ISSN 2224-087X. Електроніка та інформаційні технології. 2020. Випуск 14. С. 22–35
Electronics and information technologies. 2020. Issue 14. P. 22–35

UDC 004.94:656.1 DOI: https://doi.org/10.30970/eli.14.2

COMPARATIVE ANALYSIS OF TEXT DATA ASYMMETRIC ENCRYPTION METHODS

T. Matveichuk, V. Smychok, S. Filimonov

Department of Electromechanics and Electronics,
Hetman Petro Sahaidachnyi National Army Academy,
32 Heroes of Maidan St., UA-79012, Lviv, Ukraine
tais-28@ukr.net , smychok@ukr.net , sergnf@gmail.com

Actuality. Actuality of work is conditioned by the problem of development of effective methods of defense of information in informative networks which provide the reliable functioning of the automated systems of military purpose, that all computer systems control and connect systems. The complex of experimental researches of directed is a research object, the purpose of which there is comparison of algorithms of the asymmetric enciphering of texts information.

Method. The cycle of experimental tests was carried out after such parameters of algorithms of enciphering: time of generation of the keys; time of enciphering and decoding; carrying capacity of process of enciphering and decoding; size of in cipher and decoded file. The experiments are conducted on the Intel Core 2 Duo CPU processor 2.09 GHz from 4 Gb main memories under the operating system of Windows 7.

Results. The main evaluative results obtained are as follows: the cryptographic strength of the ElGamal algorithm is significantly higher than that of the RSA algorithm; the RSA algorithm has a higher speed when encrypting information, and the ElGamal algorithm shows better results during decryption; with the key sizes increase, the decryption time by the RSA algorithm grows exponentially, while the duration of the decryption by the ElGamal algorithm has a linear growth order; the RSA algorithm showed 2 times better bandwidth than the ElGamal algorithm in the process of encoding information, but the ElGamal algorithm showed 10 times better bandwidth compared to the RSA algorithm in the process of decryption; the length of the encrypted data by the ElGamal algorithm is 2 times longer than the original data, while the size of the data encrypted by the RSA algorithm is larger than the size of the original data by an average of 1.4; for all key lengths, the ElGamal algorithm creates a pair of public and private keys on average 10 times faster than the RSA algorithm, which is especially noticeable with a significant increase in key sizes; for the ElGamal algorithm, the key generation time increases linearly with increasing key sizes, while for the RSA algorithm it grows in geometrical progression.

Conclusions. By the developed software product the comparative analysis of asymmetric algorithms of enciphering of texts information, their advantages and failings, cryptographic firmness, is conducted, given experimental estimation of their descriptions in relation to efficiency of the use by them memory of computer, duration of processes of generation of the keys, enciphering and decoding of information, carrying capacity of algorithms, measure keys, volumes of in ciphers and deciphered files. On the basis of the got results recommendations of application of the considered methods of enciphering are given.

***Keywords*:** programming software, asymmetric algorithms, encryption, decryption, cryptostability, algorithms RSA & ElGamal.

INTRODUCTION

In today's information society, a large number of services are provided through computer networks and information technology, continuous development of which extremely exacerbates issues of information security. The information presented in digital form must be reliably protected against many threats: unauthorized access, forgery, information leak, disclosure of confidential information, and so on. Therefore, the issue of effective information protection methods in information systems is becoming especially relevant today.

The threats to information security are caused, on the one hand, by increasing use of computer networks, which transmit large streams of information, access to which is strictly forbidden to outsiders. On the other hand, the emergence of modern high-powered computers, the development of information technology and neural computing have made it possible to discredit cryptographic systems that were previously considered to be resistant to cryptanalysis. Information security obtained particular importance in the military sphere. Activation of the struggle in cyberspace becomes an integral part of military conflicts.

The object of study is the algorithms of text data encryption.

The subject of study is the means of comparison of algorithms of the asymmetric enciphering of texts information.

The purpose of the work: the objective of the pilot study is to conduct a variety of experiments to compare asymmetric text data encrypting algorithms in context of their memory usage, the duration of key generation processes, encryption and decryption, bandwidth algorithms, file sizes and keys; algorithms for encoding RSA and ElGamal are selected as asymmetric algorithms.

1 PROBLEM STATEMENT

To achieve this goal in the process must be resolved by the assignment: the choice of methods, which will allow for evaluation and comparative analysis of symmetric algorithms depending on the requirements and conditions of use; the choice of the parameters of the algorithm for frying for experimental studies; selection and analysis of algorithms of asymmetric encryption of text data; developing proposals and recommendations for the use of the investigated algorithms.

2 REVIEW OF THE LITERATURE

The severity and relevance of information security in the system of national and world security has caused the interest of domestic and foreign scientists to this problem. The cryptographic methods associated with encryption and decryption algorithms play a major role in any security system. These algorithms spend a considerable amount of time and resources on the system. However, it is not enough to invent an encryption algorithm, it is important to evaluate the effectiveness of such an algorithm with respect to existing ones. The need to study and compare the computational efficiency of encryption algorithms remains an urgent task. This will allow you to find out which algorithms should be used in a particular situation for maximum efficiency. In addition, analysis of research results may be a reason for more detailed study of algorithms to determine whether more efficient algorithms can be obtained by hybridization or concatenation of the studied algorithms.

In the last decades of the last century, cryptographic systems have been developed and widely used, built on the use of asymmetric cryptography methods. The main ones are methods based on the use of RSA and ElGamal cryptographic algorithms. Both of these algorithms work well for both data encryption and digital signatures.

