Satellite monitoring and tracking systems, thanks to the global Internet network, have been applied not only in transport and logistics areas, but also in everyday life. IT development has opened the door to the use of satellite applications, as an integral part of any intellectual system constantly expanding and providing new opportunities.

Nowadays, the volume of transmitted information in global networks is constantly increasing as rapidly as the number of unauthorized persons wishing to obtain this information is increasing. With the development of computer technology, the level of data protection is reduced, and, as a result, it appears to be a relevant task to create new algorithms with high cryptographic strength and high speed in government, military, transport (railways, airports, etc.) and other areas. One possible data protection way based on the one-time pad principle, symmetric cryptographic system with an absolute cryptographic resistance for cracking is presented in the article. Encryption/decryption system based on using involutory matrices in modified Hill encryption algorithm.

Keywords: GPS/GLONASS, matrix encryption, involutory matrix, AES, Triple DES, Internet network

1. Introduction

Satellite tracking and monitoring systems have found their main applications in the transport and logistics sector. It is difficult to imagine a modern logistics and transport world without satellite technologies, which greatly strengthened position as its integral part. The use of these systems allows tracking the location and status of the interested object in real-time. Satellite navigation system GPS/GLONASS used to determine the speed, direction and location of the object. All subsequent information of the object obtained from various kinds of sensors, including satellite data, transmitted via telecommunications and computer networks to the data center.

To transfer data from the object system needs access to the global Internet network, which is provided by a mobile GSM network. If GSM network is not available, then the system stores data in the local drive until GSM access network appears, after which the stored data is transmitted to the network. To transmit data to the Internet mobile GSM providers are using interconnections with internet service providers (ISP). Data transferred from the object passes through a large number of network equipment before they get to the destination (server), which is the main gap in data protection.
2. Modern encryption algorithms analysis used in mobile and computer networks

Nowadays, the problem of using cryptographic methods in information and intelligent transport systems has become particularly relevant, because of the routine introduction and extensive use of global computer networks, which carry large amounts of information that does not allow access to them by unauthorized persons. The rapid development of computing and neural technologies constantly reduces the time for ciphers cracking, which until now had the high cryptographic strength (Kang et al., 2013), (Kodera et al., 2013).

2.1. Encryption algorithms used in GSM networks

There are used A5 and GEA family encryption algorithms in second and third generations of mobile network. A5/3 and GEA3 are most resistant for cracking algorithms, which are based on the use of KASUMI block encryption algorithm. Data encryption in 4G mobile network is based on an AES algorithm.

The first practical attack on the KASUMI cipher, allowing to break the cipher was presented in 2010 year (Dunkelman et al., 2010). However, cryptanalysts did not stop to continue developing new methods for breaking KASUMI cryptographic algorithm (Wentan and Shaozhen, 2014).

Figure 2. Data transmission in GSM network

A5/3 and GEA3 algorithms encrypt data only in wireless segment, from the transmitter mounted on the object to GSM base stations. GSM base station equipment must support these encryption algorithms, and they must be activated. In the mobile provider network data is transmitted in clear format, which can be intercepted, corrupted or tampered by the third persons.

2.2. Encryption algorithms used in data transmission networks

Internet network is interconnection between an internet service provider networks which connect large set of servers, hosts, computers, systems, data centers and so on. Data transmission on the Internet network are based on routing so all objects need to know the real IP address of the destination (server). When transmitting data from one router to another router, the data are not encrypted, as it is done in the GSM wireless network segment, but transmitted in the form in which they were originally formed (figure 3). However, some systems support data encryption before their transmission to the destination, using advanced encryption algorithms like AES or Triple DES.
Figure 3. Encrypted data transmission

Data Encryption Standard (DES) - a symmetric block encryption algorithm based on the Feistel network structure, which encrypts the plaintext blocks of 64 bits with 56 bit key. Due to lack calculating power of computer, DES algorithm for a long time had been invincible and the only way to attack is the exhaustive search of key combinations (2^{56} combinations). With the rapid development of computer technology exhaustive search of the keys becomes a reality at the same time repeatedly minimizing spent time. So in 1993, Michael Wiener was developed specialized computer able to carry out search all keys for 3.5 hours (Weiner, 1996).

