Journal of Algorithms and Computation

journal homepage: http://jac.ut.ac.ir

Comparing encryption algorithms and providing a new algorithm based on increasing data security

Fariborz Rasouli*[1]

[1]Faculty of Electricity, Computer and Advanced Technologies, Urmia University, Iran.

ABSTRACT

In this article, we examine different algorithms in the field of cryptography. And we analyze them from different aspects. Also, we introduce a new solution and algorithm, based on increasing data security. Our proposed method can perform the task optimally and efficiently compared to similar algorithms. In addition, it also reduces the need for storage space. Therefore, our proposed method works better and optimally by increasing productivity in operational environments. Our proposed algorithm can be used by increasing the security level for data, in various platforms and in all types of communication. Because the data is linked, and if we do not have the address of a data packet, it is not possible to decode the data. This increases data security.

ARTICLE INFO

Article history:
Research paper
Received 07, October 2022
Received in revised form 14, November 2022
Accepted 04, December 2022
Available online 30, December 2022

Keyword: Encryption algorithms, RSA algorithm, data encryption, security algorithms, Serial encryption.

AMS subject Classification: 05C78.

*Corresponding author: F. Rasouli. Email: f.rasouley@gmail.com

1 Introduction

Today, with the advancement of technology and the creation of new technologies, having security and privacy in line with the confidentiality of information is one of the requirements for communication with high reliability. For this purpose, providing a suitable platform for the confidentiality and security of users' data in insecure environments is of particular importance.
Generally, the use of cryptographic algorithms is used to protect the security of our data in various conditions and on various platforms of insecure spaces. In such a way that encryption algorithms have become an integral part of our communication. The use of these algorithms provides a suitable platform for transferring our information and virtual data in insecure spaces such as the Internet.
Protecting the information that is important to us and transferring it in various unsafe spaces can only be done by encrypting the information. Therefore, using modern encryption methods such as AES, DES & RSA will be useful for data security. In such a way that information is encrypted at the source using these methods and after transmission (in an insecure environment such as the Internet) is decrypted at the destination in order to prevent unauthorized people from accessing the information. Also, a hardware-software coding approach is presented to obtain resource-efficient implementations applicable to modern platforms such as the Internet of Things [7].
In this regard, protecting and maintaining the confidentiality of our private data, which is important to us, requires security measures. Security measures that are mostly done to protect against the threats of human factors in order to prevent the access of abusive and unauthorized people. Therefore, we have three types of security measures, including 1 - preventive measures, 2 - tracking unauthorized people in case they are discovered to prevent them from doing malicious acts again, 3 - reactive measures to minimize the impact of unauthorized people's access in the present and future. As seen in Figure 1.

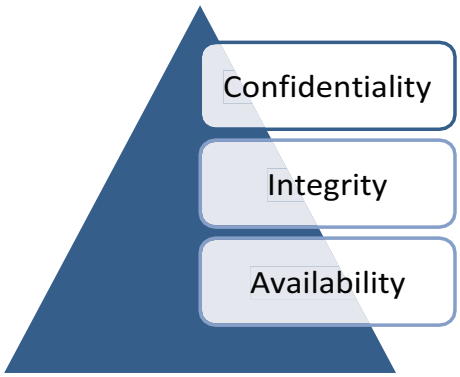


Figure 1: Three sides of the triangle are security measures to protect data in cryptographic algorithms.

2 Statement of issues and issues:

Based on this, in this article, we introduce algorithms for keeping information and data safe in anonymous spaces. Encryption in information is done to achieve data security in order to create confidentiality, integrity and authentication in data [8].
As mentioned, one of the solutions for the security of our information is the use of encryption algorithms. Meanwhile, encryption algorithms can be strengthened by using compression methods. Because compressed data is safer and easier to manage [7].
Also, fast and stable transfer with high reliability is one of the advantages of using compression method. But in general, we have two types of compression methods to transmit information, such as text, sound and image. Compression without quality loss and compression with quality loss. With loss and without loss [7].
In addition, we have two types of encryption algorithms. The first type is based on creating and having public keys where we have a key for encryption and decryption. Which is called symmetrical. In fact, a key is used for both encryption and decryption. The second type, which is called asymmetric method, has a public key and a private key for decryption. Also, the security of asymmetric methods is more, but the same method can work slower compared to the symmetric method.