The choice of these two algorithms is not random. The most common asymmetric encryption algorithm is the RSA.RSA algorithm – a classic method. Most modern security systems are based on the RSA algorithm. The RSA algorithm was proposed by three scientists, R. Rivest, A. Shamir, and L. Adleman, in 1977 [1]. In 1993, the RSA method was promulgated and adopted as a standard (PKCS # 1: RSAEncryptionstandart [2]). Although the RSA patent expired on September 21, 2000, RSA is the most popular open-key cryptosystem.

The security of the RSA algorithm is based on the complexity of large numbers decomposition on the multipliers, namely, on the exceptional complexity of the task of determining the secret key based on the public key, since this will require solving the problem of the existence of integer divisors. The most crypto-resistant systems use 1024-bit and larger numbers [3].

As the second asymmetric algorithm, encrypting algorithm by ElGamal scheme was selected. This algorithm has been existing for a long time (it was proposed by Tahir El Gamal in 1984 [4]), it became the first comprehensive public key algorithm that can be used for encryption and digital signatures and that is not patented in the United States and around the world (Patent for the Diffie- Hellman expired in 1997). In addition, it is relatively simple to understand and implement. Algorithm ElGamal is also quite popular. For many years it has been opposed to intensive cryptanalysis.

The security of the ElGamal algorithm is based on the complexity of the calculation of a discrete logarithm in a finite field. If it is sufficiently easy to extend a number to a degree in a finite field, then the restoration of an argument by value (that is, to find a logarithm) is a rather complicated task. Open and closed encryption keys are functions of the two large (1024-2048 bits in binary representation or even more) prime numbers [5].

RSA and ElGamal systems are well described in many scientific sources [1-10].

3 MATERIALS AND METHODS

Comparison of cryptostability algorithms. In terms of practical software and hardware implementation, there is no significant difference between the RSA and ElGamal algorithms, but they differ significantly in cryptostability [11].

If one considers the decomposition of an arbitrary integer with 512 bits length on simple factors for the RSA algorithm and the problem of 512 bits integer logarithm for the ElGamal algorithm, then the second problem is much more complicated than the first one.

However, there is one peculiarity. If in a system constructed using the RSA algorithm, cryptanalyst managed to decompose the public key n of one of the subscribers into two prime numbers, the possibility of abuse is limited to this particular user. In the case of the system built using the ElGamal algorithm, the threats of disclosure are experienced by all subscribers of the cryptographic network [6].

In addition, Lenstra and Manasse not only shook the stability of the RSA algorithm, having the Ninth Fermat number decomposed in 1990 to simple factors in a rather short time [7], but also pointed out a weak point in the ElGamal system, having proved that the approach applied in decomposing of the ninth-value Fermat to simple factors allows to substantially improve the methods of discrete logarithm for some special prime numbers. That is, the one who chooses a simple number p for the algorithm ElGamal has the ability to choose a special simple number for which the problem of discrete logarithm will be simple enough even for ordinary computers, not to mention the modern powerful equipment. To date, there are known decompositions to simple factors of all Fermat numbers up to F_{32} inclusive.

However, this problem is not fatal. It's enough to provide a procedure that will guarantee the randomness of choosing a simple p in the ElGamal system, and then the fact of the ElGamal algorithm cryptostability denial is not valid anymore. It should be noted that the numbers of the special type, which weaken the stability of the ElGamal method, are very rare, therefore, the chance of their choice can be neglected.

4 EXPERIMENTS

Choice of research parameters. For experimental research the following parameters of encryption algorithms are chosen:

1. Time of keys generation.

For key generation time we take the time needed to determine all open and secret keys by the encryption algorithm. The key generation time of the algorithm depends on the size (number of bits) of the key. It is calculated in seconds or milliseconds.

2. Time of encryption.

Time of encryption is the time that an encryption algorithm needs to convert plain text to encrypted format.

3. Time of decryption.

Times of decryption is considered to be the time that an encryption algorithm requires to recreate plain text from encrypted text.

4. Bandwidth of the encryption process

The bandwidth of the encryption process is equal to the number of bytes of encrypted text divided by the time of encryption. The higher the bandwidth, the higher the performance of the method.

5. Bandwidth of the decryption process.

The bandwidth of the decryption process is equal to the number of bytes of decrypted text divided by the time of decryption.

6. The size of the encrypted file.

The size of the encrypted file is equal to the number of bytes of encrypted text.

7. The size of the decrypted file.

The size of the decrypted file is equal to the number of bytes of the recreated text.

Experimental tools and data. Experiments were conducted on Intel Core 2 Duo CPU 2.09 GHz with 4 GB of RAM under the operating system Windows 7.

Initially, testing was conducted for different key lengths. In this work, the number of bits of a private key was selected in accordance with NIST-recommendations [8]. Correspondence of key sizes that provide equivalent security levels in RSA and ElGamal algorithms is shown in Table 1. These five specific security levels were chosen because they represent the five appropriate levels of work required to perform key search using asymmetric encryption algorithms: SKIPJACK, TRIPLE-DES, AES-small, AES-medium and AES-large respectively [8]. The length of the message used for encryption was 105 KB.

