Image-based symmetric message encryption method

Serhii Buchyk[1,†], Serhii Toliupa[1,*,†], Dmytro Tsapro[1,†], Anastasiia Shabanova[1,†] and Oleksandr Buchyk[1,†]

[1] *Taras Shevchenko National University of Kyiv, 60 Volodymyrska str., 01033 Kyiv, Ukraine*

Abstract

The paper is devoted to the development of a symmetric message encryption method based on the use of images. The presented method combines the book method, a disposable notebook, and the Diffie-Hellman protocol, which eliminates the disadvantages of book encryption, namely the transmission of special characters and the limitation of the key space. Modification of the Diffie-Hellman protocol for image transmission ensures the creation of the same unique image. Using such an image as a disposable notepad means that the stability will depend on the random number generator, which will be MT19937-64 with an initial startup time of seconds. To confirm the efficiency of the presented method, a software application has been developed that implements image mixing, subsequently forming a unique image for two users, and using it to perform the corresponding encoding using the book method.

Keywords

image, Diffie-Hellman algorithm, symmetric encryption, RGB, book coding

1. Introduction

The problem of information transmission and its protection dates back to ancient times, so as transmission methods have improved and become more popular, so have the methods of protection [1, 2]. At the current stage of development of information systems and technologies [3], global computer systems [4], and multimedia [5], the issue of ensuring the reliability and security of digitally stored data, as well as their reproduction and transmission via information communication channels, is extremely acute. Especially as wars evolve from full-scale conflicts into hybrid ones, it may seem that the main role will be played by information resources of state importance and big business, but collecting information on public sentiment is sometimes much more important, as it allows to identify hidden trends and prevent destabilization, providing a more accurate understanding of public sentiment and informed decision-making to maintain social stability.

Such data is usually discussed by people in private. Since it is the population that determines the political course of a state and its sovereignty, protecting private communications should be one of the top priorities of the state. The development of various encryption methods should be an important area of this work.

According to the RAS (Rise Above Research) study, around 1.13 trillion photos were taken in 2020 alone [6]. This means that if you divide the value by the total number of people with a phone, about 211 photos fall to each person. It should be noted that the number of images that can be used is much higher, as it does not include those created with graphic editors.

A study (February 2024) by Rise Above Research predicts that the number of photos taken and stored will increase in 2024 and continue to grow at a linear rate until 2028, with the vast majority of these photos being taken on smartphones. This will lead to a 10% increase in the number of photos taken this year, reaching around 1.8 trillion photos worldwide in 2024 [7].

Based on these facts, the paper proposes a cryptographic protection method that combines the book method as an encoding method and the Diffie-Hellman method for image transmission.

2. Analysis and problem statement

The use of various methods of cryptographic information protection is an important and urgent task, so when considering programming technologies for secure systems, cryptography is one of the most important areas. Of course, researchers are constantly looking for new ways to protect information and systems from cybercriminals. Hence, various methods emerge as a combination of best practices in terms of cryptographic protection.

For example, a well-known simple but rather reliable approach to security [8] is the use of the book cipher algorithm. It is stated in [8, p. 51] that “cryptanalysts generally agree that the book cipher, when used correctly, is virtually unbreakable”. This suggests that combining the use of a book cipher with other cryptographic methods can improve resistance to breaking.

Another well-known digital encryption method that provides a secure exchange of cryptographic keys between

CQPC-2024: Classic, Quantum, and Post-Quantum Cryptography, August 6, 2024, Kyiv, Ukraine

*Corresponding author.

†These authors contributed equally.

buchyk@knu.ua (S. Buchyk); tolupa@i.ua (S. Toliupa); dima.tsapro11@gmail.com (D. Tsapro); shabanovaa@knu.ua (A. Shabanova); alex8sbu@knu.ua (O. Buchyk)

0000-0003-0892-3494 (S. Buchyk); 0000-0002-1919-9174 (S. Toliupa); 0009-0003-4858-6258 (D. Tsapro); 0009-0008-4962-569X (A. Shabanova); 0000-0001-7102-2176 (O. Buchyk)

two parties over a public channel without transmitting their conversation is the Diffie-Hellman key exchange [9].