The main drawback of the DES algorithm is a short key, that DES has pushed developers to create new versions of an algorithm. The most successful modification of the algorithm was Triple DES. In the Triple DES algorithm block plaintext being three times coding with DES algorithm, in each case using a separate key, that increase key lengths up to 168 bits and increase resistance of cipher. However, high resistance of Triple DES algorithm pays a low rate, which is 3 times less than the speed of DES encryption. Triple DES algorithm gradually replaced by AES, which is six times faster.

In 1997, the National Institute of Standards and Technology announced public competition of symmetric block encryption algorithms, which should replace the DES algorithm as a new standard. AES (Advanced Encryption Standard) competition won Rijndael algorithm. Rijndael algorithm using variable block size and variable key length, which is not dependent on each other and can take the values of 128, 192 and 256 bits. In encryption is used 4 internal functions (these functions are used repeatedly - at least 10 times), each of which is reversible, therefore, for cipher decryption is used inverse functions in reverse order.

The algorithm was developed for high speed at the expense of decreasing its cryptographic strength, thus won the AES competition. However, after the adoption of Rijndael algorithm as the standard AES, cryptanalysts found many examples to break the algorithm (Kang et al., 2013), (Das and Bhauvik, 2014). To improve cryptographic strength of the algorithm its developers recommend to use 18 - 24 rounds, that lowers the algorithm speed.

The actual task of constructing a completely resistant and high-speed encryption algorithm, one of the variant for implementing such kind of algorithm is presented in this article.

3. Basic moments used to develop new high-speed and cryptographically strong algorithm

To implement the new algorithm is used a heavily modified basic idea of Hill cipher (Eisenberg, 1998), and also used a great progress in matrix arithmetic, which is the main part of a cryptosystem.

3.1. Hill cipher

In 1929, Lester Hill created an encryption algorithm, which is based on algorithm which replaces the sequence of plain text with encrypted sequence of the same length (Eisenberg, 1998).

Hill Cipher is using orthogonal matrix for data encryption/decryption, for example: let’s encrypt the word «cipher». Length of the English alphabet is 26 letters, then assign a serial number for each letter of the plaintext, hence, c=2, i=8, p=15, h=7, e=4 and r=17. Let’s take for encryption a square matrix (key)

\[
A = \begin{bmatrix}
3 & 5 & 8 \\
2 & 3 & 4 \\
1 & 1 & 1 
\end{bmatrix}; B = A^{-1} = \begin{bmatrix}
1 & -3 & 4 \\
-2 & 5 & -4 \\
1 & -2 & 1 
\end{bmatrix},
\]

(1)

where \( A \) - encryption matrix (key), and \( B \) - decryption matrix (key). Cipher text:
Decryption:

\[
\begin{align*}
A \cdot C &= \text{kazvqb} \\
&= \text{kazvqb}
\end{align*}
\]
Hill cipher didn’t find practical applications in cryptography, because of a number of significant drawbacks:

- Weak resistance for cracking, due to the use of small matrices;
- Lack algorithm description for direct and inverse matrices generation of a large size.

3.2. **Involutory matrix**

In matrix theory (Franklin, 1993), (Zhang, 1999) sometimes mentioned about the existence of an involutory matrices, for which the initial and its inverse matrix are the same, then

$$\text{cipher}$$

$$= \text{cipher}$$

(3)
The use of such matrices for encryption/decryption contributes increasing of a high speed system, as there is no need to calculate inverse matrix, and the initial matrix is a key for encryption and for decryption. Unfortunately algorithms for generating and descriptions of any involutory matrices properties didn’t mentioned in the scientific literature. However to find such matrices of the small sizes rather easily, for example:

\[
B = \begin{bmatrix}
-3 & -2 & -2 \\
2 & 1 & 2 \\
2 & 2 & 1 \\
\end{bmatrix} ; \quad B^{-1} = \begin{bmatrix}
-3 & -2 & -2 \\
2 & 1 & 2 \\
2 & 2 & 1 \\
\end{bmatrix} ; \quad \det(B) = 1; \quad B = B^{-1}; \quad \Rightarrow B^2 = E
\]

\[
C = \begin{bmatrix}
1 & 1 & 1 \\
1 & 2 & 0 \\
-2 & -2 & -1 \\
\end{bmatrix} ; \quad C^{-1} = \begin{bmatrix}
1 & 1 & 1 \\
1 & 2 & 0 \\
-2 & -2 & -1 \\
\end{bmatrix} ; \quad \det(C) = 1; \quad C = C^{-1}; \quad \Rightarrow C^2 = E
\]

