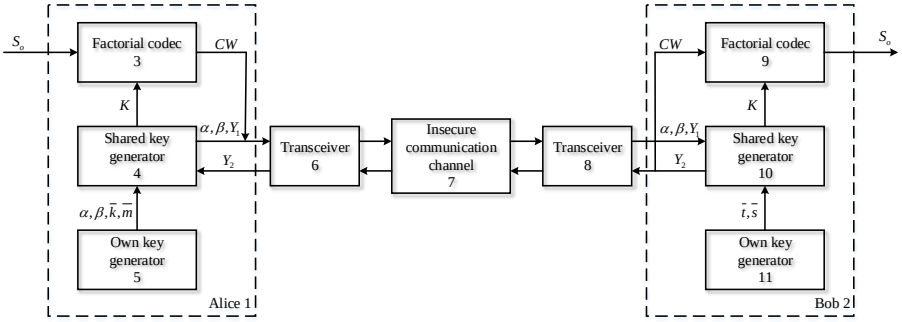

Fig. 1. A block diagram of a cryptographic system that transmits a cryptogram over an insecure communication channel

The shared key generators 4 and 10 form a common transformation key K. For this, the shared key generator 4 generates a signal Y_1 based on the values of signals $\alpha, \beta, \bar{k}, \bar{m}$ and transmits it to the shared key generator 10. In turn, the shared key generator 10 generates a signal Y_2 based on the values of signals $\alpha, \beta, \bar{t}, \bar{s}$ and transmits it to the shared key generator 4. Direct calculating of Y_1 and Y_2 values does not cause difficulties. Inverse calculating of $\bar{k}$ and $\bar{m}$ values from known α, β, and Y_1, as well as $\bar{t}$ and $\bar{s}$ from known α, β, and Y_2 is practically impossible.

Signal Y_1 is formed in such a way as to correspond to the permutation on the set $\{0,1,\ldots,M-1\}$ formed by the product of the other two permutations, σ_1 and ω_1. The permutation σ_1 is formed by exponentiation of permutation elements, which values are transmitted by signal α, to powers, which values are transmitted by signal $\bar{k} = \left(k_1, k_2, \ldots, k_{n(\alpha)}\right)$. The permutation ω_1 is formed by exponentiation of permutation elements, which values are transmitted by signal β, to powers, which values are transmitted by signal $\bar{m} = \left(m_1, m_2, \ldots, m_{n(\beta)}\right)$.

Signal Y_2 is formed in such a way as to correspond to the permutation on the set $\{0,1,\ldots,M-1\}$ formed by the product of the other two permutations, σ_2 and ω_2. The permutation σ_2 is formed by exponentiation of permutation elements, which values are transmitted by signal α, to powers, which values are transmitted by signal $\bar{t} = \left(t_1, t_2, \ldots, t_{n(\alpha)}\right)$. The permutation ω_2 is formed by exponentiation of permutation

elements, which values are transmitted by signal β, to powers, which values are transmitted by signal $\bar{s} = \left(s_1, s_2, \ldots, s_{n(\beta)}\right)$.

Receiving signal Y_2, the shared key generator 4 generates a signal K. It corresponds to the permutation formed by the product of permutations transmitted by signals σ_1, Y_2, and ω_1, and strictly in that order, $K = \sigma_1 \cdot Y_2 \cdot \omega_1$. Symbolically, the process of calculating a shared key by Alice can be represented as follows:

$$K = \sigma_1 \cdot Y_2 \cdot \omega_1 = \sigma_1 \cdot \left(\sigma_2 \cdot \omega_2\right) \cdot \omega_1 = \left(\sigma_1 \cdot \sigma_2\right) \cdot \left(\omega_2 \cdot \omega_1\right) = \prod_{i=1}^{n(\alpha)} \alpha_i^{k_i + t_i} \cdot \prod_{j=1}^{n(\beta)} \beta_j^{m_j + s_j} .$$

