DOI: https://doi.org/10.36910/6775-2524-0560-2023-50-09
UDC 004.05(075.8)

Марценюк Василь Петрович, д.т.н., професор,
https://orcid.org/0000-0001-5622-1038
Університет Бєльсько-Бяли, Польща[1)]
Сверстюк Андрій Степанович, д.т.н., професор,
https://orcid.org/0000-0001-8644-0776
Тернопільський національнийний медичний університет імені І.Я. Горбачевського, Україна[2)]
Андрущак Ігор Євгенович, д.т.н., професор,
https://orcid.org/0000-0002-8751-4420
Речун Оксана Юріївна, к.е.н., доцент,
https://orcid.org/0000-0001-7932-4769
Луцький національний технічний університет[3)]

COMPONENTS AND KEY FEATURES OF THE ANALYSIS SYMETRIC CRYPTOCIRCUIT

Martsenyuk V.P., Sverstyuk A.S., Andrushchak I.Ye., Rechun O.Yu. Components and kay features of the analisis symmetric cryptocircuit. This article discusses the key aspects and technologies of applying many cryptanalysis methods for assessing the stability of symmetric block encryption algorithms. The following aspects of analysis methods such as linear, differential, slide attack, algebraic analyzes are also considered. Approaches to the analysis of the AES standard are compared. Also, when considering the approaches to the analysis of modern symmetric cryptosystems, special attention is paid to special questions regarding the possibility of using distributed multiprocessor calculations in order to reduce the analysis time.

Keywords: secret key, cryptoanalysis, symmetric encryption, block cipher, stability, distributed multiprocessor computing.

Марценюк В.П.[1)], Сверстюк А.С.[2)], Андрущак І.Є., Речун О.Ю.[3)] Компоненти і ключові особливості аналізу симетричних криптосхем. У статті розглядаються ключові аспекти та технології застосування багатьох методів криптоаналізу для оцінки стійкості симетричних блочних алгоритмів шифрування. Також розглядаються такі аспекти методів аналізу, як лінійний, диференціальний, слайд-атака, алгебраїчний аналіз. Порівняно підходи до аналізу стандарту AES. Також при розгляді підходів до аналізу сучасних симетричних криптосистем особливу увагу приділено спеціальним питанням щодо можливості використання розподілених багатопроцесорних обчислень з метою скорочення часу аналізу на різних видах вірусів та проводиться аналіз методів їх класифікації, виявлення та знищення.

Ключові слова: секретний ключ, криптоаналіз, симетричне шифрування, блоковий шифр, стабільність, розподілені багатопроцесорні обчислення.

Formulation of the problem. Modern encryption algorithms are developed in such a way that the analyst has as few chances as possible to find the secret key that was used to encrypt the data, even if he knows the encryption algorithm itself and has several texts and their corresponding ciphertexts available. When starting the task of analysis, the analyst first of all defines the set of data known to him for analysis. The type of cryptanalysis that the analyst can use depends on this. Let's consider the main types of cryptanalysis of modern symmetric cryptosystems.

Analysis of research. If the encryption algorithm is known and there is at least one plaintext-ciphertext pair, then the most natural way of analysis that immediately comes to mind is to sequentially test all possible variants of the key that could have been used. Testing is carried out until the encryption of the plaintext with the next key leads to the receipt of the existing encrypted message. This method of analysis in different sources of literature has different names, for example, "Method of complete search" [1] or "Method of brute force" [2] or "Method of head-on attack" [3] or "Brut-force attack" [2]. This method has one indisputable advantage: sooner or later the key you are looking for will be found and a minimum set of data will be required for this. The speed of finding the key will depend on the length of the secret key used and on the computing power available to the analyst. And also from a share of luck. After all, it may happen that the key you are looking for is one of the first to be found.

At the same time, we know that one of the important properties of information is its timeliness. Therefore, the application of the method of complete enumeration is practically easy to implement, but it is usually not used. For example, when the DES encryption algorithm was developed, the length of its actual secret key was determined to be 56 bits. That is, in order to go through all possible variants of secret keys, it was necessary to make 256 tests. With the help of computing tools that were available at that time, it could be done in several decades! Of course, since the DES encryption algorithm was developed, there has been a huge leap in computing and computing power has increased thousands of times. Today, with the use of powerful computing clusters, the task of finding a secret key for the DES algorithm can be solved in a few minutes. Due to the fact that computing power is inexorably increasing every day, the DES standard was replaced by the new AES (Advanced Encryption Standard), where the length of the secret key increased to 128 bits. One way or another, in cryptography it is accepted to consider the time of analysis using the full enumeration method as a reference time. What does it mean? This means that if an analyst manages to analyze an encryption algorithm faster than it can be done using a full scan, then the given encryption algorithm will be considered vulnerable, and therefore it will not be appropriate to use it for data encryption. The task of finding the secret key of encryption by the method of complete search is well parallelized and can be easily implemented by multiprocessor computing systems.

