УДК 004.056.55: 003.26 DOI: 10.26661/2413-6549-2019-1-18

BLOCK CIPHER MODES IN THE DENIABLE ENCRYPTION

A. V. Galchenko[1], S. V. Choporov[2]

[1] *Zaporizhzhia National University, ORCID: 0000-0002-2258-9755*
[2] *Zaporizhzhia National University, ORCID: 0000-0001-5932-952X*
s.choporoff@znu.edu.ua

Key words:
algorithm, block cipher, deniable encryption, encryption mode, text encryption.

The problems of unauthorized access to the top systems and data stealing are considered. The deniable encryption is one of the most common in computer systems. But this encryption has some restrictions that are related to the capacity and block cipher. In this article, the block cipher algorithm has been developed. This block cipher has good capacity. Nevertheless, electronic codebook encryption mode causes the vulnerability. The objective of the article is to improve the deniable encryption algorithm searching the safe mode. Initially, possible modes of the block cipher are studied. Next, each mode is tested in block cipher for capacity and vulnerability. Finally, the optimal mode is determined. The optimal mode allows to patch the vulnerability. In addition, the modes are classified for data processing in parallel.

РЕЖИМИ БЛОЧНОГО ШИФРУ У ЗАПЕРЕЧНОМУ ШИФРУВАННІ

А. В. Гальченко[1], С. В. Чопоров[2]

[1] *Запорізький національний університет, ORCID: 0000-0002-2258-9755*
[2] *Запорізький національний університет, ORCID: 0000-0001-5932-952X*
s.choporoff@znu.edu.ua

Ключові слова:
алгоритм, блочний шифр, заперечне шифрування, режим шифрування, шифрування текстів.

У даній статті розглянуто проблеми несанкціонованого доступу та викрадення конфіденційних даних, причини, які призводять до їх виникнення, та можливі наслідки. Для вирішення зазначених проблем запропоновано використовувати криптографічні системи, які ґрунтуються на методі заперечуваного шифрування даних. Разом з тим, технічні обмеження, пов'язані з реалізацією алгоритмів на базі заперечуваного шифрування, суттєво впливають на продуктивність їх роботи та обмежують їх практичне використання. Для розв'язання цієї задачі запропоновано підхід для побудови блочних схем шифрування даних з використанням алгоритмів заперечуваного шифрування. У результаті цього розроблено прототип блочного алгоритму заперечуваного шифрування даних, який має досить високу продуктивність. Але в ході досліджень прототипу встановлено, що в запропонованому підході використовується режим шифрування електронною кодовою книгою, недоліки безпеки якого суттєво впливають на надійність кінцевого алгоритму шифрування. Основною метою даної статті є вивчення існуючих режимів блокового шифрування, продуктивності та безпеки їх реалізацій, а також пошук оптимальних режимів для подальшого використання в алгоритмах заперечуваного шифрування даних, без зменшення рівня їх захищеності. За результатами проведених досліджень досліджено існуючі режими шифрування та їх реалізації. З-поміж зазначених режимів виявлено найбільш оптимальний, який не впливає на рівень захищеності кінцевих алгоритмів шифрування. Додатково встановлено, що обраний режим шифрування дозволяє не лише реалізувати вихідні алгоритми заперечуваного шифрування, але й підвищити їх продуктивність за допомогою багатопотокової обробки даних.

Introduction. In the last several decades, a large number of private and public companies have been attacked by crime and the situation is escalated [1]. Possible targets of such attacks are corporate networks and IoT devices. Typically, attackers try to get business-sensitive data or user credentials. The target of plenty cyber-attacks is the another popular target of cyber-attacks.

A set of well-known techniques has been used for the top systems protection [1]. Nevertheless, some protected information systems have been compromised. In some cases, the penetration has been caused by violating of security requirements. It has become to the users' keys stealing. That's why authors have suggested the deniable encryption algorithms using, which doesn't depend on keys using [2].

The subject of the research is the blocked deniable encryption.

The objective of the research is to improve the deniable encryption algorithm searching the safe mode.