In [10], the authors point out that "today, encryption of color images is important to ensure their confidentiality during transmission over insecure networks or storage... Security analysis confirms that RGB image encryption is fast and secure against several known attacks, so it can be used in real-time applications where high security is required". Paper [10] presents a fast RGB image encryption algorithm based on the general characteristics of a simple image and an optimized pseudo-random sequence from a 1D logistics map.

Paper [11] presents a hybrid network security algorithm based on the Diffie-Hellman algorithm and the Text-to-Image encryption algorithm.

In [12], the authors propose to improve the critical stability of the DNA algorithm by combining the genetic algorithm and the Diffie-Hellman key exchange algorithm.

The purpose of the study is to combine message transmission and encryption algorithms, to create an application for data transmission using the Diffie-Hellman protocol, using images and the book method.

The object of the study is the process of encrypting and transmitting photos over insecure communication channels.

The subject of the study is the Diffie-Hellman protocol, the book method of encryption, and the complexity of finding the key.

3. Summary of the main material

This method of cryptographic protection is based on the book method of messaging: when users have the same book and replace words with a reference to a specific page and line in it. Of course, it is assumed that the attacker does not have this book or that it is extremely difficult for him to guess what kind of book it is. The method is easy to implement and secure, but its use in real life is difficult. Cryptanalysts agree that, if used properly, the book's cipher is virtually unbreakable, almost as good as a disposable notebook [8].

Of course, such a book can be not only a book as an object, but, say, a video file, an audio message, a song, or, as in the case of this work, an image.

That is, you need to choose a common image and use the values of its pixels, which, as you know, can be encoded in different systems: such as HSV, HUE, CMYK, and, probably, the most famous RGB. It is RGB that we will use. So now a small image, such as Fig. 1, which has a size of 284×177, i.e. 50268 pixels, can encode 150804 letters, special characters, or numbers. So, our interlocutor, who has the same photo, just needs to send the pixel coordinates (conventionally x and y) and one of the 3 colors corresponding to our letter, in the presented work the following correspondence of values is performed: 0—Red, 1—Green, 2—Blue.

Sometimes it is necessary to encode a message containing numbers or special characters, so it is very difficult to read the decoded text without spaces. To solve this problem, we used the ASCII table, which allows us to encode any characters present in it.

At the same time, there is a threat of using frequency analysis. Frequency analysis is the study of the frequency of letters or groups of letters in an encrypted text. The method is used to help break substitution ciphers (e.g., monoalphabetic substitution cipher, Caesar shift cipher). Frequency analysis involves counting the occurrence of each letter in the text and is based on the fact that in any particular piece of text, certain letters and combinations of letters occur with different frequencies [13]. Therefore, using one pixel for one letter is impractical, but this leads to another problem: if we use the ASCII system, the first number of our encoding pixels will be significantly reduced, for example, in Fig. 2, you can see that almost a third of the image is blue—this means that all blue values will automatically disappear (because ASCII accepts values only from 0 to 127), and the predominance of blue indicates that its values are close to 256. You may also encounter a situation, although its chance is very small when there is no corresponding value at all. To solve it, you need to add another parameter to pass to the interlocutor—a conditional number to which you need to add the color of our pixel and take mod 256. This operation will ensure that all pixels are used regardless of their color. For example, we want to encode the letter f with the first pixel and its blue value. The blue value of the first pixel is 226, and the letter f is 102. According to this value of the additional parameter, the result of the operation will be (226+102) mod 256 = 72. Therefore, all we need to transmit is 0 0 2 72. Also, since books do not consist of a single page, it is possible to use several images at once to increase the volume of messages. Of course, the number of this image will also need to be transmitted, so the final encoded symbol will look like 0 0 0 0 2 72, where the first number indicates the index of the photo, the second and third are the pixel coordinates ("x" and "y" respectively), the fourth is the base color, and the fifth is the value of the additional parameter.

The next problem is sharing photos, so the Diffie-Hellman transfer protocol was used to solve it. Diffie-Hellman key exchange is a digital encryption method that securely exchanges cryptographic keys between two parties over a public channel without transmitting their conversation over the Internet. Messages are transmitted using the Diffie-Hellman method according to the following formula, where g and n are primes representing the public keys, and k is the private key. Of course, we cannot substitute the values of the base pixels because their moduli are not large enough and the range includes zero, which will make the key selection very easy, so we need to create a table of primes that we will use to select the initial value with the interlocutor. This will increase the use of any prime numbers, i.e. in our case:

- A prime number from the table with the corresponding index equal to the value of the base color in the open image.
- A prime number from the table with an index as the sum of two values of other base colors in the open image.
- A prime number from the table with the corresponding color index in the closed image. A schematic image is shown in Fig. 1.