Table 1 – Correspondence of closed keys sizes in algorithms RSA and ElGamal

Algorithms	Keys sizes (bit)				
RSA	1024	2048	3072	7680	15360
ElGamal	160	224	256	384	512

Then testing was conducted for different sizes of input files. At that, the size of the private key for the RSA algorithm was assumed to be 1024 bits, and for the ElGamal algorithm, respectively, 160 bits. The sizes of text files with which the tests were performed were selected for 68 KB, 105 KB, 124 KB and 235 KB. To achieve a satisfactory level of the parameter values reliability, each operation for each test parameter was performed 20 times and its average value was calculated.

Test results. The results of the tests are presented in the tables below. Table 2 show the results of the tests for different lengths of RSA and ElGamal algorithm keys, respectively.

Table 3 show test results for the key of the same size, but for different sizes of input files. As a rule, it is recommended to use the 1024-bit key for the RSA algorithm and the corresponding 160-bit key for the ElGamal algorithm.

Table 2 – Results of RSA and ElGamal algorithms testing for different key lengths

	RSA key (bit)				ElGamal key (bit)				
	1024	2048	3072	7680	160	224	256	384	512
Key generation time (sec)	1,31	6,80	32,10	322,84	0,20	0,21	0,24	0,29	0,45
Encryption time (sec)	0,20	0,35	0,38	0,43	0,70	3,85	1,42	2,99	5,59
Decryption time (sec)	10,08	81,99	196,93	970,60	1,18	1,41	1,65	3,69	8,56

Table 3 – Test results of RSA and ElGamal algorithms for different sizes of input files

	RSA (1024- bit key)				ElGamal (160- bit key)			
	Input File Size (KB)				Input File Size (KB)			
	68	105	124	235	68	105	124	235
Encryption Time (sec)	0,16	0,20	0,32	0,62	0,475	0,69	0,74	1,92
Decryption Time (sec)	6,02	10,08	11,04	19,04	0,404	1,18	1,32	1,97
Encrypted File Size (KB)	85,79	151,50	172,67	331,87	136,00	210,01	249,00	470,01
Decrypted File Size (KB)	68	105	124	235	68	105	124	235
Encryption Bandwidth (KB/s)	532,87	751,22	544,70	536,14	286,48	304,83	335,06	244,26
Decryption Bandwidth (KB/s)	11,29	10,41	11,24	12,34	168,48	89,18	94,04	119,21

Analysis of the literature shows that similar studies were conducted by a number of authors and the results obtained by them selectively confirm some of the results presented in this paper. Thus, AderemiElishaOkeyinka [9] and Megah Mulya [10] compare the speed of encryption and decryption of text data of different sizes by RSA and Elgamal algorithms. In [11], a team of authors (Cindy Himawan, Toni Wibowo, Budi Sulityo, Rusdianto Roestam, Yuyu Wahyu, RB. Wahyu) conducts in-depth studies of RSA and ElGamal algorithms

regarding the influence of message symbols and key values on the duration of cryptographic processes and cryptographic processes memory and also considers the difference in the characteristics of the process results for each algorithm.

Jakir Hossain and Eklas Hossain in [12] and Shahzadi Farah and others in [13] provide graphical representations of the time of encryption and decryption, as well as the size of encrypted and decrypted files on the dimensions of input files for RSA, ElGamal and Paillier algorithms.

5 RESULTS (analysis of test results conducted for different key sizes)

Comparison of key generation time. In encryption algorithms, key generation time is the most important sub-process, which requires the generation of random numbers and testing them for simplicity. In the RSA algorithm an additional search is performed for an integer that is relatively simple with the Euler function value. This is a rather laborious process. It depends on the size of the key, but does not depend on the size of the input data.

For the time of key generation, the RSA algorithm received values ranging from 1312 ms to 322843 ms, as shown in Fig. 1. As a general rule, it is recommended to use a 1024-bit key, which is computed in 1312 ms.

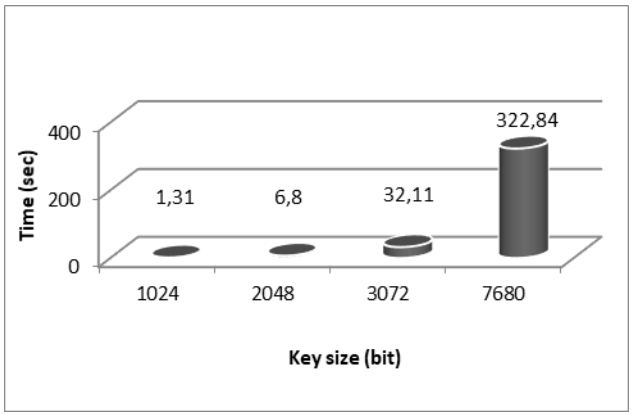

Figure 1 – Key Size Dependence of key generation time by RSA algorithm

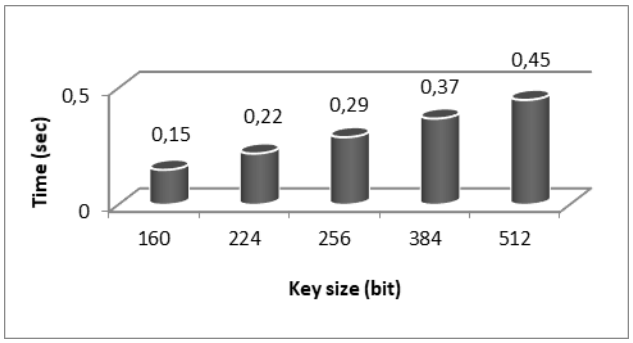

Figure 2 – Key Size Dependence of the key generating time by ElGamal algorithm

The time to generate keys by ElGamal algorithm, except for generating random numbers and testing them for simplicity, depends only on the size of the key.