Obviously, the design of optimal high-speed cryptosystem requires the use of involutory matrices with numbers $\pm 2^n$, where $n$ – smallest possible integer. Involutory matrices of size 2x2 and 3x3 with numbers $\pm 2^n$ (expression (5)), where $n$ – integer, do not exist, but matrices of size 4x4 with numbers $\pm 2^n$ are possible to construct. For example:

\[
A = \begin{bmatrix}
2 & -1 & -2 & 2 \\
-1 & -2 & 2 & 2 \\
1 & 1 & 1 & 2 \\
-1 & 1 & 2 & -1 \\
\end{bmatrix} ; \quad \det(A) = 1; \quad A = A^{-1}
\]

Operation of encryption/decryption using involutory matrix as a key, will looks like:
Text: ABB; \Rightarrow Text = [0 \ 1 \ 1 \ 0];

Cipher text:

\[
\begin{bmatrix}
1 & 1 & 1 & 1 \\
-1 & -1 & -1 & -1
\end{bmatrix}
\]

\[\mod(26) = \begin{bmatrix}
X \\
C
\end{bmatrix} = XWCD
\]

(7)
There is used the same matrix for encryption and decryption in expression (7), therefore, the processor at the receiving end doesn’t need to spend time for inverse (decryption) matrix calculation, thus repeatedly increasing the speed of a cryptosystem. At the same time processor for data encryption/decryption will used only shift and addition operations by using a matrix with numbers \( \pm 2^n \), where \( n \) – minimum integer, that also increase the speed of the cryptosystem.

3.3. **Algorithms for forming an expanded set of involutory matrices**

It is obvious that the number of involutory matrices is strictly limited, so we’ll try to find their quantity based on the found properties of these matrices:

- Formation of the involutory matrices using permutation of its rows and columns
  If in any involutory matrix we substituted with places any two rows and appropriate two columns, then we obtain a new involutory matrix.
- Formation of a involutory matrix rotated it for 180 degrees
  If any involutory matrix rotated for 180 degrees, then we obtain a new involutory matrix.
- Formation of a new involutory matrix using transpose
  If any involutory matrix transpose, then we obtain a new involutory matrix.
- Formation of a new involutory matrix using transpose and then rotated it for 180 degrees
  We can use these two operations in another order and we still get a new involutory matrix.
- Formation of a new involutory matrix by changing sign of elements in initial involutory matrix rows and columns
  If in any involutory matrix we replaced the elements sign in any row and in the appropriate column, then we obtain a new involutory matrix.

From any involutory matrix we replaced the elements sign in any row and in the appropriate column, then we obtain a new involutory matrix.

From one involutory matrix we replaced the elements sign in any row and in the appropriate column, then we obtain a new involutory matrix.

From any involutory matrix we can obtained a large set of involutory matrices, using the above described properties. If for received new matrices we again used the properties described in this section, then we obtain a large number of involutory matrices, many of which will be repeated, so in the last steps we’ll need to sort the results.

From any involutory matrix \( 4 \times 4 \), that contains numbers \( \pm 2^n \), where \( n \) – minimum possible number, we can obtain a full set of involutory matrices, total number of which is \( 4 \times 2^4 = 1152 \).

3.4. **Formation algorithms for expanded set of mutually inverse pairs of matrices**

When we create a generation algorithm for mutually inverse pairs of matrices, we should use a fast and simple operation, not requiring complex calculations. As basic matrices we use the involutory matrices, and then the generation algorithms for mutually inverse pairs of matrices are the following:

- Formation a mutually inverse pairs of matrices by permutation of rows and columns in the initial involutory matrix

If in the initial involutory matrix expression (8), in the first step permute any two rows, and in the second step permute appropriate two columns, then we obtain a mutually inverse pairs of matrices. This condition is true for all eight combinations of permutations, hence, this algorithm from one involutory matrix can form eight different mutually inverse pairs of matrices.