Formal Problem Statement. The reliability of traditional encryption schemes is based on keys secrecy. However, such systems aren't completely protected. The most dangerous cyber-attack is the coercion to users of telecommunication systems. Large keys, schemes without keys and neural networks are typical solutions of this problem. Nevertheless, traditional encryption schemes have private data that might cause vulnerabilities [1].

In [2], authors suggested to use deniable encryption algorithms to protect users from the coercion and their own data from leakage. The reliability of these algorithms is based on probabilistic encryption schemes (see Fig. 1).Initially, both plain and private texts are employed as input data of the encryption algorithm (see Fig. 1). Next, the encryption algorithm transforms these texts in a ciphertext using a private key. Finally, the user must submit the same private key into the decryption algorithm. The deniable encryption schema doesn't defend the private keys from the coercion attack, but this schema denies the existence of the ciphertext.

However, the deniable encryption has the following restrictions.

1. The size of the plain text depends on the key size as the following inequality:

$$||\text{Plain text}|| < ||\text{Key}||. \quad (1)$$

2. The limited possibility of the implementation block ciphers schemes into the deniable encryption algorithms: encryption modes, SB-Transforms, round encryptions, etc.

3. The low performance of data processing (during encryption and decryption).

The first restriction has been solved by the developing of basic block deniable encryption scheme in [2]. The performance of the block cipher method has been improved in [3].

Nevertheless, the vulnerability of in the mode of block encryption has been found during the primary analysis of the algorithm. This vulnerability is associated with the process of ciphertext blocks substitution. A cybercriminal could catch user for lie during the analysis of decrypted prepared block data.

The same vulnerability exists in the block cipher mode. This vulnerability is called «electronic code book» (ECB), but it's lacking in other cipher modes (see Fig. 2). Initially, the encryption algorithm gets the public data file **M** and the private data file **T**. Next, these files are divided into k blocks of the equal size $(\mathbf{M}_i, \mathbf{T}_i) < N$ ($i = 1, 2, \ldots, k$). Each block is pre-processed for the encryption by the function f_P. Pre-processed blocks are encrypted into ciphertexts $\mathbf{C}_i$ by the function f_E with the private key. Finally, ciphertexts $\mathbf{C}_i$ are joined into the ciphertext **C**.