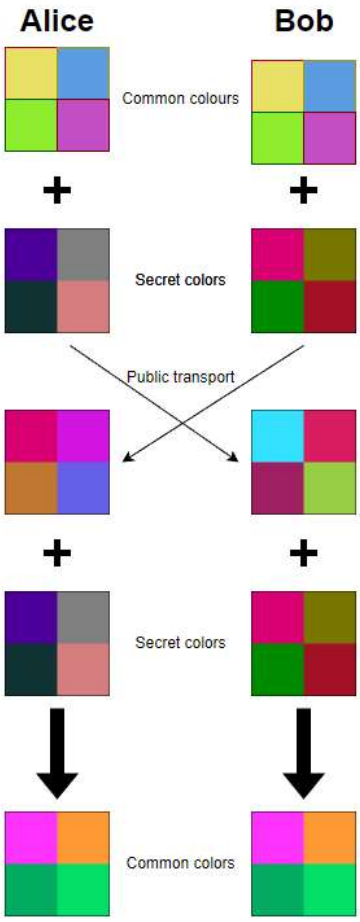

Figure 1: Schematic representation of the Diffie-Hellman method

The two parties use symmetric cryptography to encrypt and decrypt their messages, so the photos used will act as a public key (for example, Fig. 2) and a private key (Fig. 3 for the first user and Fig. 4 for the second user).

Figure 2: Open image

Criteria for images:

1. The open and closed images must not be identical.
2. The closed image must have more pixels than the open one, otherwise you reduce the number of coding pixels.
3. The images should not be monotonous (i.e., consist of only one color, as this will simplify the selection of the private key).

Figure 3: Private image for a first user

Figure 4: Private image for a second user

Thus, the algorithm consists of two stages: encrypting the message and forming a shared key with the interlocutor. The encryption will be performed by selecting a random pixel from the generated closed image and converting the character value from the ASCII table into a ciphertext. The ciphertext will consist of five values: 1—the number of the photo, in case you decide to create an array of closed images, 2 and 3—the coordinates of the randomly selected pixel (this pixel will not be used later), 4—its hue, i.e. red, green or blue, and 5—the value required to make the pixel parameter independent of the transmitted character.

The formation of a shared key is implemented using the Diffie-Hellman method and a table with primes, in which each prime is the corresponding index equal to the hue value of a particular pixel.

Let us consider the implementation of the symmetric message encryption method based on the use of images.

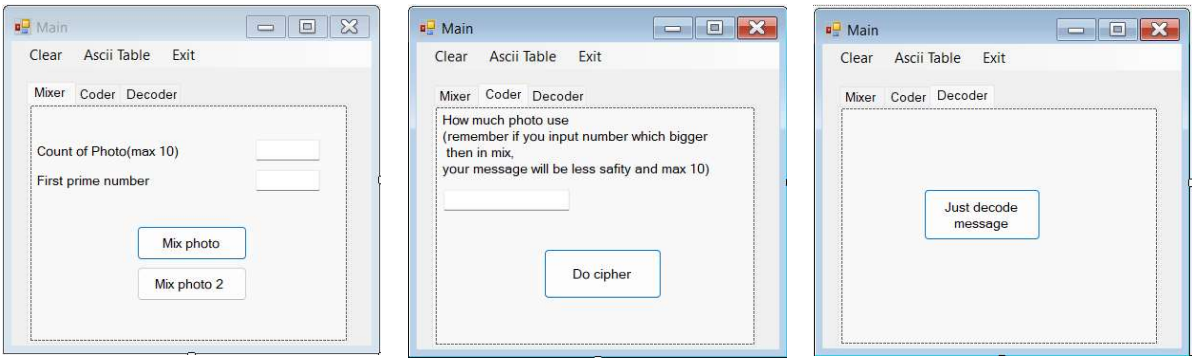

Figure 5: Interface of the implemented application

To demonstrate how the algorithm works, we have developed an application shown in Fig. 5. The application can mix several images—the Mixer section, encrypt user text with the appropriate number of mixed images—the Coder section, and decode the message that was sent—the Decoder section.