Receiving signal Y_1, the shared key generator 10 generates a signal K. It corresponds to the permutation formed by the product of permutations transmitted by signals σ_2, Y_1, and ω_2, and strictly in that order, $K = \sigma_2 \cdot Y_1 \cdot \omega_2$. Symbolically, the process of calculating a shared key by Bob can be represented as follows:

$$K = \sigma_2 \cdot Y_1 \cdot \omega_2 = \sigma_2 \cdot \left(\sigma_1 \cdot \omega_1\right) \cdot \omega_2 = \left(\sigma_1 \cdot \sigma_2\right) \cdot \left(\omega_2 \cdot \omega_1\right) = \prod_{i=1}^{n(\alpha)} \alpha_i^{k_i + t_i} \cdot \prod_{j=1}^{n(\beta)} \beta_j^{m_j + s_j} .$$

Signals K corresponding to the same permutation key formed by the shared key generators 4 and 10, are transmitted to the inputs of the codecs 3 and 9. There they are used to encode a plaintext S_o and decode a codeword CW, respectively.

Similar to [8], we can show that the scheme of practical implementation of the proposed method may be different from shown in Fig. 1. Signals α and β may be in the public domain and not generated by the key generator of one of the parties. In addition, there may be significantly more than two parties. It is then advisable to place the value Y_i of the i-th party to an open directory, a public file or directory, rather than transmit it between users each time. In this case, the two parties, i and j, that establish a secure connection form a shared encryption key by computing $K_{ij} = \sigma_i \cdot Y_j \cdot \omega_i$ and $K_{ji} = \sigma_j \cdot Y_i \cdot \omega_j$.

5 Conclusions

The proposed key exchange method allows forming a symmetric key, permutation, for data factorial coding without using a secure communication channel.

This method can be used not only for factorial coding tasks. This is explained by the fact that the obtained permutation, being the key for factorial code, corresponds to a certain number in the factorial number system. This number can easily be represented in any other number system (binary, decimal, etc.).

A detailed study of the cryptographic strength of the proposed key exchange method is an area for further research in this field.

6 Acknowledgments

The authors express their sincere appreciation to Ph.D., Associate Professor, Honorary Professor of Cherkasy State Technological University Valerii Shvydkyi for the full support of this area of work, constructive comments and suggestions when writing the work, and useful discussion of the results.

References

1. Faure, E.: Methodology of information security based on factorial data coding. Dr. Sc. Eng. Thesis, National Aviation University, Kyiv, Ukraine (2018).
2. Faure, E., Shvydkyi, V., Shcherba, A.: Information integrity control based on factorial number system. Journal of Baku engineering university – Mathematics and computer science 1(1), 3-13 (2017).
3. Faure, E., Shvydkyi, V., Shcherba, V.: Combined factorial coding and its properties. Radio Electronics, Computer Science, Control 3, 80-86 (2016).
4. Faure, E.: Factorial coding with data recovery. Visnyk Cherkaskogo derzhavnogo tehnologichnogo universitetu 2, 33-39 (2016).
5. Faure, E.: Factorial coding with error correction. Radio Electronics, Computer Science, Control 3, 130-138 (2017).
6. Faure, E., Shcherba, A., Kharin, A.: Factorial code with a given number of inversions. Radio Electronics, Computer Science, Control 2, 143-153 (2018).
7. Diffie, W., Hellman, M.: New directions in cryptography. IEEE Transactions on Information Theory (6), 644-654 (1976).
8. Hellman, M., Diffie, W., Merkle, R.: Cryptographic apparatus and method. US Patent 4,200,770 (1980).
9. Günther, C.: An identity-based key-exchange protocol. Advances in Cryptology – Eurocrypt 89, 1989. Lecture Notes in Computer Science, vol. 434, pp. 29-37. Springer-Verlag, Berlin/New York (1989).
10. Diffie, W., van Oorschot, P., Wiener, M.: Authentication and authenticated key exchanges. Designs, Codes and Cryptography 2(2), 107-125 (1992).
11. ElGamal, T.: A public-key cryptosystem and a signature scheme based on discrete logarithms. IEEE Transactions on Information Theory 31(4), 469–472 (1985).
12. Shor, P.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal on Computing 26(5), 1484-1509 (1997).
13. Jao D., De Feo L.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In: Yang BY. (eds) Post-Quantum Cryptography. PQCrypto 2011. Lecture Notes in Computer Science, vol. 7071, pp. 19-34. Springer, Berlin, Heidelberg (2011).
14. Initial recommendations of long-term secure post-quantum systems. PQCRYPTO.EU. Horizon 2020 ICT-645622.
15. Ding, J., Xie, X., Lin, X.: A simple provably secure key exchange scheme based on the learning with errors problem. Cryptology ePrint Archive, Report 2012/688 (2014).
16. Folger, T.: The quantum hack. Scientific American 314, 48-55 (2016).
17. Korchenko, O., Vasiliu, Ye., Gnatyuk, S.: Modern quantum technologies of information security against cyberterrorist attacks, Aviation 14(2), 58-69 (2010).
18. S. Gnatyuk, T. Zhmurko, P. Falat, Efficiency Increasing Method for Quantum Secure Direct Communication Protocols, Proceedings of the 2015 IEEE 8th International Confer-