3 Comparison of different algorithms in the field of cryptography:

In Table 1, we have compared cryptographic algorithms. And we describe the characteristics of five algorithms in the field of encryption for data. For example, the CBC encryption algorithm cannot be parallelized for encryption. Because in the algorithm, the output of the first sections will be needed in the next sections. But for decoding, it can be parallelized. Because the algorithm does not need a specific output to perform the task.

Parallelizable: means it can be encrypted simultaneously. And the result of the algorithm can be needed for the next part or not. (encoding or decoding).
Transmission error propagation: When one bit of data is corrupted somewhere for any reason, the rest of the data and information is no longer usable.
Fragment sending: In this method, the data are divided into specific fragments and then encrypted in the algorithm.
Streaming: In this method, we don't need to fragment the data. Rather, any data that reaches the encryption algorithm can be encrypted at the same time. As an example, we can take satellite communications as an example.
Also, the Data Encryption Standard (DES) encryption algorithm is one of the most prominent algorithms for data encryption. which is based on the Feistel cipher network. In addition, this algorithm works on bits. If the number of rounds in this algorithm is 15 or less, it can be decrypted by known plaintext attack method. Therefore,

	Algorithm	Parallelizable		Description
		Decryption	Encryption	
1	Electronic Code Book (ECB)	✓	✓	It is not safe, in the algorithm every data is always mapped to a ciphertext
2	Cipher Block Chaining (CBC)	✓	×	Fragment transmission, transmission error propagation, algorithm works based on authentication
3	cipher feedback (CFB)	✓	×	It has transmission error propagation, suitable for real-time applications, streaming
4	Output Feedback (OFB)	×	×	Suitable for real-time applications, streaming
5	Counter Mode (CTR)	✓	✓	Suitable for high-speed applications, it has fragment sending but can be converted to stream sending, efficient implementation and algorithm in hardware and software

Figure 1: Table 1. Comparison of encryption algorithms and their different aspects

it is better to have more rounds for this algorithm. And it can be vulnerable to comprehensive test attacks.
On the other hand, the Advanced Encryption Standard (AES) algorithm is one of the popular algorithms in the field of information encryption. Unlike the DES algorithm, the AES algorithm performs encryption operations on bytes. It also works based on the Substitution-permutation network (SPN).

4 Algorithm Rivest–Shamir–Adleman (RSA)

A cryptographic algorithm known by the name of its inventors. The RSA algorithm is one of the prominent and important algorithms for securing our information and data during transmission, especially in insecure spaces. Also, many scientists are working on making RSA more useful in IoT devices, smart gadgets, and lightweight devices [2]. In this algorithm, in the first step, we create a public key for the data. After that, encryption is done based on the special encryption function. And finally, after receiving the data at the destination, based on the special decryption function, the data is decrypted with the specified key.
Based on scaling exponentiation with large numbers, also based on the difficulty of factoring very large numbers into prime factors. In fact, it can be said that RSA security

basically relies on the practical difficulty of factoring the product of two large prime numbers [8]. Also, this algorithm is an asymmetric algorithm.
The RSA algorithm is described below. In this algorithm, based on the input data, the encryption key is created first, and then our data is encrypted with the corresponding function. Also, using the key, the encrypted data will be decrypted.
Among the applications of this encryption, we can mention things such as daily use in emails, vpns, global web browser programs, social network communication and anywhere that requires secure communication between the source and the destination.