For the keys generation time, the algorithm ElGamal received values in the range from 198 ms to 447 ms, as shown in Fig. 2. As a rule, it is recommended to use a 160-bit key, computing time of which is 145 milliseconds, that is approximately 10 times better than in the RSA algorithm. However, in the course of research, the team of authors in [11] obtained the same result as the authors of this article, that the computation time of the keys by the ElGamal algorithm is shorter than the RSA algorithm.

Comparison of encryption time. As is known, the time required to encrypt using fast exponentiation is proportional to the number of single bits in the exponent e. Therefore, simple numbers that contain a small number of single bits in a binary record are usually taken as the encryption key e, for example, the simple Fermat numbers 17, 257, or 65537. In the study of RSA algorithm encryption time the value e = 65537 was taken as e key, which is usually recommended for a 1024-bits key for commercial use. When encrypting a 105-kilobyte message for key sizes in the range from 1024 bits to 7680 bits, results were obtained in the range from 0.202 sec to 0.429 seconds, as shown in Fig. 3.

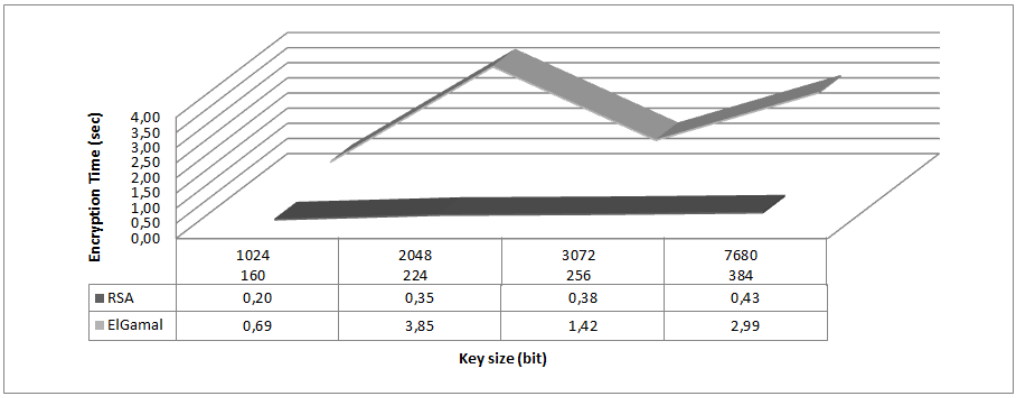

Figure 3 – The graph of the encryption process duration dependence on the key size

For testing the ElGamal algorithm when encrypting a 105-kilobyte message for key sizes in the range from 160 bits to 384 bits, the results were obtained in the range from 0.689 seconds to 3.847 sec (Fig. 3).

As can be seen from Fig. 3 that the time of the RSA algorithm encryption is better than ElGamal algorithm encryption for all key sizes.

Comparison of decryption time. In the cryptosystems, the Chinese Remnant Theorem is used to facilitate decryption operations, which asserts that if the *n* number prime factorization

$$n = n_1 \cdot \ldots \cdot n_k$$

is known, where all n_i are pairwise mutually prime numbers, and the result of bringing the number *x* to the module $n_i \; \forall i = 1,\ldots,k$ is the same, then the result of bringing the number *x* to the module *n* will be the same number, i.e.

$$\forall x,\ a - \text{integers},\ x \equiv a \bmod n \Leftrightarrow x \equiv a \bmod n_i \; \forall i = 1,\ldots,k\ .$$

When investigating decryption time of an encrypted 105-kilobyte message for key sizes in the range from 1024 bits to 7680 bits with the RSA algorithm, the results were obtained in the range of 10.082 seconds to 970.597 seconds, as shown in Fig. 4.

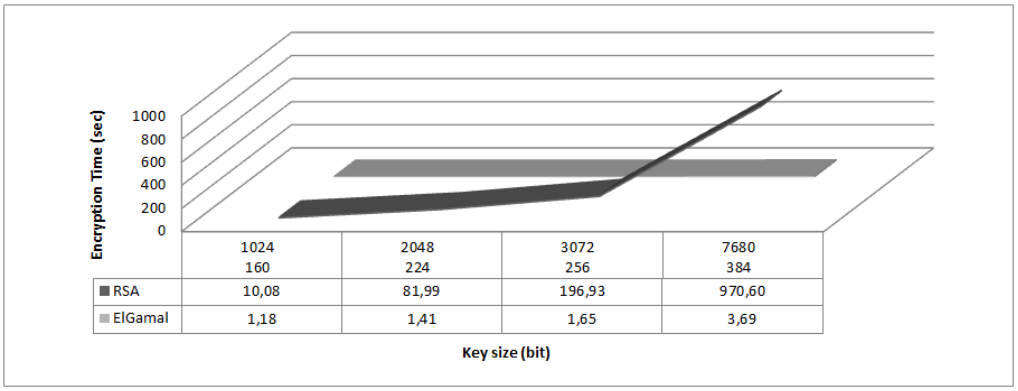

Figure 4 –The graph of the decryption process duration dependence on the key size

When investigating ElGamal algorithm decryption of encrypted 105-kilobyte message for key sizes in the range from 160 bits to 384 bits, the results were obtained in the range from 1.177 sec to 3.689 sec, as shown in Fig. 4.

