Journal of
Data Acquisition and Processing

ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777182

BREAKING THE MOLD: THE RAILVE ALGORITHM OUTPERFORMS TRADITIONAL CRYPTOGRAPHIC APPROACHES

Adnan Adel Bitar
Research Scholar, Computer Science, CMS College of Science and Commerce, Bharathiar University, Coimbatore, India, Adnan.bitar.4m@gmail.com

Dr. V. Sujatha
Associate Professor, Computer Application, CMS College of Science and Commerce, Bharathiar University, Coimbatore, India, sujathapadmakumar4@gmail.com

ABSTRACT -- Transmitting data from a source (transmitter) to a destination (receiver) requires security procedures and methods to be protected. And it is done by using symmetric or asymmetric keys. There are many important factors in ciphering and transmitting procedures; such as "Space-Time Trade Off" that shows the relation between the space and the time of each algorithm; the speed of transmission depends on it, "Cipher Text Size", and the accuracy of encryption algorithm. These are considered some of the most important parameters in converting a plain text to a cipher text. Though, the speed of transmitting data is affected by how big the message is that wanted to be sent. In this paper, the parameters will be compared through some of the most famous algorithms to show how effective is Railve algorithm compared to others.

Keywords -- Security, Railve, AES, encryption, The IoT.

I. INTRODUCTION

Securing data is the main concern to all the people in the different departments throughout the world. Encryption procedure starts by taking the plain text that the sender is willing to send and applying a key on it to transform it to unintelligible text "cipher text". Encrypted text has size as big as the plain text or maybe much bigger. However, the size takes part in how fast is the transmission of the data from one device to another, and that depends on how big is the cipher text size.

In this paper, three of the well-known encryption algorithms "AES", "Twofish", "CHACHA20" as well as the presented algorithm "Railve" by [1] are applied on a device with a range of different files sizes from (1MB to 128MB). All the algorithms are compared in three parameters: Space-Time Trade Off, Cipher Text Size, and Accuracy. The comparative study is done to prove the efficiency of Railve algorithm compared to some encryption algorithms.

Railve algorithm is a stream cipher algorithm that has the same way of encryption as "CHACHA20". Not like "AES" and "Twofish " as they are block cipher algorithms. Nevertheless, all the algorithms are under symmetric algorithms. Symmetric algorithms use one key to perform the encryption and decryption of the texts. For example: AES, 3DES, Twofish, RC4, and Railve algorithms. [1][2] While, the Asymmetric algorithms use two different keys: one for encryption (public) and one for decryption (private). For example: RSA algorithm. [2]

II. IMPLEMENTED ALGORITHMS

A. Block Cipher:

- AES (Advanced Encryption Standard)

AES is an encryption algorithm produced by “National Institute of Standards and Technology” in 2001. The AES algorithm has come to replace DES to overcome the shortcoming. AES is a block cipher, and that means a whole block of 128 bits is processed at a time [3]. And the key lengths used for it are 128, 192, or 256 relating to the number of rounds of encryption that AES includes. [3][4]

- Twofish

Twofish is a symmetric encryption that has the same block size and key sizes of AES. Features of Twofish encryption are: use of S-boxes and key schedule. Twofish has a Feistel structure like DES. [3]

B. Stream Cipher:

- CHACHA20

ChaCha20 was published in 2008 as a type of Salsa20. ChaCha20 replaces a round of function of Salsa20 and allows more updates in a single cycle. ChaCha20 is faster than AES in the implementation of software only. [6]

- Railve

Railve algorithm is an encryption algorithm that is combining two encryption algorithms “Vernam” and “RailFence” to make use of their specification. [1]

III. EXPERIMENTAL SETUP

In order to do the comparison of the three parameters for all the mentioned algorithms, the implementation is done using Python language using “Spyder (Anaconda3)” on two laptops. Table 1 describes the specification of the hardware devices:

Table 1
Hardware specifications

Laptop (HP 350 G1)	
Processor	Intel Core i7 - 5500U CPU (2.40 GHz)
RAM	8 GB

IV. REVIEW OF LITERATURE

It is illustrated in [1], how the authors merged two simple encryption algorithms to present the “Railve” algorithm. When algorithms were applied on The IoT devices, Railve algorithm showed a great result comparing to the base algorithms.

The authors have explained about symmetric and asymmetric encryption algorithms in [1], [2]. In [3], AES encryption algorithm is explained exhaustively and its performance is tested and mentioned in detail with other encryption algorithms. And all the implemented algorithms are executed on Intel Core i7 – 3700 CPU.

Analysis of various execution techniques of AES encryption algorithm for better performance to face the different challenges of security of The IoT based systems. The analysis is stated in [4].
Consideration of this matter, the authors in [7] and [8] have shown an excellent implementation of different encryption algorithms like AES on The IoT devices.

In [9], the design and the implementation of Twofish algorithm are explained in brief.

ChaCha20 has been illustrated deeply in [6]; their paper is a guide for implementing the ChaCha20 algorithm and Poly1305 separated and combined.

Super Chacha Lightweight is presented by the authors in [10] which consists of 10 rounds of encryption to complete the execution.