Algorithm RSA

Input primes number p, q, r, s

Compute n = p*q*r*s, φ(n) = (p-1)*(q-1)*(r-1)*(s-1)

Input e 1<e< $\varphi(n)$, $gcd(e, \varphi(n)) = 1$

Compute e*d mod φ(n) = 1

Input massage = m

Compute C, C = m^e mod(n)

Compute m, m = c^d mod n)

End algorithm

For example:
q= 53 , p= 61
n= pq = 3233

Calculation of Euler's function:

$\varphi(n) = (p-1)(q-1)$
noindent $\varphi(3233) = 3120$

Choosing any number 1<e<3120 that is prime compared to 3120. We consider e = 17

Calculation of the inverse multiplicative co deposition d

$e(mod\varphi(n))$
d = 2753
e*d mod φ(n) = 1 thus 17*2753 mod 3120=1

noindent So, actually the public key is e=17 and n=3233 for message m. Also, the function for its password is given below:

C(m) = m $\wedge$17 mod 3233

In addition, the private key in the algorithm is d=2753. Therefore, the code function in the algorithm for decoding is as follows:

M(c)=c $\wedge$2753 mod 3233

Therefore, when we want to encrypt data using the function and key mentioned above in the RSA algorithm, it will be as follows (as an example of our data, m=65 is assumed):

Encryption: c= 65 ∧17 mod 3233 thus **2790**
Decryption: m= 2790 ∧2753 mod 3233 thus **65**

5 Our new proposed algorithm and solution to improve security:

New work: We propose a solution in the form of a new algorithm, which can make the implementation of cryptographic algorithms easier, and at the same time increase the security of our data several times. In fact, our proposed algorithm in this article can improve the security of encryption algorithms. We map data to special tables. For example, as can be seen in Figure 2. When data of any size are mapped to a specific table, data security will naturally increase and our data protection and privacy will be dramatically improved. Because it is impossible to distinguish and separate the data related to each data set.
Therefore, when this mapping is done, the abuser or the attacker cannot determine which set the corresponding data is related to. As can be seen in the figure, the X, y data are in both sets, but in the image and on the right side, it is not possible to determine which data set belongs to the left data set.
Therefore, when even data from the final mapped table falls into the hands of attackers, it will be worthless. Because it is not possible to recognize the original data from the written table. And this issue significantly increases the safety of our data over time.

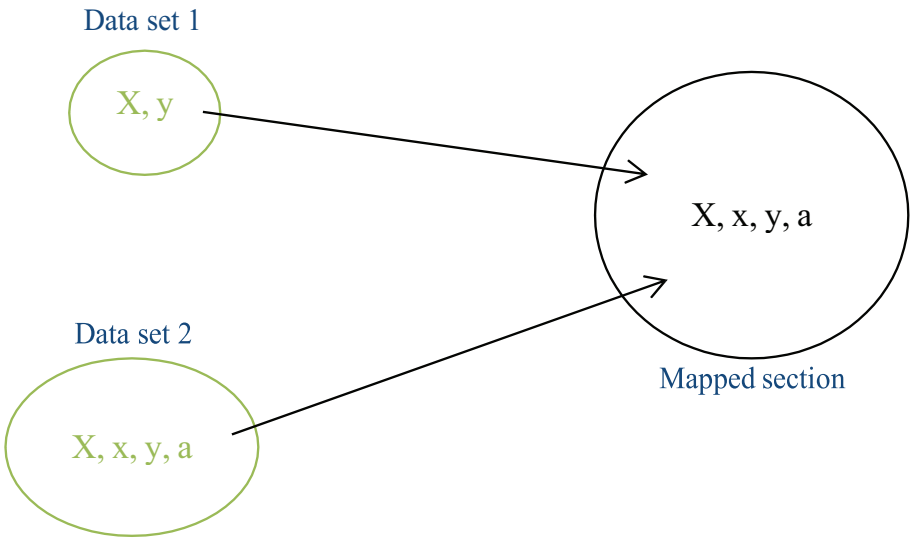


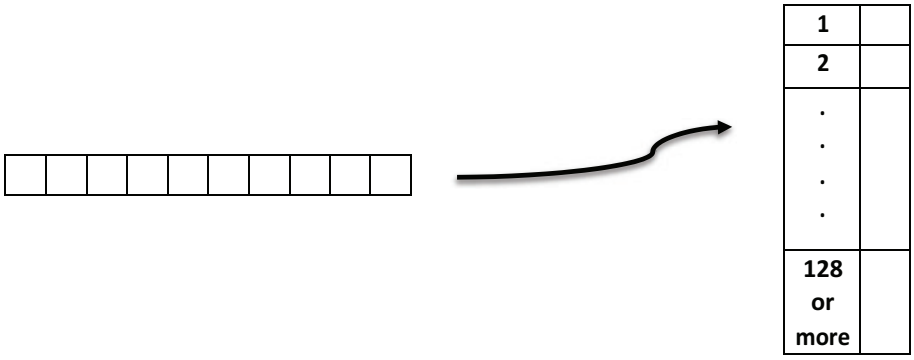
Figure 2: Master data mapping for specific locations, in order to improve data security. (Misleading by creating duplicate or worthless data