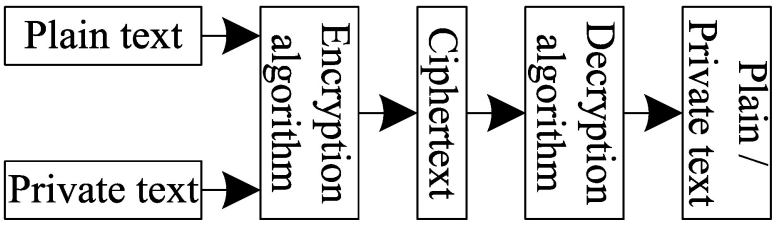

Fig. 1. The deniable encryption schema

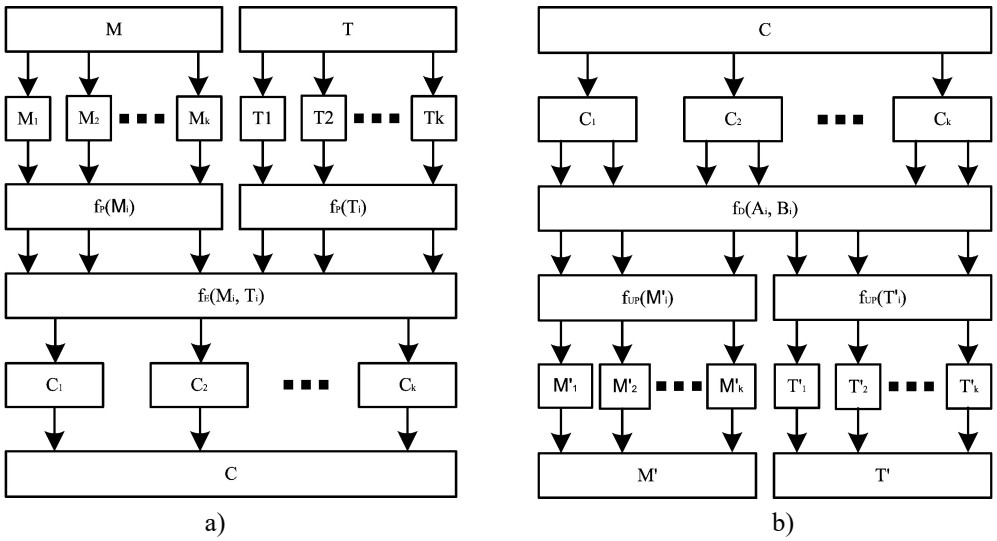

Fig. 2. The deniable encryption schema in the ECB mode: a) encryption, b) decryption

In the decryption algorithm, all operations are performed in the reverse order (see Fig. 2, b). Initially, the decryption algorithm gets the ciphertext **C** and the private key. Next, the ciphertext is divided into k blocks $(\mathbf{A}_i, \mathbf{B}_i) < N$ ($i = 1, 2, \ldots, k$). Each block is decrypted by the function f_D. Decrypted blocks are processed by the function f_{UP}. This function allows to separate private blocks from public blocks. Finally, private blocks are joined into the private data file **T** and public blocks are joined into the public data file **M**.

However, cybercriminals could get the private key using specially pre-processed data or the ciphertext.

Literature Review. Deniable encryption algorithms are known since 1990. The original deniable encryption algorithm has been suggested in [2]. This algorithm is based on the bitwise encryption. The algorithm could encrypt large size data but it doesn't support the block encryption.

The RD-PKE algorithm has been developed by Hamada Ibrahim in [5]. This algorithm includes mRSA scheme keys and the protocol. In this algorithm, a plain text must be transformed into against bits and then inputted into the prepared container. Furthermore, the algorithm requires a third-party to authorize the decrypting of plain text. Nevertheless, the algorithm doesn't allow to deny the plain text existence, because unauthorized decrypting might return broken data. This algorithm also has the restriction specified by the inequality (1). Hence, this algorithm also requires high computing resources.

RSA and El-Gamal encryption schemes have been combined by Jing-Quing Wang and Bo Meng in [6]. This algorithm has good reliability, but the size of initial data is also restricted by (1). Thus, this encryption scheme also requires high computing resources. In addition, it also allows to catch public and private data during the ciphertext decryption.

In [6], S. Goldwasser and C. Mikali suggested an invulnerable pseudo-probabilistic encryption algorithm. This algorithm doesn't allow an unauthorized access to any part of the plain text. It requires to use the same private key to decrypt each block. But there is only one validated block that user could decrypt at the same time. Hence, user couldn't deny the existence of any secret data in case of coerces using.

Finally, M. Moldovyan has suggested the original encryption algorithm in [8]. This algorithm has modified for the block encryption in [3]. The original algorithm is based on the Rabin's encryption schema. This scheme requires to input public data and private data to transform them into the ciphertext. However, it's possible to get some public or private data during the ciphertext decryption. The algorithm allows user to deny the private data existence, but this algorithm also has the same restriction, as in [5].

In the end M. Moldovyan suggested the another deniable encryption algorithm in [9-11]. This algorithm supports the block data encryption. The reliability of this algorithm is based on common block encryption algorithms. But

the algorithm requires two different decryption key. Hence, users couldn't deny the private data existence, so users aren't protected from the coercion. Besides, this algorithm has low capacity and doesn't provide deniable encryption properties [2, 8].

Block Cipher Modes. The block cipher mode provides the privacy and authorization for users. This mode could be applied for the symmetric and some asymmetric encryption algorithms. Common cipher modes are following: electronic codebook (ECB), the cipher block chaining (CBC), cipher and output feedbacks (CFB and OFB), the counter (CTR). These modes have been described in [4].