The mixing functions consist of 8 threads that call the exponentiation and mod functions. Since the numbers we are entering are of high power—they will be very large and will not fall within the range of ordinary C++ data types, we split the number into an array, where each index is a bit of the number, so the possible number is increased to 100000 characters instead of 36 for the largest long data type. The corresponding functions for exponentiation and mod were created.

The next problem is the transmission of these numbers, because when taking mod 256, our numbers will end up being different, so an auxiliary image was implemented to store the values that need to be multiplied by 256 and added to the value of the corresponding pixel color in the sent image. This increases the number of maximum values to 65536.

Hence the difference between the first and second blending: the first one uses 2 photos (closed and open), and the second one operates with 4 images, namely:

1. An open photo so that the value of n remains unchanged.
2. The image was sent by the interlocutor.
3. Auxiliary image provided by the interlocutor.
4. Our closed image.

The result of the first blending: the open image and the first user's closed image (Figs. 2 and 3, respectively) is shown in Fig. 6.

Figure 6: The result of the application when mixing Fig. 2 and Fig. 3

The result of blending this image with the second user's closed image is shown in Fig. 7.

Figure 7: The result of the application when mixing Fig. 6 and Fig. 4

The result of blending the open and closed images of the second user, shown in Fig. 2 and Fig. 4, respectively, is shown in Fig. 8.

Figure 8: The result of the application when mixing Fig. 2 and Fig. 4

The result of mixing Fig. 8 and Fig. 3. is the image shown in Fig. 9.

As you can see, Figs. 7 and 9 are identical, which proves that the Diffie-Hellman protocol was implemented correctly. Based on this fact, we can use the images for further encoding using the book method.

Figure 9: The result of the application when mixing Fig. 8 and Fig. 3

The encoding is implemented as follows: the function selects four random values and subtracts the fifth, taken as mod 256, from the sum of the corresponding pixel and the character value in the ASCII table. The first random value is the index of the photo, the next two are random values—x and y (pixel coordinates), the fourth indicates the corresponding base color and the fifth is an additional parameter. Since security is directly proportional to the generation of random numbers, we chose one of the generators with high entropy, namely MT19937-64 [14], with a reference to calendar time in seconds, thus adding the possibility of complicating the selection of the initial value. We also use an array to check whether this pixel has been used before. This check is necessary to ensure that the pixel is a one-time use because if the same pixel is used several times for the same letter, the attacker can calculate the difference in the fifth value and then understand the distance between the characters, which will facilitate decryption.

As an example, consider encrypting the following text: "*This_text is ex@mp!e?*". The program will split it into characters and encode each of them separately. An example of the finished encoding is shown in Fig. 10, and the decoded text on another device, which was implemented, is shown in Fig. 11.

If the decoded message is as originally set, it means that the message and photos have not been modified during transmission. We also make sure that the values are chosen randomly and do not repeat.

Let's consider the vulnerability analysis of the symmetric image-based encryption method, presenting the disadvantages and advantages.

The success of an attacker directly depends on the amount of knowledge about the system. Let's simulate the worst-case scenario where the attacker knows the public keys, the list of primes to be used, and the initial prime—that is, all the data that we will transmit over the unsecured channel. Based on this, we will calculate the speed of finding the encryption key—of course, this will depend on the direction of the attack.

```
0 32 141  0 109
0 39 274  0 89
0 5 249  0 100
0 158 28  0 200
0 104 122  0 100
0 99 168  1 138
0 106 209  2 10
0 119 160  2 36
0 92 240  0 111
0 74 243  2 32
0 108 184  1 128
0 69 213  1 206
0 131 50  2 33
0 10 225  2 196
0 70 206  1 25
0 110 279  2 15
0 8 207  0 137
0 73 152  1 203
0 151 200  1 97
0 110 276  0 107
0 68 36  1 104
0 128 167  1 75
```

Figure 10: The result of encoding the text "This_text is ex@mp!e?"