Therefore, when the data are mapped to special tables, the security and safety of the data increases. To this end, we introduce the following data mapping structure for a coordinated and integrated environment. In this structure, we map data of different size X to special tables. For example, when our data size (X) is less than 128 bits, it is mapped to a 128-bit table. Also, when our data are greater than the value of 128 but less than the value of 256, it is written to a table with the size of 256. And when the size of different data to be written is between 256 and 512, they are written to a table of size 512. Or data with a size larger than 1024 should be divided into parts and sizes of 1024.
So, for example, when we have data of size 32, it will be written to a table of size 128. And also, when we have data with the size of 128 or 64, according to the structure, it can be written to the 128 tables. Also, empty places in tables for smaller data can be filled with secondary and worthless data. By doing this, data can be sent encrypted, with higher security, from origin to destination.

$$X < 128$$
$$128 < X < 256$$
$$256 < X < 512$$
$$X > 1024$$

On the other hand, security can be improved by hashing data. For this purpose, when we hash the data into special tables, we will face a lot of unclear and irrelevant data. This can increase data security.
In this regard, the following relationship can be used to hash the tables. where k is the key and n is the number of data.

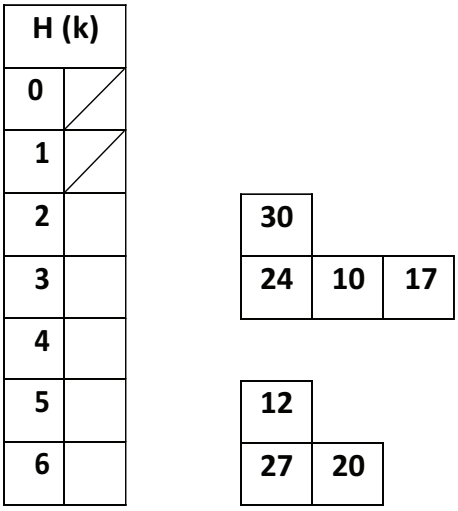
$$H(k) = k \bmod n$$



For example: we consider the following data to be mapped to a table.

The table mapped for it will be as follows. For example, the address of key 17 is mapped in house three. After that address 20 is mapped to house 6 and in the same way other data can be mapped using the relationship.

K	17	20	12	10	30	24	27
H (k)	**3**	**6**	**5**	**3**	**2**	**3**	6



6 Description and definition of the problem and expression of the new algorithm:

Following the solution of security measures and algorithms for information protection, we introduce trail data. As can be seen in Figure 3. Serial data have the advantage of significantly increasing information security. And they have the ability to hide and protect the main data among secondary or worthless data.
In the explanation of sequential data, it can be said that these data are related to each other in the form of a sequence of data related to each other, and at the same time with very high security. In this way, when we have an initial vertex of the data in a table, we can decode the data in a certain order. In such a way, if we have other vertices besides the initial vertex of the data, it has no value and credibility for us, and the data cannot be decoded. Therefore, we require receiving the address of the first data from the mapped data string, along with the associated value to decode the first packet from the origin.
For example, suppose we have the initial vertex of a data set in a particular table. (tables described in Figure 2). Therefore, when we have the initial vertex of a string of data mapped in a table, the next data can be extracted from it. Thus, suppose we have an initial vertex and an associated numerical value with the data packet set 1. Inside the package 1, the numerical value we have is added to the value of X inside the package. And if the sum of this calculation is correct, the data of package 1 can be extracted. And in the next step, we get a numerical value after reading the data. This value is needed to obtain the address and data of the next packet. And in the last step of the packet one operation, it gives us the data address of the next packet after reading the data contained in packet 1, packet 1. Along with a numeric value to decode the next packet.