The ECB mode is the first and the weakest algorithm. This mode masks patterns of data blocks in the encrypted data using the equation (2).

$$\mathbf{C}_i = \mathrm{f}_E(\mathbf{M}_i, \mathbf{K}). \quad (2)$$

Typically, the implementation of this mode is simple. However, it makes ciphers more vulnerable to reverse engineering attacks. Overall, it allows to get patterns of private data from the ciphertext. Hence, it isn't recommended to use, because other cipher modes haven't such vulnerabilities.

The second algorithm is the CBC mode. It suggests to encrypt a common block of plain texts $\mathbf{M}_i$ with the previous block of ciphertext $\mathbf{C}_{i-1}$ by the function f_E simultaneously:

$$\mathbf{C}_i = \mathrm{f}_E(\mathbf{M}_i \oplus \mathbf{C}_{i-1}). \quad (3)$$

The equation (3) adds the pseudo-randomness property to each block of the ciphertext. But the first block of the ciphertext requires some initialization vector **IV**. Blocks padding also causes a vulnerability. This vulnerability allows to steal the encryption key. Besides, the CBC mode allows to parallelize decryption. Generally, the CBC mode is more protected than the ECB mode.

The third algorithm is the CFB mode. This mode processes data like the CBC mode. The CFB mode transforms the plain text **M** into the ciphertext **C** using the function f_E. It also requires to save the initialization vector **IV** in the first block of the ciphertext $\mathbf{C}_0$:

$$\mathbf{C}_i = \mathrm{f}_E(\mathbf{C}_{i-1}) \oplus \mathbf{M}_i. \quad (4)$$

The FCB could be transformed in the stream cipher. However, this mode is vulnerable to bit changing in the ciphertext **C**. This vulnerability allows the cybercriminal to determine patterns in the nearest blocks of the ciphertext **C**. It also could be parallelized during the decryption.

The fourth algorithm is the OFB mode. This mode is also classified as a synchronous stream cipher. The ciphertext is generated from the pre-generated stream blocks and the blocks of plain text. But each block of the ciphertext depends on the previous encrypted blocks as the following equation:

$$\mathbf{C}_i = \mathrm{f}_E(\mathrm{f}_E(\mathbf{C}_{i-1})) \oplus \mathbf{M}_i. \quad (5)$$

The OFM model allows to use a hardware implementation of the CBC mode. But the OFM mode can't be implemented in parallel.

The last algorithm is the CTR mode. It's the same as the OFB mode. The ciphertext **C** is a result of the counter value $\mathbf{CTR}_i$ and the block of the plain text encryption:

$$\mathbf{C}_i = \mathrm{f}_E(\mathbf{CTR}_i) \oplus \mathbf{M}_i. \quad (6)$$

The CTR mode is suitable for in parallel computing systems. It allows a random access to properties during the decryption and supports a parallel encryption. Besides, this mode is widespread and haven't the ECB mode's weakness.

Generally, the encryption algorithms don't require data blocks padding in CFB, OFB, or CTR modes.

Models. In [3], the vulnerability of the deniable encryption has been found. The vulnerability is caused by the ECB mode; it allows to determine patterns of data in the ciphertext **C**. Hence a cybercriminal might to catch users on lying about the existing data. Some deniable encryption schemes and block cipher modes have been studied to prevent such vulnerabilities. As a result, some models of cipher modes have been developed. These models could be used in the deniable encryption algorithm.

The first model uses the CBC mode. In this model, public data **M** and private data **T** are transformed into the ciphertext **C**. The encryption algorithm f_E and the public are employed in the model. In the decryption process, public data **M'** and private data **T'** are recovered using the same private key and the function f_D. (**see Fig. 3**).