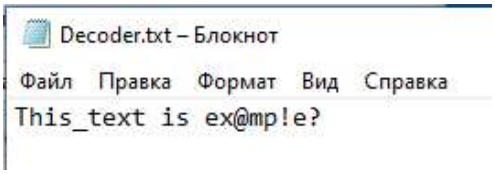

Figure 11: The result of decoding the message

The first direction is an attack on specific pixels, i.e. those that were used in the transmission of the message. In such an attack, the security of each pixel will directly depend on the Diffie-Hellman protocol. Unfortunately, it can be cracked using the following algorithm for finding the value of a closed pixel ():

1. Arrange the table of degrees from 1 to, at.
2. Finding by the formula.
3. Finding by the formula.

Whatever the length of the key used, the number of operations will remain equal to 3, except for the fact that there is a search by degrees. However, this will not be a big problem if the calculations are performed using a program with a given algorithm. The speed of

performing fairly simple operations is quite high, so the key will still be obtained in the end.

Of course, the time spent on the selection of the degree will be longer the larger the degree, and, according to this statement, the time for selecting the parameters will be longer, or there is a second solution—increasing the number of unknown operators by using double image mixing. That is, we will blend our image not once before sending it to our interlocutor, but twice (the number of blends can be more). Thus, for one pixel, the attacker will have neither the value of the private key nor the value of the first operation. If it is necessary to find a formula to reveal the desired value, the attacker will face the problem of finding the logarithm in the logarithm, provided that he does not know the intermediate results of the calculations that could make the task easier. Thus, it will be impossible to calculate the secret key. If you wish to increase the cryptographic strength of the encryption algorithm, the mixing procedure can be repeated more than twice, thereby increasing the number of unknowns in the final formula.

The second direction is to create a three-dimensional matrix that will store all possible outcomes of the operation. This table can also take a lot of time to create, which again depends on the size of the selected primes, but this situation is also possible to use.

To prevent the creation of this table, closed photos, along with open ones, should be changed regularly. The photo expiration time depends on the values of, and, i.e. on the values of the primes we use to mix the photos.

The blending time is a clear disadvantage of the system, as the program performs three operations for each pixel of the photo, so it takes a lot of time per pixel with small numbers. An example is shown in Table 1 and the graph in Fig. 12.

Since the program operates with very large numbers, of course, its speed will increase. To solve this problem, we used multithreading. Currently, the program uses only 8 threads, but if the number of threads is increased, the execution speed will increase accordingly.

Table 1
Time dependence on the value of the initial value

Initial value	First blending time of 100 pixels, s	Second blending time of 100 pixels, s	Total mixing time, s
1019	3.75	3.97	7.72
2143	10.17	11.52	21.69
3083	24.81	25.32	50.13
4159	32.23	35.24	67.47
5507	47.39	49.63	96.75
6991	59.03	62.41	121.44

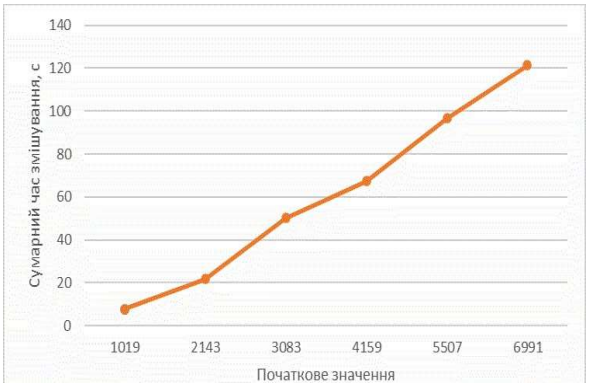

Figure 12: Graph of the dependence of the encoding time of 100 characters on the initial value

The second disadvantage is the vulnerability of the Diffie-Hellman method itself, as it is susceptible to man-in-the-middle attacks [15]. The solution to this problem can be the use of a read-only communication channel, i.e. without the possibility of modification, such as stickers in the Telegram messenger.

The third disadvantage is the use of pseudo-random numbers since it is impossible to achieve perfect entropy without using quantum computers, but we used one of the high entropy number generators, namely MT19937-64, and, according to the initial value, we chose a garter during the time, since the probability of selecting a specific second is very low.

The advantages of this method include:

1. The speed of its encoding, compared to the creation of a common image—encoding large text is very fast.
2. It is security against linear and differential cryptanalysis since the book cipher is resistant to them.
3. The possibility of using spaces and special characters is implemented, which is also a disadvantage of the book method.
4. Difficulty in selecting a private key.
5. The use of shorthand in transmission is because, for an ordinary user, photos will not carry any information.
6. The ability to create a key anywhere, as you just need to create an image.