The analysis shows that both algorithms have almost the same good results with a low level of security. But with the keys size increase, the decryption time of the RSA algorithm increases exponentially, while the decryption time of the algorithm ElGamal has a linear growth order

Analysis of test results conducted for different sizes of incoming messages.

For further research, the key size for the RSA algorithm was 1024 bits, and for the ElGamal algorithm, respectively, 160 bits, which corresponds to the current recommendations for these algorithms use.

The sizes of text files selected for performed tests were 68, 105, 124 and 235 KB.

Comparing the size of the source files. Fig. 5 and Fig. 6 show the comparison of encrypted and decrypted files sizes respectively for RSA and ElGamal algorithms.

RSA algorithm encrypted messages received results ranging from 85.792 KB to 331.868 KB, and for ElGamal algorithm in the range from 136.003 KB to 470.012 KB. When decrypting messages with both algorithms, the size of the decrypted files coincided with the size of the corresponding input files.

When encrypting with the RSA and ElGamal algorithms, the size of the encrypted data depends on the key size and the input data size.

The comparison showed that the RSA algorithm provides the best savings for the bandwidth. At this, the encrypted data size is larger than the input data size on average by a factor of 1.4. The encrypted data size in the ElGamal algorithm is almost twice as long as the input data.

Also, the numerical values and graphs for comparing file sizes obtained in this paper are in line with the numerical values and graphs given in [13].

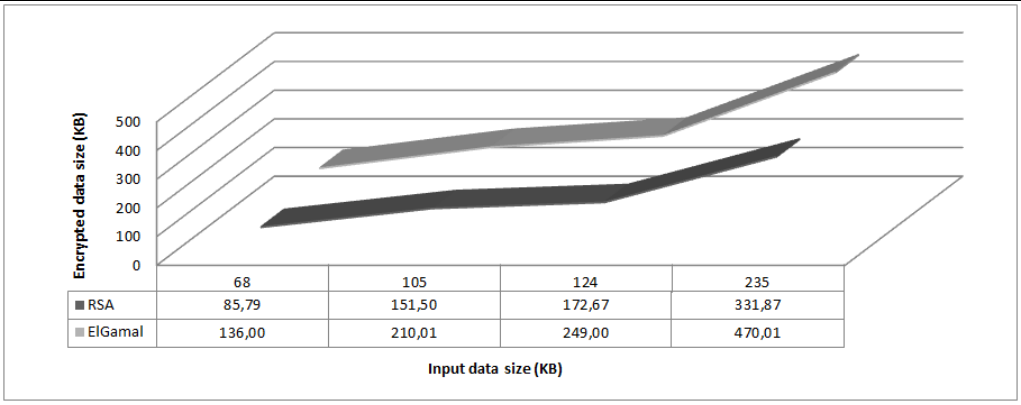

Figure 5 – The graph of the encrypted data size dependence on the input datasize

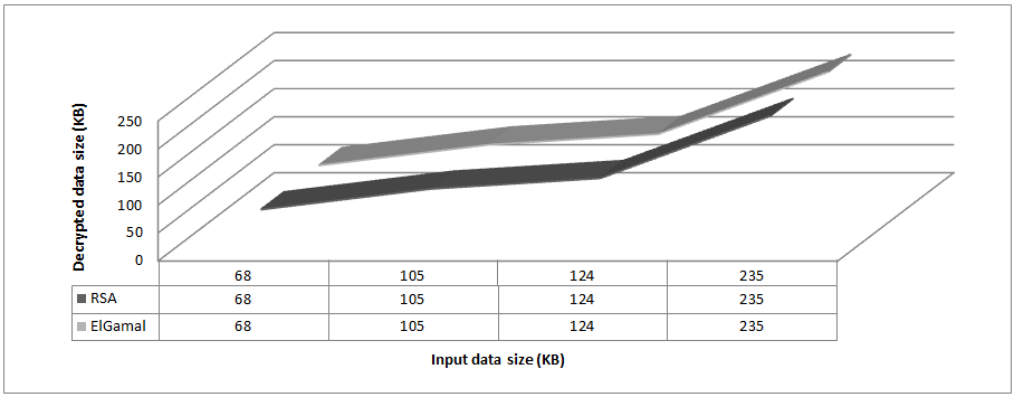

Figure 6 – The graph of the decrypted data size dependence on the input datasize

Comparison of encryption time and decryption time. In the investigation of RSA algorithm encryption time, the results were obtained in the range from 0.161 seconds to 0.619 seconds, as shown in Fig. 7, and when decrypted by this algorithm – in the range from 6.021 s to 19.043 s, as shown in Fig. 8.

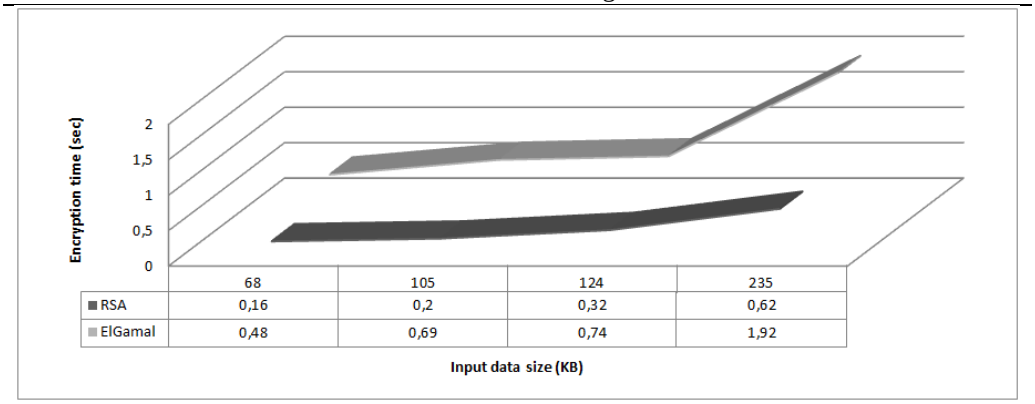

Figure 7 – The graph of the encryption process duration dependence on the input datasize