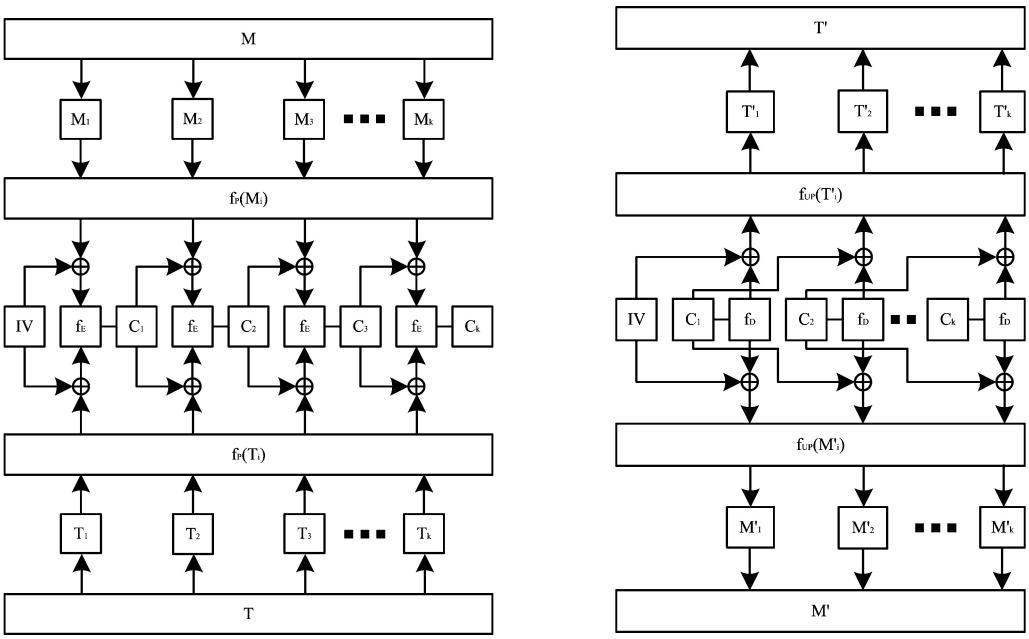

Fig. 3. The implementation of the CBC mode in the deniable encryption algorithm: a) encryption, b) decryption

The second model uses on the CFB mode. Temporary values and the initialization vector **IV** are encrypted in this model by the function f_E (**see Fig. 4**).

In the second model, the initialization vector **IV**, blocks of public data $\mathbf{M}_i$ and private data $\mathbf{T}_i$ are transformed into the ciphertext $\mathbf{C}_i$. Next, the initialization vector **IV** is changed by the previous block of ciphertext $\mathbf{C}_i$. The cycle of encryption continues, while there are blocks of data. The user must encrypt the initialization vector **IV** again to get blocks of public data $\mathbf{M'}_i$ or private data $\mathbf{T'}_i$. Finally, the model summates blocks of ciphertext C_i. Hence, public data **M** and private data **T** mustn't be decrypted to determine the encrypted content.

The third model is based on the OFB mode. This model includes some additional values (like the previous model). These values are also encrypted by the same encryption scheme f_E (see **Fig. 5**).

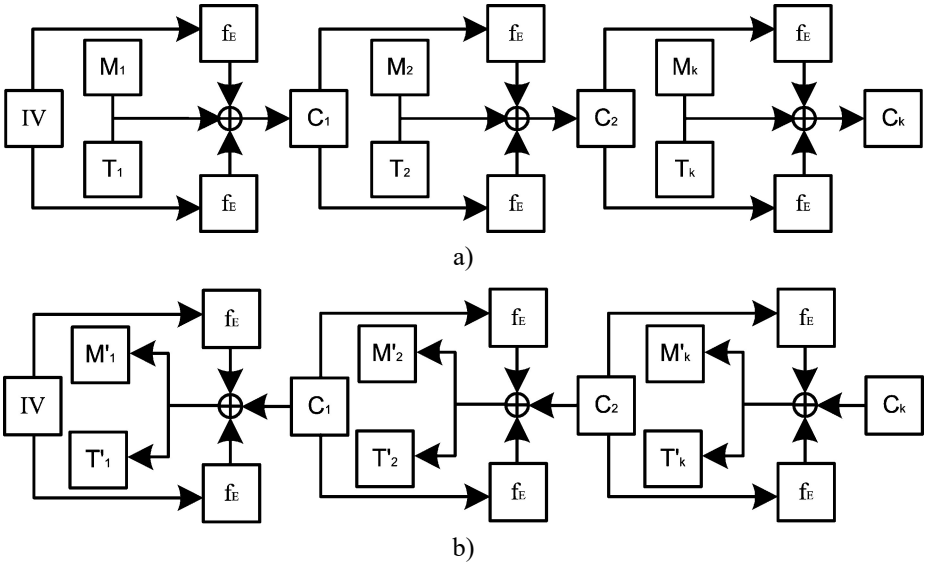

Fig. 4. The implementation of the CFB mode in the deniable encryption algorithm: a) encryption, b) decryption