Presentation of the main material and the justification of the results. The meet-in-the-middle method is applicable to encryption algorithms that use two different keys K. This can be achieved if the secret subkeys appear with some periodicity, or, for example, if the data has been double-encrypted, the data was first encrypted on one key K_1, and then the obtained encryption result was encrypted again on another secret key K_2.

Let us know the plaintext-plaintext pair encrypted in this way. In this case, it is necessary to encrypt the plaintext on all possible values of the K_1 key. In parallel with this, it is necessary to decrypt the closed text of all possible values of the K_2 key. The pair of keys (K_1, K_2) for which the result of encryption of the open text and the result of decryption of the closed text match, will be searched. As is clear from the explanations, the analysis taking into account the "meeting in the middle" method can be parallelized and implemented using distributed multiprocessor computing. As an example of how the method works, you can consider options for analyzing the double DES algorithm or, for example, analyzing an algorithm in which the same key is actually used four times.

The method of linear cryptanalysis was first proposed by the Japanese scientist M. Matsui. In his work [4], M. Matsui showed how an attack on the DES encryption algorithm can be carried out, reducing the complexity of the analysis to 2^{47}. A significant drawback of the method was the need to have a large volume of data encrypted with the same secret key, which made the method unsuitable for practical use to crack the cipher. However, if we assume that the analyst got into the hands of an encrypted text containing important information, as well as a certain black box (device or program) that allows you to execute any number of texts encrypted using a known encryption algorithm on a secret key, no while revealing the key itself, the application of the method of linear cryptanalysis becomes quite real. Many encryption algorithms, known at the time of the publication of Matsu's work [4,5], were subsequently tested for resistance to this method and not all of them turned out to be sufficiently stable and, as a result, required refinement (fig. 1).

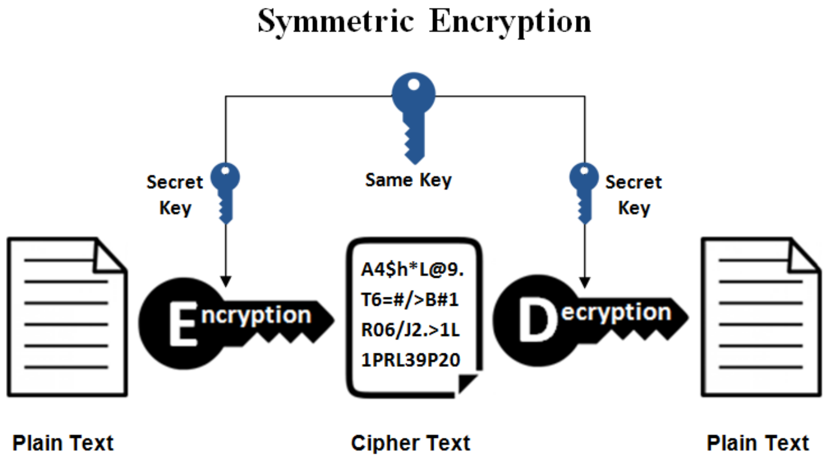

Figure 1. Symmetric encryption

Knowledge of the working mechanisms of the linear cryptanalysis method allows cryptographers to ensure the stability of ciphers even at the stage of designing crypto-algorithms. That is why it is so important to be able to apply known methods of cryptanalysis in practice. So, let's consider the main concepts related to the method of linear cryptanalysis. Any encryption algorithm in its most general form can be represented as some function E depending on an input message X, a secret key K and an encrypted message Y that returns:

$$Y = E(X, K). \quad (1)$$

Knowing exactly the transformation E and the input message X, it is not possible to say unequivocally what the output message Y will be. In this case, the nonlinearity of the function (1) depends on the internal mechanisms of the transformation E and the secret key. K. M. Matsui showed that it is possible to imagine the encryption function (1) in the form of a system of equations that are performed with a certain probability p. At the same time, for a successful analysis, the probability of the equations p should be as far as possible from the value of 0,5 (that is, approach either 0 or unity). Since the equations obtained during the analysis of the crypto-algorithm are probabilistic, they have come to be called linear statistical analogues. The linear statistical analogue of the non-linear encryption function (1) is called the quantity Q, which is equal to the sum modulo two scalar products of the input vector X, the output vector Y and the secret key vector, corresponding to the binary vectors α, β и γ, have at least one coordinate equal to one:

$$Q = (X, \alpha) \oplus (Y, \beta) \oplus (K, \gamma),$$

if the probability that Q = 0 is different from 0,5 (P(Q = 0) = 0.5). Unlike differential cryptanalysis, in which a large probability value guarantees the success of the attack, in linear cryptanalysis, the success of the analysis can be ensured both by equations with a very high probability and equations with a very small probability. In order to understand which of the possible equations is the best to use for analysis, the concept of deviation is used.

The deviation of a linear statistical analog is called a quantity $\eta = |1–2p|$, where p - is the probability from which the linear analog is performed. The deviation determines the effectiveness of the linear statistical analogue. The greater the deviation, the higher the probability of a successful analysis. In fact, the deviation shows how far the probability of the statistical counterpart is from the value p = 0,5.

To successfully apply the linear cryptanalysis method, the following tasks must be solved:

- find the most effective (or close to them) statistical ones linear analogs. When finding analogues, pay attention to the fact that as much of the battle secret key K as possible should be involved in them.

- get statistical data: the required amount of pairs of texts (open - closed text) encrypted using the analyzed algorithm on the same secret key.

- determine the keys (or some bits of the key) by analyzing statistical data using linear analogs.

The first step of the analysis is to find effective statistical counterparts. For encryption algorithms in which all blocks are known in advance, this step can be performed once, based on an analysis of the linear properties of all cryptographic elements of the cipher. As a result of the analysis, a balanced system should be obtained, which are performed with certain probabilities. The left part of the equations should contain the sum of the bits of the input and output messages, the right part of the equations should contain the bits of the secret key. The system of equations must be defined to contain all bits of the original secret key. This stage is not particularly difficult, but requires more knowledge, work logic and attentiveness. It can be automated. However, it must be remembered that for each specific encryption algorithm, the system of linear analogs is built only once and can be used later to find various secret encryption keys that are used to encrypt data using the analyzed cipher. If the first step of the analysis is purely theoretical and completely depends on the structure of the algorithm, then the second step is an exclusively practical part, which consists in the analysis of known pairs of open-closed text using the previously obtained system of statistical analogues. And so the following algorithm is used [6].

Algorithm. Let N be the number of all open texts and T be the number of open texts for which the left part of the linear statistical analogue is equal to 0. Let us consider two cases.

1. If T > N/2, then in this case the number of open texts for which the left part of the analogue is zero is more than half, i.e. in most cases a value equal to zero appears in the left part of the analogue, then:

a) if the probability of this linear statistical analogue p >1/2, this means that in most cases the right and left parts of the analog are equal, which means that the left part of the analog containing the bits of the key is equal to 0.

b) if the probability of this linear statistical analogue p < 1/2, this means that in most cases the right and left parts of the analog are not equal, which means that the left part of the analog containing the bits of the key is equal to 1.

2. If T< N/2, then in this case the number of open texts for which the left part of the analogue is zero is less than half, in most cases a value equal to one appears in the left part of the analogue: a) if the probability of this linear statistical analogue p >1/2, this means that in most cases the right and left parts of

the analog are equal, which means that the left part of the analog containing the bits of the key is equal to 1.

b) if the probability of this linear statistical analogue $p < 1/2$, this means that in most cases the right and left parts of the analog are not equal, which means that the left part of the analog containing the bits of the key is equal to 0.

This algorithm will be successful when analyzing a large number of texts N. Therefore, the second step of the analysis is computationally complex. Therefore, parallel computing can and should be used to speed up the analysis time.