\[
A = \begin{bmatrix}
2 & -1 & -2 & 2 \\
-1 & -2 & -2 & 2 \\
1 & 1 & 1 & 2 \\
-1 & 1 & 2 & -1
\end{bmatrix}; \quad A = A^{-1} \Rightarrow A_1 = A_{[\text{row}]}^{\text{2+}} = \begin{bmatrix}
-1 & -2 & -2 & -2 \\
2 & -1 & -2 & 2 \\
1 & 1 & 1 & 2 \\
-1 & 1 & 2 & -1
\end{bmatrix}
\]

\[
A_1^{-1} = A_{[\text{column}]}^{\text{2+}} = \begin{bmatrix}
-1 & 2 & -2 & 2 \\
-2 & -1 & -2 & -2 \\
1 & 1 & 1 & 2 \\
1 & -1 & 2 & -1
\end{bmatrix}; \quad A_1 * A_1^{-1} = E
\]
• Formation a mutually inverse pairs of matrices by rotating initial involutory matrix respectively for 90 and -90 degrees.
  If in any initial involutory matrix in the first step we rotate it for 90 degrees, in the second step, rotate it for -90 degrees, then we obtain a mutually inverse pairs of matrices.
• Formation a mutually inverse pairs of matrices by changing signs of elements in initial involutory matrix rows and columns.
  If in any involutory matrix in the first step we change the sign of elements in any row, and in the second step, change the sign of elements in the appropriate column, then we obtain a mutually inverse pairs of matrices.

Any involutory matrix forms a unified basis for the formation of a large number of mutually inverse pairs of matrices.

3.5. Recursive creation of involutory and mutually inverse pairs of matrices for a large size

To generate involutory and mutually inverse pairs of matrices we’ll use the tensor product.

\[
A = \begin{bmatrix}
  2 & -1 & -2 & 2 \\
-1 & 2 & -2 & -2 \\
 1 & 1 & 1 & 2 \\
-1 & 1 & 2 & -1 \\
\end{bmatrix}; \quad A^2 = E; \quad B = \begin{bmatrix}
  1 & 1 & 1 & 2 \\
-2 & -2 & -1 & -2 \\
-2 & -1 & 2 & 2 \\
 2 & 1 & -1 & -1 \\
\end{bmatrix}; \quad B^2 = E
\]

\[
H_1 = kron(A, B); \quad H_1^{-1} = kron(A^{-1}, B^{-1}); \quad H_1 = H_1^{-1} \\
H_2 = kron(B, A); \quad H_2^{-1} = kron(B^{-1}, A^{-1}); \quad H_2 = H_2^{-1} \\
H_3 = kron(A, A); \quad H_3^{-1} = kron(A^{-1}, A^{-1}); \quad H_3 = H_3^{-1} \\
H_4 = kron(B, B); \quad H_4^{-1} = kron(B^{-1}, B^{-1}); \quad H_4 = H_4^{-1} \\
H_1 \neq H_2; H_1 \neq H_3; H_1 \neq H_4; H_2 \neq H_3; H_2 \neq H_4; H_3 \neq H_4;
\]

(9)

Let’s calculate an involutory matrix for the first case of expression (9):

\[
H_1 = kron(A, B) = \begin{bmatrix}
2 & 2 & 1 & 2 & 4 & 4 & 2 & 4 & 4 & 2 & 4 & 4 & 4 & 2 & 4 & 2 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\
2 & 1 & -1 & -1 & 2 & 1 & -1 & -1 & 2 & 1 & -1 & -1 & 4 & 2 & -2 & -2 \\
-1 & -1 & -1 & 2 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 4 & 1 & -1 & -1 & -2 \\
-2 & -1 & 1 & 1 & 2 & 1 & -1 & -1 & 4 & 2 & -2 & -2 & -2 & -1 & 1 & 1 \\
\end{bmatrix}
\]

(10)