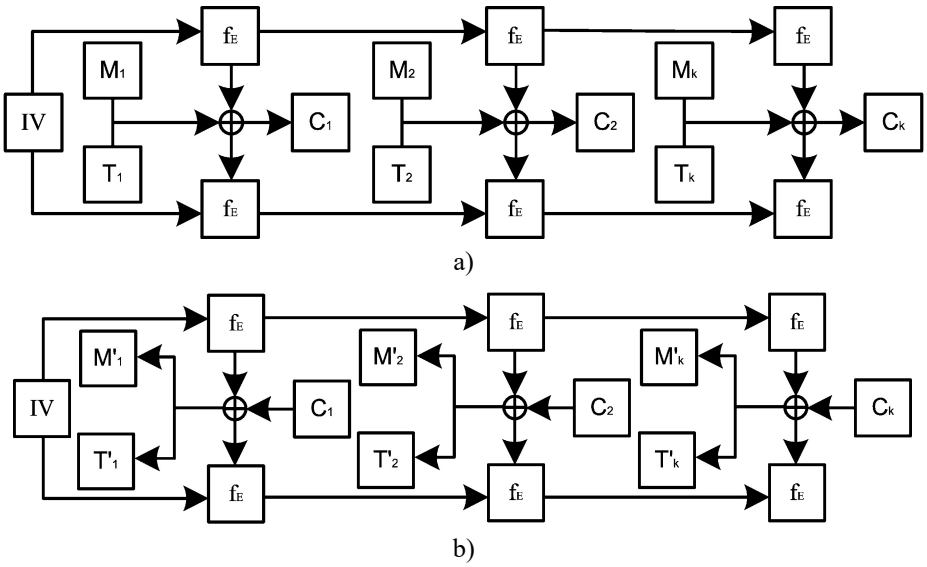

Fig. 5. The implementation of the OFB mode in the deniable encryption algorithm: a) encryption, b) decryption

In this model, the encrypted vector **IV** is summated with blocks of public data **M** and private data **T**. The user must encrypt the vector **IV** again. Next, it's summated with blocks of the ciphertext $\mathbf{C}_i$. Finally, the model returns blocks of public data $\mathbf{M}'_i$ and private data $\mathbf{T}'_i$. Hence, blocks of public data **M** and private data **T** mustn't be decrypted. This way changes the common security schema on the XOR encryption. It makes possible to determine the encrypted content.

The last model is based on the CTR mode. There are some random values $\mathbf{CTR}_i$ are used in this model. The block of the ciphertext $\mathbf{C}_i$ is a sum of the block of public data **M** or private data **T** with the encrypted counter value $\mathbf{CTR}_i$ (**see Fig. 6**).