V. PROPOSED WORK

In this paper, the most famous algorithms are compared to the produced algorithm “Railve” to point out the differences among them in the three parameters:
Firstly, the execution time: how fast does each algorithm take to produce the cipher text.
Secondly, cipher text size: measure of how much data has been produced after being encrypted.
Finally, time space trade off: means how accurate is the text before encryption and after decryption.
The input for all the algorithms is a random string ranging from (1MB to 128MB) generated by the laptops in addition to the used random key for encryption (256 bits).
The algorithms are applied on the IoT device. Each algorithm is executed ten times for each size of the file repeatedly to get the average of the result to show the important parameters that resulted from the execution. Starting with Railve algorithm there are two keys one size of 256 bits and it is an automatic generated random key and the second key is a number from 2 to 9 selected randomly. On the other hand, the other three algorithms are applied with random keys of 256 bits, thus making the brute force attack impossible.
The parameter “time-space trade-off” is measured by the equation:
Time-space trade-off = Time required / Space required
Whereas the "Time required" is the amount of time required to perform a task and the "Space required" is the amount of memory or storage required to perform that task. This ratio can be used to compare different algorithms or systems and to determine which is more efficient in terms of time and space.

VI. EXPERIMENTAL RESULTS

Firstly, the algorithms are executed on the laptop and the results of the implementation are:
Table 2
(Comparing execution time among the three encryption algorithms on the laptop)
File Size (MB)Executing Time (Sec)

Table 2

(Comparing execution time among the three encryption algorithms on the laptop)

File Size (MB)	Executing Time (Sec)			
	CHACHA2	AES	Twofish	Railve
1	1.33	1.4	1.4	1.3
2	2.36	2.3	2.7	3.4
4	4.47	4.4	4.3	5.4
8	8.72	9.2	8.5	13
16	19.3	19	17	23
32	33.9	36	34	42
64	71.2	74	73	82
128	150	151	148	160

The results in “Table 2” are showing that the Railve algorithm is taking more time compared to other algorithms and the differences range depending on the size of the files.

In every encryption algorithm, the size of the plain text matters but the produced cipher text is playing more important role. Here in the following table 3, the encryption algorithms are applied with the same plain text size and compared by the size of the cipher text.

Table 3

(Message sizes before and after the execution)

Plain File Size (MB)	Cipher Text Size (MB)			
	CHACHA2	AES	Twofish	Railve
1	2	2	2	1
2	4	4	4	2
4	8	8	8	4
8	16	16	16	8
16	32	32	32	16
32	64	64	64	32
64	128	128	128	64
128	256	256	256	128

So, whatever plain text size enters for implementation in Railve algorithm; the input equals the output, which means the plain text size equals the cipher text size Whereas, the other algorithms’ cipher texts are approximately double the size of the plaint text.

From the above tables 2,3 the following chart of “space time trade off” is produced as follows:

Diagram 4 shows how the Railve algorithm is better compared to the rest.

VII. CONCLUSION

To conclude, through this research work, the four algorithms are executed on the IoT device to show how strong the Railve algorithm is. The results have shown that the Railve algorithm is outperforming other algorithms in a way that it is taking more time but better file sizes and that plays an important role of transmitting less data. When the size of the message is small that means faster the transfer from one device to another.

However, after encryption tests, all the encryption algorithms are accurate of 100 percentage including the Railve algorithm.

As future enhancement for the Railve algorithm, execution speed could be improved to overwhelm other algorithms in all aspects.

VIII. REFERENCES

[1] Bitar. A.A, and Sujatha. V, "Merging Vernam Cipher stream and Rail Fence Algorithms and How Effective They are on Internet of Things Devices", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), May-June, 2021, Vol 7, Issue 3, pp.686-691.

[2] Verma. O.P, Agarwal. R, Dafouti. D, and Tyagi. S, “Notice of Violation of IEEE Publication Principles: Performance analysis of data encryption algorithms”. In 2011 3rd International Conference on Electronics Computer Technology, April, 2011, Vol. 5, pp. 399-403.

[3] Panda. M, “Performance analysis of encryption algorithms for security”. International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), 2016, pp. 278-284.

[4] Farooq. U, and Aslam. M.F, “Comparative analysis of different AES implementation techniques for efficient resource usage and better performance of an FPGA”. Journal of King Saud University-Computer and Information Sciences, 2016, October, 29(3), pp.295-302., 295–302

[5] Ferguson. N, Kelsey. J, Lucks. S, Schneier. B, Stay. M, Wagner. D, and Whiting. D, “Improved cryptanalysis of Rijndael. In International Workshop on Fast Software Encryption”, April, 2000, pp. 213-230, Springer, Berlin, Heidelberg.

[6] Nir. Y, and Langley. A, “ChaCha20 and Poly1305 for IETF Protocols”. Internet Engineering Task Force, 2015.

[7] Kulkarni. S, Durg. S, and Iyer. N, “Internet of things (IoT) security”.In 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), March, 2016, pp. 821-824.

[8] Suo, H., Wan, J., Zou, C., Liu, J., 2012. “Security in the internet of things: A review”. International Conference on Computer Science and Electronics Engineering, vol. 3. pp. 648–651.

[9] Schneier. B, “The Twofish encryption algorithm”. Dr. Dobb's Journal: Software Tools for the Professional Programmer, 1998, 23(12), pp.30-34.

[10] Mahdi. M.S, Hassan. N.F, and Abdul-Majeed. G.H, “An improved Chacha algorithm for securing data on IoT devices”, SN Applied Sciences, 2021, 3(4), pp.1-9.