Using the involutory matrix 4x4 we can obtain 1327104 combinations of involutory matrix 16x16. All the above described properties of involutory matrices are valid for any involutory matrix of any size. Combining volutory matrices of different sizes we can obtain a large set of matrices. But for obtaining a full set of matrices 2^n x 2^n, where n = 2, 3, ..., ∞, it’s necessary to
obtain a involutory matrix 8x8. The author of this article developed a completely new algorithm for generating involutory matrices of size 8x8. For example:

\[
A = \begin{bmatrix}
-2 & -2 & 1 & -1 & 2 & -2 & -1 & -2 \\
-2 & -2 & 1 & -1 & 2 & -2 & -1 & -2 \\
-2 & -2 & 1 & -1 & 2 & -2 & -1 & -2 \\
-2 & -2 & 1 & -1 & 2 & -2 & -1 & -2 \\
2 & 2 & -2 & 2 & -2 & 2 & -2 & -1 \\
-2 & -2 & 2 & -1 & 2 & -2 & -2 & -2 \\
-2 & -2 & 1 & -2 & -2 & -2 & -1 \\
-1 & -2 & 2 & -2 & 1 & -2 & -2 & -1 \\
2 & 2 & -1 & 2 & -2 & 2 & 2 & 2
\end{bmatrix}
\]

\(A = A^{-1}\) (11)

Mutually inverse pairs of matrices are generated in the same way, using different combinations. Tensor product allows us to generate a large set of matrices, but due to its structure, the resulting matrices will always contain elements \(\pm 2^n\). For generation matrices 16x16 we use matrices 4x4 with numbers \(\pm 1\) and \(\pm 2\), so resulting matrix will always contain numbers in one order higher, namely \(\pm 1, \pm 2\) and \(\pm 4\). The author of this article developed a completely new algorithm, that allows to generate a involutory matrix with elements \(\pm 1\) and \(\pm 2\) of any size \(2^n\times2^n\), where \(n = 2, 3, \ldots, \infty\). For example involutory matrix 16x16:

\[
A = \begin{bmatrix}
-2 & -2 & 1 & 2 & 2 & 2 & 1 & 1 & 1 & 2 & -2 & -1 & -1 & -2 & 1 \\
-2 & -2 & 1 & 2 & 2 & 2 & 1 & 1 & 1 & 2 & 2 & -2 & -1 & -2 & 1 \\
-2 & -2 & 1 & 2 & 2 & 2 & 1 & 1 & 1 & 2 & 2 & -1 & -1 & -2 & 2 & 2 \\
1 & 1 & -2 & 1 & 2 & 2 & 2 & 2 & 1 & 2 & 2 & 1 & -2 & -2 & -1 & 2 \\
\end{bmatrix}
\]

\(A = A^{-1}\) (12)

4. Development of high-speed and cryptographically strong encryption system based on matrix transformations

Below are some examples of cryptosystems that can be used in different situations, depending on required cryptographic strength.

4.1. Cryptosystem with constant key length

To encrypt data we proposed to use the key with a constant length, in each encryption block the key is changed to a new one. To improve system’s resistance, we offer to use the multiplication of several matrices (keys). The general scheme of system encryption/decryption
is shown in figure 4, where m - a length of plaintext block, and n – selected number of matrices.

\[ m \quad \text{Plaintext} \quad m \quad \text{Plaintext} \quad m \quad \text{Plain text} \]

**Encryption key**

\[
\begin{array}{ccc}
A_1 * A_2 * \ldots * A_n & B_1 * B_2 * \ldots * B_n & \ldots & C_1 * C_2 * \ldots * C_n \\
\end{array}
\]

**Cipher text**

\[
\begin{array}{ccc}
\text{Cipher text} & \text{Cipher text} & \ldots & \text{Cipher text} \\
\end{array}
\]

**Decryption key**

\[
\begin{array}{ccc}
A_1^{-1} * A_2^{-1} * A_1^{-1} & B_1^{-1} * B_2^{-1} * B_1^{-1} & \ldots & C_1^{-1} * C_2^{-1} * C_1^{-1} \\
\end{array}
\]

**Plaintext**

\[
\begin{array}{ccc}
\text{Plaintext} & \text{Plaintext} \quad \ldots \quad \text{Plaintext} \\
\end{array}
\]

*Figure 4. Cryptosystem structure with constant key length*

### 4.2. Cryptosystem with variable key length

The structure of this algorithm is the same as in section 4.1, the main difference lies in the fact that the length of plaintext is changed randomly, and therefore encryption/decryption matrix are changed in the same way (Figure 5). Length of the plaintext block is equal \(2^n\), where \(n\) – random integer, which varies in the range from 2 to a number, which is set by the user.