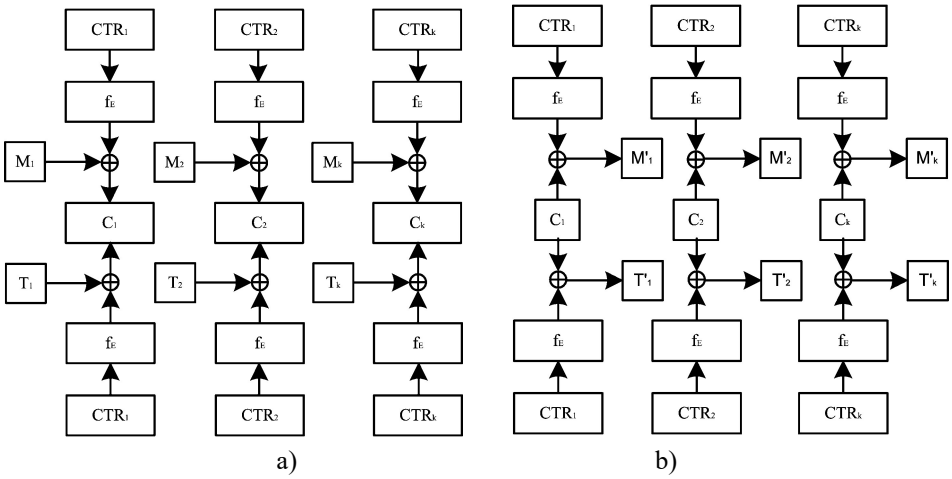

Fig. 6. The implementation of the CTR mode in the deniable encryption algorithm: a) encryption, b) decryption

Initially, some counter values $\mathbf{CTR}_i$ must be encrypted. Next, blocks of public data $\mathbf{M}_i$ or private data $\mathbf{T}_i$ could be evaluated. The encryption schema is used in this model. Thus, this schema has higher capacity than the ECB mode. But it also allows to determine the private content in the ciphertext.

Generally, the reliability of the CFB, OFB, CTR modes aren't based on deniable encryption algorithms.

Experiments and Results. Models of block cipher modes have been tested on the capacity and vulnerabilities. Experiments have been carried out using the following schema (see **Fig. 7**).

The optimal model has been determined by the following indicators: the deniability of common data, the random access to data, the parallel encryption and decryption, the leakage of keys, the brute force, the factorization, the discrete logarithm, the analysis of blocks. But the most important indicators are following: the deniability of common data, the parallel encryption and decryption, the leakage of keys and the analysis of blocks. Results of the experiment are collected in the Table 1 and **Fig. 8**.

Hence the ECB and the CBC modes allow to provide the deniable decryption of data. Public data **M** and private data **T** are decrypted by the same private key. But the ECB mode isn't recommended to use in the cryptography protocols. Other cipher modes (CFB, OFB, CTR) couldn't be used in the block deniable encryption algorithms. These modes don't provide the deniable decryption of data and allow to get access to private data in blocks of the ciphertext.

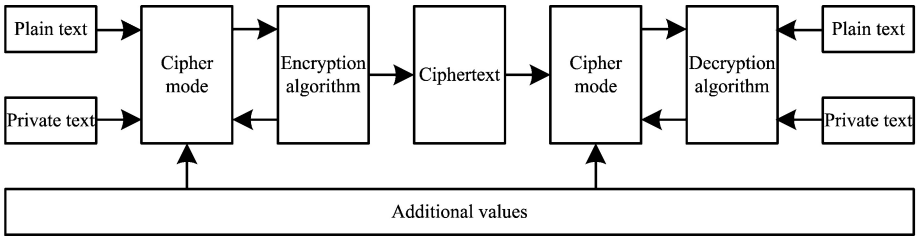

Fig. 7. Schema of the experiment

Table 1. Possibility of cipher modes implementation tests

No.	Indicators	The cipher modes				
		ECB	CBC	CFB	OFB	CTR
1	Basic indicators:	Implementation, units				
1.1	The deniable decryption	pass	pass	fail	fail	fail
1.2	The random access	pass	fail	fail	fail	pass
1.3	The parallelize encryption	pass	pass	pass	fail	pass
1.4	The parallelize decryption	pass	pass	pass	fail	pass
The average rank by basic characters		075%	056%	038%	000%	056%
2	Indicators of strength:	Implementation, units				
2.1	The key leakage	pass	pass	fail	fail	fail
2.2	The brute force	pass	fail	fail	fail	fail
2.3	The factorization	pass	pass	pass	pass	pass
2.4	The discrete logarithm	pass	pass	pass	pass	pass
2.5	The analysis of blocks	fail	pass	pass	pass	pass
The average rank to cyber-attacks strength		040%	040%	030%	030%	020%
Summary rank		058%	048%	034%	015%	038%