In testing with the ElGamal algorithm results were obtained in the range from 0.475 seconds to 1.924 seconds, as shown in Fig. 7, and when decrypted by this algorithm – in the range from 0.404 seconds to 1.971 seconds, as shown in Fig. 8.

In general, the dynamics of the processes presented in this article are graphs of the dependence of the encryption time and the decryption time on the size of the input data for the RSA and ElGamal algorithms coincide with the graphs presented in [12] and [13]. The differences in numerical values are explained by the differences in the capacities of the computing equipment, on which specific studies were conducted.

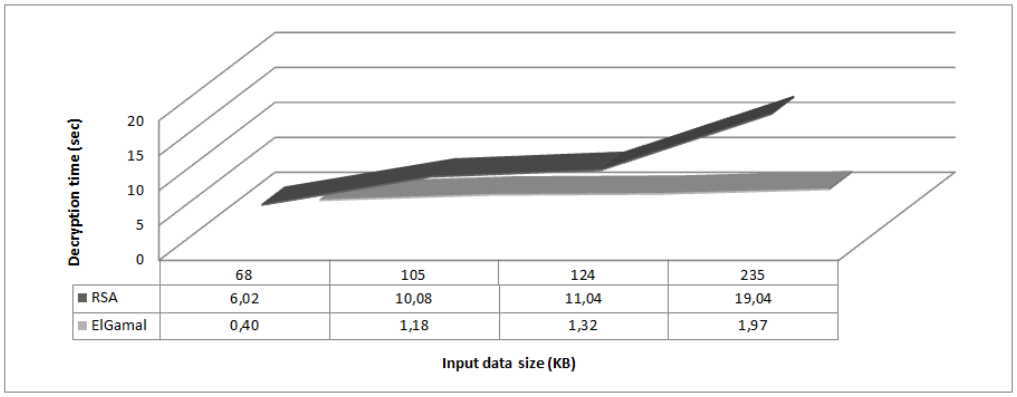

Figure 8 – The graph of the decryption process duration dependence on the input datasize

The comparison of the algorithms encryption and decryption processes duration showed that the RSA algorithm has higher performance during encryption, and ElGamal algorithm is better during decryption. The same conclusion was reached by the authors in [14].

Comparison of algorithms bandwidth. Bandwidth is the most important parameter that demonstrates the effectiveness of any algorithm. Fig. 9 shows the bandwidth of RSA and ElGamal algorithms for the encryption process and Fig. 10 for the decryption process.

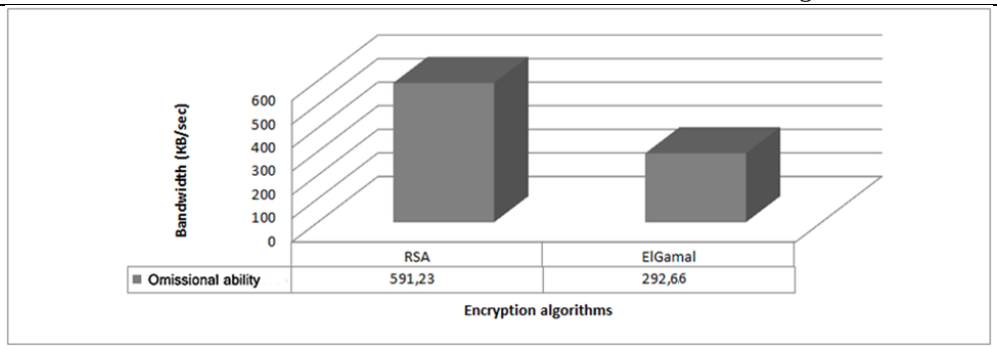

Figure 9 – Bandwidth of RSA and ElGamal algorithms in the encryption process

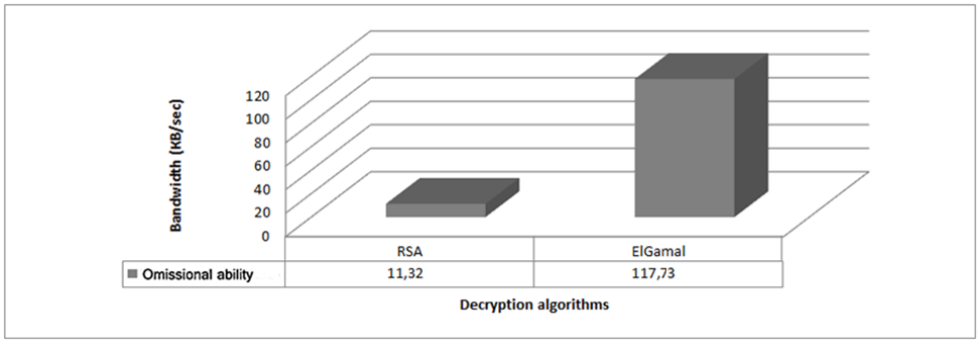

Figure 10 – Bandwidth of RSA and ElGamal algorithms in the decryption process

As can be seen from these diagrams, the RSA algorithm showed 2 times better bandwidth than the ElGamal algorithm in the encryption process, but the algorithm ElGamal showed 10 times better bandwidth than the RSA algorithm in the decryption process.