X is a value inside the package. And it is collected with a numerical value in our possession so that the data can be decoded.
So, if we don't have the starting packet and its value to compute X, we can't get any data from the associated data strings.

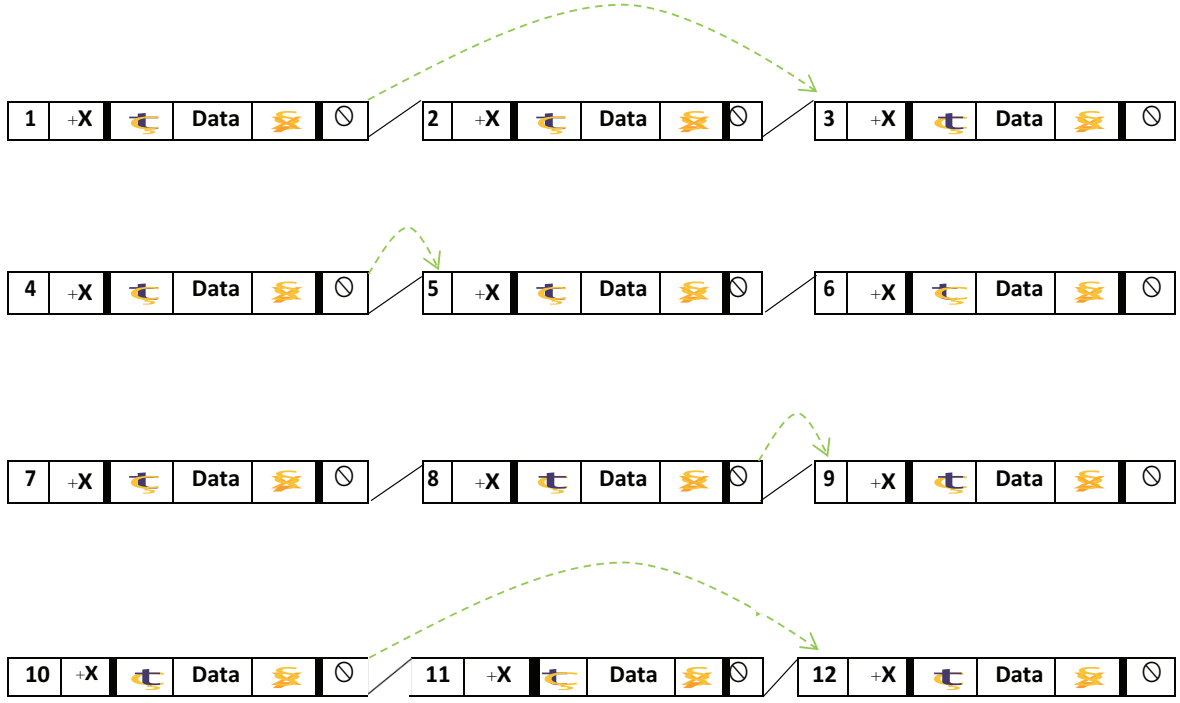


Figure 3: Data decoding process in the new proposed algorithm

Thus, as seen in Figure 3, we first decode data packet 1. After that, packet 1 directs us to packet 3 by providing the address and the corresponding numerical value. In the next step of data decoding, data packet 3 guides us to the address of packet 4. Then in the same way, we will decode packets 5, 8, 9, 10 and finally the packet containing data 12.
It is clear that the data cannot be decrypted without having the initial data packet in hand.
Also, we have sketched the sequential data algorithm below, for better representation. In this algorithm, different steps for decoding data after mapping to tables are described. And this algorithm can be used to extract data.