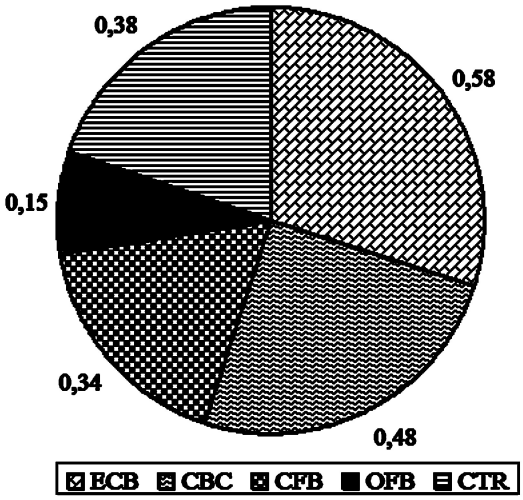

Fig. 8. The diagram of cipher modes implementation tests

Conclusion. The optimal cipher mode has been determined. This mode has allowed to patch the vulnerability of the ECB mode in the block deniable encryption algorithm [3].

The following results have been obtained: common cipher modes have been studied and applied to the deniable encryption scheme, cipher modes have been tested for capacity and vulnerability, the possibility of leakage data from the ciphertext has been found in some cipher modes. Hence, some cipher modes couldn't be applied in the common deniable encryption algorithms.

The novelty of work consists of study common cipher modes and their implementation in the deniable encryption algorithms. In [8, 9], using common block ciphers in the deniable encryption schemes has been suggested. However, these algorithms don't use cipher modes separately. Hence, these algorithms have low capacity comparing to described in [3].

The deniable encryption schema could be used with cipher modes that is more secured than the ECB mode.

The prospects for further researchers are to develop the parallel implementation of the block deniable encryption scheme, to implement other block ciphers schemes into the common deniable encryption algorithms, to evaluate the security of cipher.

References

1. Simis, B. (2019). Cybersecurity threatscape 2018: trends and forecasts. *Positive Technologies*. Retrieved from https://www.ptsecurity.com/ww-en/analytics/cybersecurity-threatscape-2018/ Accessed 25 March 2019.
2. Canetti, R., Dwork, C., Naor, M. & Ostrovsky, R. (1997) Deniable Encryption. *Advances in Cryptology*, pp. 90–104.
3. Galchenko, A. V. & Choporov, S. V. (2019). Deniable encryption based on hybrid cryptographic systems using. *Radio Electronics, Computer Science, Control*, pp. 178–191.
4. Bujari, D. & Aribas, E. (2017). Comparative Analysis of Block Cipher Modes of Operation. *International Advanced Researches & Engineering Congress-2017*, pp. 1345–1352.
5. Ibrahim, H. (2009). Receiver-Deniable Public-Key Encryption. *International Journal of Net-work Security*, pp. 159–165.
6. Bo Meng, Jiang Qing Wang (2009). A Receiver Deniable Encryption Scheme. *Proceedings of International Symposium on Information Processing (ISIP09)*, pp. 254–257.
7. Goldwasser, S. & Micali, C. (1984). Probabilistic encryption. *Journal of Computer and System Sciences*, pp. 270–299.
8. Moldovyan, N. A., Goryachev, A. A. & Vaichikauskas, M. A. (2014). Extension of Rabin cryptosystem: public key deniable encryption algorithm. *Information Security Questions*, No 1, pp. 12–16.
9. Moldovyan, N., Birichevskiy, A. R. & Mondikova, Y. (2014). Deniable encryption based on block ciphers. *Information and Control Systems*, No. 5(72), pp. 80–86. (in Russian)
10. Moldovyan, N. A., Moldovyan, A. A., Moldovyan, D. N. & Shcherbacov, V. A. (2016). Stream Deniable-Encryption Algorithms. *Computer Science Journal of Moldova*, pp. 68–82.
11. Moldovyan, A. A. & Moldovyan, N. A. (2018). Methods and Algorithms for Pseudo-Probabilistic Encryption with Shared Key. *PIIRAS Proceedings*, Issue 6(61), pp. 119–142.