\[ m_1 \quad \text{Plaintext} \quad m_2 \quad \text{Plaintext} \quad m_3 \quad \text{Plaintext} \quad m_4 \quad \text{Plaintext} \]

**Encryption key**

\[
\begin{array}{ccc}
A_1 * A_2 * \ldots * A_n & B_1 * B_2 * \ldots * B_n & \ldots & C_1 * C_2 * \ldots * C_n \\
\end{array}
\]

**Cipher text**

\[
\begin{array}{ccc}
\text{Cipher text} & \text{Cipher text} & \ldots & \text{Cipher text} \\
\end{array}
\]

**Decryption key**

\[
\begin{array}{ccc}
A_1^{-1} * A_2^{-1} * A_1^{-1} & B_1^{-1} * B_2^{-1} * B_1^{-1} & \ldots & C_1^{-1} * C_2^{-1} * C_1^{-1} \\
\end{array}
\]

**Plaintext**

\[
\begin{array}{ccc}
\text{Plaintext} & \text{Plaintext} \quad \ldots \quad \text{Plaintext} \\
\end{array}
\]

*Figure 5. Cryptosystem structure with variable key length*

This system uses a transition from one matrix size to another depending on the length of plaintext block. Resistance of the algorithm is increased many times, as there is no information what length of plaintext is used in first, second stages and so on.

### 4.3. Cryptosystem with a double transformation

First block of plaintext is encrypted using the small size of matrices, and then obtained result is encrypted using the larger size of matrices (Figure 6).
Due to the large number of transformations, this algorithm will lose speed compared with previously described algorithms, but it resistance will exceed. This algorithm can be implemented with a variable key, which was described in section 4.2.

5. Conclusion

Satellite tracking and monitoring systems like mobile technology have become massively popular and available at any point of the globe thanks to the global Internet network, but their safety is doubtful. Time for popular strong encryption algorithm crack is constantly decreasing due to the rapid development of neural and nano technologies that challenge cryptography to create new stronger encryption algorithms. One embodiment of such system based on modified Hill encryption algorithm is proposed in the article. Modified Hill system is based on a new algorithm for generating initial (encryption) and inverse (decryption) matrices of a large size, in which all of the numbers are ±2\(^n\), where n - the minimum possible number. As all matrices contain the numbers of computer arithmetic’s ±2\(^n\), then encryption/decryption operations will be performed without any multiplication that significantly increases the speed of the algorithm.

The paper also suggests an algorithm for constructing involutory matrices with elements ±2\(^n\), for which the initial and its inverse matrix are the same. Using these matrices enables to realize high-speed system, because the key for encryption and decryption is the same. Shown, that any involutory matrix offers with significant advantages, allowing to obtain from any involutory matrix a complete set of involutory and mutually inverse pairs of matrices. In article is shown a recursive method to forming from
any involutory matrix a large set of involutory and mutually inverse pairs of matrices gradually increasing size. Because of the almost inexhaustible set of matrices obtained in this article, the proposed cryptosystems allow for encryption/decryption using more than one pair of keys, but a few that greatly increases system resistance. Each block of plaintext is processed by a separate key (matrices), which is equivalent to the principle of “one-time pad” in a completely resistant system for cryptanalysis.

To break Triple DES (Kelsey et al., 2000) we’ll need to use exhaustive search for $2^{72}$ combinations, but to break 10 round AES 128 (Bogdanov et al., 2011) we’ll need to use exhaustive search for $2^{126.1}$ combinations, then ceteris paribus to break a new algorithm, which consists of 10 rounds (using 10 matrix 16x16) we’ll need to use exhaustive search for $2^{264}$ combinations.

These algorithms can be used not only in the basic cryptosystem, but also as additional algorithms for increasing the resistance for the existing block system, for example, to improve resistance for SAFER+ algorithm.

All considered models of cryptosystems have following properties:

- Diffusion - influence spread of one plaintext character into a large number of cipher text characters that hides statistical properties of the plaintext. Development of this principle is influence spread of one key character into a large number of cipher text characters that prevents key recovery in parts;
- Mixing - use of encryption transformations that complicate the restoration relationship for statistical properties of plaintext and cipher text;
- Random keys;
- Equality of key length and the length of the plaintext;
- Using the key only once, without its repeating.

References