These results are confirmed by Okeyinka A.E. in his work [9]. He concludes that although the RSA algorithm overall outperforms the Elgamal algorithm, it is not as effective as the benefits of Elgamal when considering the speed of data decryption.

A similar view is held by Megah Mulya. He claims that the sequence of descending encryption rates is RSA, then Elgamal and then Eliptic Curve. Unlike encryption, the order of decryption rates in descending order is Elgamal, followed by Eliptic Curve, then RSA [10].

6 DISCUSSION

In this paper we have comprehensively compared the performance of two cryptographic algorithms to determine which algorithm is best suited for a particular application. The comparative analysis is based on such characteristics as the efficiency of computer memory usage, the duration of the key generation process, the speed of the encryption and decryption processes, the throughput of the algorithm, the dimensions of the keys, the volumes of encrypted and decrypted files. Although both algorithms are quite popular methods in data protection practice, we can conclude from the results analysis and discussion that the RSA algorithm performs the encryption process faster and the ElGamal algorithm is the decryption process.

CONCLUSIONS

Summarizing the conducted research, it can be stated:

1. For all key lengths, the ElGamal algorithm creates a pair of open and closed keys 10 times faster than the RSA algorithm, which is especially noticeable with a significant increase of key sizes.

2. For the ElGamal algorithm, the key generation time increases linearly with the increase of key size, while for the RSA algorithm it increases in geometric progression.

3. The time of encryption of the message by the RSA algorithm is better than the ElGamal algorithm for all key lengths.

4. When decoding, both algorithms show almost identical good results with a low level of security, but with increasing key sizes, decoding time with the RSA algorithm increases exponentially, while the decoding time with the ElGamal algorithm has a linear order of growth.

5. The RSA algorithm provides the best savings for the bandwidth. At this, the size of the encrypted RSA data algorithm is larger than the input data on average by a factor of 1.4. The length of the encrypted data in the ElGamal algorithm is twice as long as the input data.

6. When decrypting messages with both algorithms, the size of the decrypted files coincided with the size of the corresponding input files.

7. The RSA algorithm has a higher speed of encryption, while ElGamal algorithm is better during decryption.

8. The RSA algorithm showed 2 times better bandwidth than the ElGamal algorithm in the encryption process, but the ElGamal algorithm showed 10 times better bandwidth than the RSA algorithm in the decryption process.

9. The cryptostability of the algorithm ElGamal is much better than the cryptostability of the RSA algorithm.

To increase the speed of algorithms, you can apply a method of the key length reduction. However, such an increase in speed can reduce the cryptostability of the algorithm. This approach is recommended to use, for example, if there is a need to encrypt data that loses relevance over a short period of time. In the case of the impossibility cryptostability reduction, it is proposed to increase the speed through the computations parallelization in multiprocessor systems.

The scientific novelty of the obtained results is that with the help of the developed software product the comparative analysis of asymmetric algorithms of encryption of text data, their advantages and disadvantages is thoroughly investigated and an experimental evaluation of their characteristics is given.

The practical significance of the obtained results is that the results of the experiments allow us to provide suggestions and recommendations for the use of the investigated algorithms.

Prospects for further research: it is expedient to continue work on this topic with the use of achieved results. This work can become the basis or component of a larger project, for which the important factor is the preservation of authenticity and security of information.