4. Conclusions

This paper presents a symmetric encryption method that uses the Diffie-Hellman protocol and the book method. The Diffie-Hellman protocol is used to transfer keys between users. The paper proves its effectiveness for

this task and the possibility of its strengthening by repeated mixing. The book method is used to encode and decode the messages of the interlocutors using the generated joint image.

Also, in this paper, for the sake of demonstration, a software application was developed that implements the symmetric image-based message encryption method and demonstrates its effectiveness.

The disadvantages of the system, such as the speed of image mixing and the vulnerability of the Diffie-Hellman protocol, are listed, followed by the provision of solutions. Examples of possible attacks on this method are given and defense methods against them are proposed.

The advantages of this method are listed, which are associated with an increase in the level of security, encoding speed, and creation of a private and public key anywhere.

It should be noted that this method is not a panacea and has certain limitations in its application. It should also be noted that it can be used for malicious activities, such as crypto crackers, which is a clear disadvantage of any encryption method.

This research can also be used to implement information hiding in images using the solutions proposed by the authors in [16].

References

[1] P. Anakhov, et al., Increasing the Functional Network Stability in the Depression Zone of the Hydroelectric Power Station Reservoir, in: Emerging Technology Trends on the Smart Industry and the Internet of Things, vol. 3149 (2022) 169–176.

[2] V. Grechaninov, et al., Decentralized Access Demarcation System Construction in Situational Center Network, in: Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3188, no. 2 (2022) 197–206.

[3] M. Vladymyrenko, et al., Analysis of Implementation Results of the Distributed Access Control System. in: IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (2019) 39–44. doi: 10.1109/picst47496.2019.9061376.

[4] A. Carlsson, et al., Sustainability Re-search of the Secure Wireless Communication System with Channel Reservation, in: IEEE 15th Int. Conf. on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (2020) 973–977. doi:10.1109/tcset49122.2020.235583.

[5] V. Sokolov, P. Skladannyi, A. Platonenko, Video Channel Suppression Method of Unmanned Aerial Vehicles, in: IEEE 41st International Conference on Electronics and Nanotechnology (2022) 473–477. doi: 10.1109/elnano54667.2022.9927105.

[6] V. Svanadze, Doctoral Thesis "Cyber-security Policy and Strategy of Management," Georgian Technical University (2023). URL: https://riseaboveresearch.com/rise-above-research-estimates-that-1-13-trillion-photos-will-still-be-taken-in-2020/

[7] 2024 Worldwide Image Capture Forecast: 2023–2028. URL: https://riseaboveresearch.com/rar-reports/2024-worldwide-image-capture-forecast-2023-2028/

[8] D. Ristanovic, J. Protic, The Book Cipher Algorithm (2020). URL: https://www.researchgate.net/publication/293291732_The_book_cipher_algorithm

[9] Diffie-Hellman Key Exchange. URL: https://www.techtarget.com/searchsecurity/definition/Diffie-Hellman-key-exchange

[10] M. A. Murillo-Escobar, et al., RGB Image Encryption Algorithm Based on Total Plain Image Characteristics and Chaos, Signal Process. 109 (2015) 119–131. doi: 10.1016/j.sigpro.2014.10.033.

[11] A. Abusukhon, et al., A Hybrid Network Security Algorithm Based on Diffie Hellman and Text-to-Image Encryption Algorithm, J. Discret. Math. Sci. Cryptogr. 22(1) (2019) 65–81. doi: 10.1080/09720529.2019.1569821.

[12] E. Vidhya, R. Rathipriya, Key Generation for DNA Cryptography Using Genetic Operators and Diffie-Hellman Key Exchange Algorithm, Int. J. Math. Comput. Sci. 15(4) (2020) 1109–1115.

[13] Frequency Analysis. URL: https://www.101computing.net/frequency-analysis/

[14] F. Le Floc'h, Entropy of Mersenne-Twisters (2021). doi: 10.48550/arXiv.2101.11350.

[15] TLS Diffie-Hellman Key Exchange Logjam Vulnerability (CVE-2015-4000). URL: https://community.gigamon.com/gigamoncp/s/article/TLS-Diffie-Hellman-Key-Exchange-Logjam-Vulnerability-CVE-2015-4000

[16] S. Buchyk, et al., Applied Steganographic System for Hiding Textual Information on Audio Files, Emerging Networking in the Digital Transformation Age, TCSET 2022, LNEE 965 (2023) 317–334. doi: 10.1007/978-3-031-24963-1_18.