Algorithm: Sequential and linked data algorithm with high security

First find the starting data address

Receive a numerical value to decode data packet one

Calculate the result of the numerical value with the value of x inside the package

If the answer to the equation was correct:
Decrypt the data packet

If the answer is not correct:

An error will be encountered and the package will not be decoded

After successful decryption of packet data

The address and numerical value of the next packet should be received to decode the next packet

End algorithm

7 Conclusion

In this article, we examined and analyzed the types of encryption algorithms in the field of data protection. Among other things, we examined the RSA algorithm, which works by creating a key and creating encryption and decryption functions. Also, in this article, we introduced a new solution called sequential data. That these data are mapped to special tables, to increase the security of our data even more. Sequential data in a table is worthless and indecipherable until we have the starting address of a string of data. As a result of this issue, it causes a significant increase in the security of our data and information. Our algorithm protects the data packets in the best way. And it has the possibility to be used in different platforms, in all communication and different conditions, to have higher security. On the other hand, our algorithm reduces costs and increases productivity due to the use of less storage space than other algorithms. It is obvious that our proposed algorithm performs the task optimally and efficiently. And it can bring us an increase in speed in operational environments.

References

[1] Ahmad, S., Alam, K. M. R., Rahman, H., and Tamura, S. (2015). A comparison between symmetric and asymmetric key encryption algorithm based decryption mixnets. Proceedings of 2015 International Conference on Networking Systems and Security, NSysS 2015. https://doi.org/10.1109/NSysS.2015.7043532

[2] Bai, K., and Wu, C. (2016). An AES-like cipher and its white-box implementation. Computer Journal, 59(7), 1054–1065. https://doi.org/10.1093/comjnl/bxv119

[3] Imam, R., Areeb, Q. M., Alturki, A., and Anwer, F. (2021). Systematic and Critical Review of RSA Based Public Key Cryptographic Schemes: Past and Present Status. IEEE Access, 9, 155949–155976. https://doi.org/10.1109/ACCESS.2021.3129224

[4] Lee, S., Cho, S. M., Kim, H., and Hong, S. (2019). A Practical Collision-Based Power Analysis on RSA Prime Generation and Its Countermeasure. IEEE Access, 7, 47582–47592. https://doi.org/10.1109/ACCESS.2019.2909113

[5] Li, N., Han, Q., Zhang, Y., Li, C., He, Y., Liu, H., and Mao, Z. (2022). Standardization Workflow Technology of Software Testing Processes and Its Application to SRGM on RSA Timing Attack Tasks. IEEE Access, 10(July). https://doi.org/10.1109/ACCESS.2022.3196934

[6] Murtaza, A., Hussain Pirzada, S. J., and Jianwei, L. (2019). A new symmetric key encryption algorithm with higher performance. 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies, ICoMET 2019, 1–7. https://doi.org/10.1109/ICOMET.2019.8673469

[7] Mustafa, I., Khan, I. U., Aslam, S., Sajid, A., Mohsin, S. M., Awais, M., and Qureshi, M. B. (2020). A Lightweight Post-Quantum Lattice-Based RSA for Secure Communications. IEEE Access, 8, 99273–99285. https://doi.org/10.1109/ACCESS.2020.2995801

[8] Nitaj, A., Ariffin, M. R. B. K., Adenan, N. N. H., Lau, T. S. C., and Chen, J. (2022). Security Issues of Novel RSA Variant. IEEE Access, 10, 53788–53796. https://doi.org/10.1109/ACCESS.2022.3175519

[9] Wahab, O. F. A., Khalaf, A. A. M., Hussein, A. I., and Hamed, H. F. A. (2021). Hiding data using efficient combination of RSA cryptography, and compression steganography techniques. IEEE Access, 9, 31805–31815. https://doi.org/10.1109/ACCESS.2021.3060317

[10] Yihan, W., and Yongzhen, L. (2021). Improved Design of des Algorithm Based on Symmetric Encryption Algorithm. Proceedings of 2021 IEEE International Conference on Power Electronics, Computer Applications, ICPECA 2021, 220–223. https://doi.org/10.1109/ICPECA51329.2021.9362619

[11] Yu, L., Zhang, D., Wu, L., Xie, S., Su, D., and Wang, X. (2018). AES Design Improvements Towards Information Security Considering Scan Attack. Proceedings - 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications and 12th IEEE International Conference on Big Data Science and Engineering, Trustcom/BigDataSE 2018, 322–326. https://doi.org/10.1109/TrustCom/BigDataSE.2018.00056