REFERENCES

1. *Rivest R.* A method for obtaining digital signatures and public-key cryptosystems / R. Rivest, A. Shamir, L. Adleman // New York City: ACM, 1978. – Vol. 21, N 2. – P. 120–126. DOI : 10.1145/357980.358017
2. *Johnson J.* Public-key cryptography standards (PKCS) #1 / J. Johnson, B. Kaliski // RSA Cryptography Specifications Version 2.1. Network Working Group, 2003. DOI: 10.17487/RFC3447
3. *Boneh D.* Twenty years of attacks on the RSA cryptosystem / D. Boneh // Notices Amer. Math. Soc. F. Morgan-AMS. – 1999. – Vol. 46, N 2. – P. 203–213.
4. *ElGamal T.* A public-key cryptosystem and a signature scheme based on discrete logarithms / T. ElGamal // Advances in Cryptology-CRYPTO'84. – 1985. – P. 10–18. DOI : 10.1109/TIT.1985.1057074
5. *Tsiounis Y.* On the security of ElGamal based encryption / Y. Tsiounis, M. Yung // Public Key Cryptography. PKC-1998. –Berlin, Heidelberg : Springer, 1998. – Vol. 1431. – P. 117–134.
6. *Bakhtiari M.* Serious Security Weakness in RSA Cryptosystem / M. Bakhtiari, M.A. Maarof // IJCSI – 2012. – Vol. 9, N 3. – P. 175–178.
7. *Lenstra A.K.* Factoring by electronic mail / A.K. Lenstra, M.S. Manasse // Advances in Cryptology EUROCRYPT '89 Proceeding, Springer Verlag, 1990. – P. 355–371. DOI : 10.1007/3-540-46885-4_35
8. *Barker E.B., Roginsky A.M.* Transitioning the Use of Cryptographic Algorithms and Key Lengths. National Institute of Standards and Technology, Gaithersburg, Maryland, NIST Special Publication (SP) 800-131A, Rev. 2, March 2019. DOI : 10.6028/NIST.SP.800-131Ar2
9. *Okeyinka A. E.* Computational speeds analysis of RSA and ElGamal algorithms on text data / A. E. Okeyinka // Proceedings of the world congress on engineering and computer science, 21-23 October 2015, San Francisco, USA. – 2015. – P. 115-118.
10. *Mulya M.* Perbandingan kecepatan algoritma kriptografi asimetri / Mulya M. // Journal of Research in Computer Science and Applications. – 2013. – Vol. I, N 2. – P. 7–12.
11. *Himawan C.* Studi perbandingan algoritma RSA dan algoritma El-Gamal / C. Himawan, T. Wibowo, B. Sulityo, R. Roestam, Y. Wahyu, R.B. Wahyu // Seminar National APTIKOM (SEMNASTIKOM), Hotel Lombok Raya Mataram, 28-29 Oktober 2016. – Vol 1, N 1. – P. 695–700.
12. *Hossain J.* A novel keyless and key based encryption algorithm to handle cyber security in microgrid application / J. Hossain, E. Hossain // Gazi University Journal of Science 30(4), Dec. 2017. – 2017. – P. 330–343.
13. *Farah S.* An experimental study on performance evaluation of asymmetric encryption algorithms / S. Farah, M. Younas Javed, A. Shamim, T. Nawaz // WSEAS 3rd European conference of computer science (WSEAS ECCS 2012), Greece, Dec. – 2012 – P. 121-124.
14. *Siahaan A., Elviwani, Oktaviana B.* Comparative analysis of RSA and ElGamal cryptographic public-key algorithms, 2018. DOI: 10.4108/eai.23-4-2018.2277584

ISSN 2224-087X. Electronics and information technologies. 2020. Issue 14

ПОРІВНЯЛЬНИЙ АНАЛІЗ МЕТОДІВ АСИМЕТРИЧНОГО ШИФРУВАННЯ ТЕКСТОВИХ ДАНИХ

Т. Матвейчук, В. Смичок, С. Філімонов

Кафедра електромеханіки та електроніки,
Національна академія сухопутних військ імені гетьмана Петра Сагайдачного,
вул. Героїв Майдану, 32, Львів 79012, Україна
tais-28@ukr.net , smychok@ukr.net , sergnf@gmail.com

Актуальність. Актуальність роботи обумовлена проблемою розроблення ефективних методів захисту інформації в інформаційних мережах, що забезпечують надійне функціонування автоматизованих систем військового призначення, тобто всі обчислювальні системи, системи управління та зв'язку. Об'єктом дослідження є здійснення комплексу експериментів, метою яких є порівняння алгоритмів асиметричного шифрування текстових даних.

Метод. Цикл експериментальних випробувань здійснювався за такими параметрами алгоритмів шифрування: час генерування ключів; час шифрування і дешифрування; пропускна здатність процесу шифрування і дешифрування; розмір зашифрованого і дешифрованого файлу. Експерименти проведені на Intel Core 2 Duo CPU процесор 2.09 ГГц з 4 ГБ оперативної пам'яті під операційною системою Windows 7.

Результати. Отримані основні оціночні результати полягають в наступному: криптостійкість алгоритму ElGamal значно вище криптостійкості алгоритму RSA; алгоритм RSA має більш високу швидкість при зашифровуванні інформації, а алгоритм ElGamal кращий під час розшифрування; при збільшенні розмірів ключів час розшифрування алгоритмом RSA зростає експоненціально, в той час, як тривалість розшифрування алгоритмом ElGamal використовує лінійний порядок зростання; алгоритм RSA показав в 2 рази кращу пропускну здатність ніж алгоритм ElGamal в процесі зашифровування інформації, зате алгоритм ElGamal показав в 10 разів кращу пропускну здатність у порівнянні з алгоритмом RSA в процесі розшифровки інформації; довжина зашифрованих даних алгоритмом ElGamal в 2 рази довше вихідних даних, в той час як розмір зашифрованих алгоритмом RSA даних більше розміру вихідних даних в середньому на коефіцієнт 1,4; для всіх довжин ключів алгоритм ElGamal створює пару відкритого і закритого ключів в середньому в 10 разів швидше, ніж алгоритм RSA, що особливо помітно при значному збільшенні розмірів ключів; для алгоритму ElGamal час генерування ключів зростає лінійно зі збільшенням розмірів ключів, в той час як для алгоритму RSA воно росте в геометричній прогресії.

Висновки. За допомогою розробленого програмного продукту проведено порівняльний аналіз асиметричних алгоритмів шифрування текстових даних, їх переваг і недоліків, криптографічної стійкості, дано експериментальну оцінку їх характеристик щодо ефективності використання ними пам'яті комп'ютера, тривалості процесів генерування ключів, зашифровування і розшифрування даних, пропускної здатності алгоритмів, розмірностей ключів, обсягів зашифрованих і розшифрованих файлів. На підставі отриманих результатів надано рекомендації застосування розглянутих методів шифрування.

***Ключові слова*:** програмне забезпечення, асиметричні алгоритми, шифрування, дешифрування, крипостійкість, алгоритми RSA, ElGamal.

Стаття: надійшла до редакції 03.07.2020,
доопрацьована 01.12.2020,
прийнята до друку 02.12.2020