As a result of the operation of the above algorithm, a certain (and possibly redefined) system of equations will be obtained, which reflects the relationship of the bits of the key. The third step of the analysis consists in solving this system, for example, by the Gaussian method, which will allow obtaining the values of the bits of the secret encryption key. You can read more about the linear cryptanalysis of various block encryption algorithms in [1]. The differential cryptanalysis (DC) method was first proposed by E. Biham and A. Shamir to analyze the DES encryption algorithm. Although B. Schneier's book [3] mentions that the developers of the DES algorithm knew about the possibility of such an analysis even during the development of the algorithm, the general public learned about differential cryptanalysis precisely from the works [5, 6]. The DC method turned out to be the first method that allows you to crack DES when the complexity of tasks is estimated to be less than 2^{55}. According to [5], using this method, you can perform DES cryptanalysis with an effort of the order of 2^{37}, but with the presence of 2^{47} variants of the selected plaintext. Although 2^{47} is obviously much less than 2^{55}, the need to have 2^{47} variants of the selected plaintext turns this version of the cryptanalysis scheme into a purely theoretical exercise [7]. This is due to the fact that the DC method was known at the time of the development of DES, but classified for obvious reasons, which is confirmed by the public statements of the developers themselves [3]. It is shown in [6] that if you change the order of replacement blocks in the DES encryption algorithm or use other sets of substitution and permutation tables, the algorithm immediately becomes much weaker and can be broken in less than half the time required to analyze the DES algorithm using complete search. This shows the importance of knowing the possible ways of analyzing the algorithm being developed. With the help of the DK method, the complexity of the analysis was reduced to 2^{37}. However, to conduct the analysis, it was necessary to have 2^{37} specially selected texts encrypted with the same secret key. Despite the limitations imposed on the use of the new proposed methods of analysis - it was a breakthrough! Further development of this method showed the possibility of its application to a whole class of different types of ciphers, made it possible to identify the weak points of many encryption algorithms in use and being developed. Today, this method, as well as some of its derivatives, such as the linear differential method, the impossible differential method, and the boomerang method, are widely used to evaluate the stability of newly generated ciphers. That is why an information protection specialist needs to have an idea of the mechanisms of cipher analysis using modern cryptanalysis methods.

The very name differential cryptanalysis comes from the English word difference. That is why in domestic literature this type of analysis is sometimes called the difference method. Based on the name, it can be understood that when considering the possibility of analyzing some block encryption algorithm, scientists thought of using not individual texts, but pairs of texts. It is clear that the two texts will have differences in some positions. In order to determine this difference, it is enough to combine two modules of a pair of texts. The result of such addition at the output is the value 0 in those positions in which the source texts were equal to each other, and, accordingly, the value 1 in those positions in which the source texts differed. For example, consider two 4-bit messages: $X = 0011$ and $X' = 1010$. The difference $\Delta X = 1001$ was obtained as a result of composing the texts X and X', the obtained value ΔX is called the differential or difference. In differential cryptanalysis, the value of the difference (differential) is usually denoted by the symbol Δ. The difference obtained as a result of adding texts X and X' shows that in the second and third positions the original messages X and X' were equal, and in the first and fourth they differed from each other [8].

In general, the differential analysis of block encryption algorithms is reduced to the following main points (fig. 2):

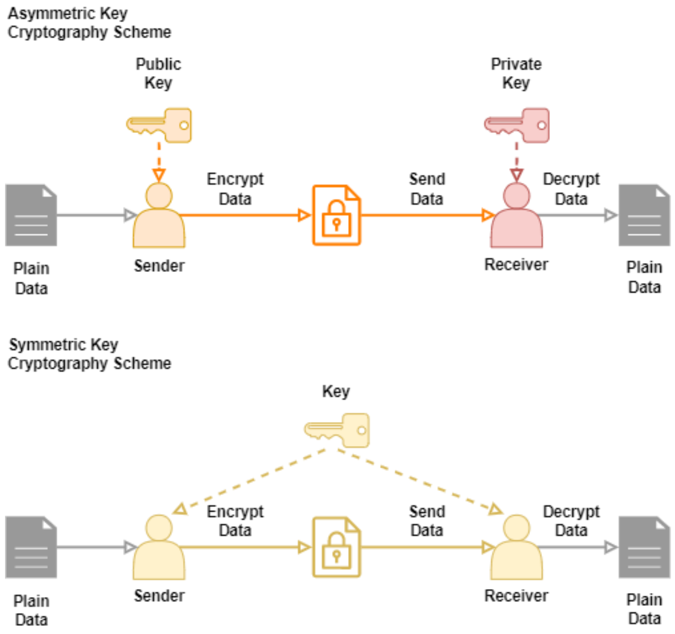

Figure 2. Processes to encrypt and decrypt data in asymmetric and symmetric cryptography algorithms

- finding characteristics for the encryption algorithm with maximum characteristics. The search for characteristics is conducted taking into account the differential properties of nonlinear cryptographic primitives that are part of the encryption algorithm.

- searching for correct pairs of texts using the found characteristics.