ence on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2015), Warsaw, Poland, September 24-26, Vol. 1, 2015, pp. 468-472.
19. S. Gnatyuk, A. Okhrimenko, M. Kovtun, T. Gancarczyk, V. Karpinskyi, Method of Algorithm Building for Modular Reducing by Irreducible Polynomial, Proceedings of the 16th International Conference on Control, Automation and Systems, Oct. 16-19, Gyeongju, Korea, 2016, pp. 1476-1479.
20. Hu Z., Gnatyuk S., Kovtun M., Seilova N. Method of searching birationally equivalent Edwards curves over binary fields, Advances in Intelligent Systems and Computing, Vol. 754, pp. 309-319, 2019.
21. S. Gnatyuk, V. Kinzeryavyy, M. Iavich, D. Prysiazhnyi, Kh. Yubuzova, High-Performance Reliable Block Encryption Algorithms Secured against Linear and Differential Cryptanalytic Attacks, CEUR Workshop Proceedings, Vol. 2104, pp. 657-668, 2018.
22. Tynymbayev S., Gnatyuk S.A., Aitkhozhayeva Y.Z., Berdibayev R.S., Namazbayev T.A. Modular reduction based on the divider by blocking negative remainders, News of the National Academy of Sciences of the Republic of Kazakhstan, Series of Geology and Technical Sciences, №2 (434), pp. 238-248, 2019.
23. Korobiichuk I., Syerov Y., Fedushko S. (2020) The Method of Semantic Structuring of Virtual Community Content. Mechatronics 2019: Recent Advances Towards Industry 4.0. MECHATRONICS 2019. Advances in Intelligent Systems and Computing, vol 1044. Springer. pp 11-18. https://doi.org/10.1007/978-3-030-29993-4_2
24. Kalimoldayev M., Tynymbayev S., Gnatyuk S., Ibraimov M., Magzom M. The device for multiplying polynomials modulo an irreducible polynomial, News of the National Academy of Sciences of the Republic of Kazakhstan, Series of Geology and Technical Sciences, №2 (434), pp. 199-205, 2019.
25. Gnatyuk S., Kinzeryavyy V., Kyrychenko K., Yubuzova Kh., Aleksander M., Odarchenko R. Secure Hash Function Constructing for Future Communication Systems and Networks, Advances in Intelligent Systems and Computing, Vol. 902, pp. 561-569, 2020.
26. Iavich M., Gagnidze A., Iashvili G., Gnatyuk S., Vialkova V. Lattice based Merkle, CEUR Workshop Proceedings, Vol. 2470, pp. 13-16, 2019.