- analysis of correct pairs of texts and accumulation of statistics on possible values of the encryption secret key [9].

The first point, which consists in finding the best characteristics for most algorithms, is performed once and is a theoretical task. The values of the characteristics completely depend on the structure of the encryption algorithm and cryptographic primitives. Otherwise, it applies only to those algorithms that have non-fixed elements. Such algorithms include, for example, an encryption algorithm whose S-blocks of replacement can be chosen arbitrarily. For such algorithms, the search for characteristics must be started from the beginning every time, based on the differential properties of the selected S-blocks. To automate the analysis process, it is possible to develop an algorithm for finding the best characteristics, based on tree search algorithms. For such algorithms, parallel models can be used to speed up the search for characteristics.

The second step of the analysis is a computationally robust task for any encryption algorithm, whether it has fixed or non-fixed elements. Analysis consists of testing a large number of text pairs to determine whether they are a valid text pair, that is, a text pair that can later be used for analysis to find a secret encryption key. This step can and should be easily imagined in the form of parallel calculations to reduce analysis time [10].

The last step is easy to implement and requires much less computation than the second step. It can be implemented both separately in the form of a sequential algorithm and be included in parallel algorithms for searching for correct pairs of texts. In the latter case, upon finding the correct pair of texts, you can immediately analyze it to gather statistics on the possible value of the secret key. The essence of algebra analysis methods consists in obtaining equations describing nonlinear transformations of replacing S-blocks, followed by solving the found systems of equations and obtaining an encryption key. This cryptanalysis method refers to attacks with known plaintext, so for successful analysis it is enough to have one plaintext/ciphertext pair. Algebraic methods of cryptanalysis consist of the following stages: drawing up a system of equations describing transformations in nonlinear cryptographic primitives of the analyzed cipher (most often, for symmetric encryption algorithms, such nonlinear components are S-blocks of substitution); solution of the obtained system of equations.

Conclusion and prospects for further research. Both symmetric and asymmetric encryption play an important role in keeping confidential information and communications secure in today's digital world.

Both ciphers can be useful, because each of them has its advantages and disadvantages, so they are used in different cases. As cryptography as a science continues to evolve to protect against new and more serious threats, symmetric and asymmetric cryptographic systems will always be relevant to computer security. These two types of encryption have both advantages and disadvantages relative to each other. Symmetric encryption algorithms are much faster and require less computing power, but their main disadvantage is key distribution. Since the same key is used to encrypt and decrypt information, this key must be shared with anyone who needs access, which creates some risks (as described earlier).

In turn, asymmetric encryption solves the problem of key distribution by using public keys for encryption and private keys for decryption. The trade-off is that asymmetric systems are very slow compared to symmetric systems and require significantly more computing power due to the length of the key.

References

1. Grusho A.A., Timonina E.E., Primenko E.A. Analysis and synthesis of cryptoalgorithms. Lecture course. - Yoshkar-Ola: Publishing house of the MF MOSU, 2000.
2. Stallings V. Cryptography and network protection: principles and practice. - 2nd ed.: Per. from English. – M.: Ed. house "Williams", 2001.
3. Babenko L.K. Ischukova E.A. Modern block cipher algorithms and methods for their analysis. – M.: Helios ARV, 2006.
4. Schneier B. Applied cryptography: Protocols, algorithms, source texts in the C language. - M.: TRIUMPH, 2002. - 648 p.
5. Matsui M., Linear Cryptanalysis Method for DES Cipher, Advances in Cryptology. - EUROCRYPT'93, Springer-Verlag, 1998. - 386 p.
6. Biham E., Shamir A., Differential Cryptanalysis of the Full 16-round DES // Crypto'92, Springer-Velgar, 2008. - P. 487.
7. Biham E., Shamir A. Differential Cryptanalysis of DES-like Cryptosystems. Extended Abstract // Crypto'90, Springer-Velgar, 2008. - P. 2.
8. Panasenko S. Encryption algorithms. Special guide. - St. Petersburg: BHV-Petersburg, 2009. - 576 p.
9. Courtois N., Gregory V. Bard. Algebraic Cryptanalysis of the Data Encryption Standard // 11th IMA Conference, 2007. - P. 152-169.
10. Martseniuk V. Features of multifunctional Backdoor technology. Scientific journal “Computer-Integration Technologies: Education, Science, Engineering” / V. Martseniuk, A. Sverstiuk, I. Andrushchak, O. Sivakovska, M. Poteichuk // Issue No. 40, Lutsk. - 2020 - p. 123-127.