*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
*DOI: https://doi.org/10.17762/ijritcc.v11i10.8485*
***Article Received:** 28 July 2023 **Revised:** 15 September 2023 **Accepted:** 04 October 2023*

# New Era of Micro Cipher Algorithm (MCA) : Cipher Text for any Length of Plaintext

**Ummadi Thirupalu[1], A. Rajasekhar Reddy[2], P. Chandrakanth[3]**
[1]Computer Science and Engineering
ASCET,
Gudur,India
ummadi70@gmail.com
[2]Computer Science and Engineering
NBKR Institute of Science and Technology,
[2]rajareddy4u@rediffmail.com
[3]Computer Science and Engineering
NBKR Institute of Science and Technology,
[3]chandrakanthc4u@gmail.com

**Abstract**— Cipher text is text which is in an unreadable format. It is the text which is transmitted between the people as an exchange of secret messages. The secret message is also called cipher text. In General, the existing algorithm-generated cipher text size is equal to or greater than the actual plaintext. In a cryptosystem, there is a chance to reduce the cipher text. We introduce a new algorithm called "Micro Cipher Algorithm (MCA)" in this connection. It is an algorithm that produces a little bit of cipher text for any length of the plaintext. This feature achieves many things like unbreakable code, transmitting data through different kinds of networks and machines very easily and quickly (simply throughput) and storage of cipher at different machines as a minute. It also reduces congestion and increases throughput in a networking. throughput in networking.

**Keywords**- Plaintext, Cipher text, Drop Dupe Characters (DDC), Drop Dupe Digits (3D), Key Generation, Code Generation.

## I. INTRODUCTION

This document is a template. An electronic copy can be downloaded from the conference website. For questions on paper guidelines, Cryptography is the mathematical science which is used to encrypt and decrypt data. To encrypt and decrypt data, we use different kinds of algorithms. There are several algorithms [1] [8] to do this task. Generally, existing algorithms which took the plaintext as input and produce the output with same length or greater the length. So that, the cipher text which is generated by the existing algorithms is took time to generate cipher text, transmitting the cipher or storing the cipher especially for cloud environment is a cost effective way [3][6]. In this scenario, we concentrate to reduce the size of cipher text compared to the plaintext. There is a facility to reduce the plaintext before generating cipher text. Usually, in a text entry, the letters of alphabet are repeatedly entered for a line of sentence. So that the plaintext which is entered by us should contains duplicate characters because in a text entry the 26 letters (alphabets) only repeats. These duplicates characters are higher in presence, if the text size is large. In this paper, we drop these repeated character before generate cipher text. After reducing the plaintext, the cipher text which is generated by the algorithm is lesser that the actual plain text. If the cipher text size is less, it took less time, high transmitting rate, and more advantage for cloud environment and also it is a notorious cipher for the cryptanalysis[7][9]. To meet this demand we introduce the Micro Cipher Algorithm (MCA). Through this algorithm a micro cipher is generated, and it occupies less storage, high transmitting rage and also a great advantage for the cloud environment organizers.

## II. METHODOLOGY

All Methodology is a system of methods, which are used in a particular area of study or activity. In this paper, we introduce different kinds of techniques which are new kinds of techniques. They are:

### *A. Micro Cipher Algorithm (MCA)*

Micro Cipher Algorithm is an algorithm which generates minimized cipher text. This is the tiny cipher than exiting cipher like DES [2][4][14], AES [13], RC [5][10] family etc. The key process of Micro Cipher Algorithm is to reduce any length of plaintext into small size text by dropping the duplicate character and digits at different levels.

At first level, it produce the little bit of plaintext by dropping duplicate characters from the original plaintext

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
*DOI: https://doi.org/10.17762/ijritcc.v11i10.8485*
***Article Received:** 28 July 2023 **Revised:** 15 September 2023 **Accepted:** 04 October 2023*

(performing Drop Dupe Characters). The text which is produced from the first level process is submitted as input to the second level process[11][12]. At second level, it produce the tiny plaintext as a micro text by generating and dropping (Drop-Dupe-Digits (DDD)) from the first level DDC PT. This DDD (3D) plaintext is act as Micro plaintext to generate cipher text. Simply it produces two digits length of micro plain text at the maximum length of characters (bytes) as an input [15]. Off-course it generates at most micro characters[16] for any length of characters (Original plaintext). For example, the plaintext "Rivers never goes reverse." text is given as an input to produce micro cipher like the following.

The operations, which are used to generate Micro Cipher is as follows.

- Sort Text(ST)
- Drop Dupe Characters (DDC)
- Generating Bits (GB)
- Sort Bits (SB)
- Drop Dupe Digits (3D)
- Key Generation (KG)
- Code Generation (CG).

Plaintext: Plaintext is a text which is enter by the user for converting it into scrambled text. For instance, the text "Rivers never go reverse." is used as plaintext

Rivers never go reverse.

Figure.1 Plaintext

Sort Text (ST): It is the process of sorting the given plaintext to drop the duplicate text easily at next level. Before sorting the text, we keep the index position of the plain text to generate Index-1 which act as one of the sub-key of the core key.

Drop Dupe Characters (DDC): It is the process of dropping of duplicate character [ ] from the given plain text. In the above plain text contains duplicate characters. Characters are dropped with this method. After dropping the duplicated characters, we get the DDC text like the following.

3.Re6ginor4s2v3

Figure 2 Drop Dupe Characters (DDC)

Generating Bits (GB): Generate bits is the process of generating bits for every bytes of DDC Plaintext. These bits are usually contains two type of digits only that are 0s and 1s. These set of bits are passed for next phrase for further processes.

Sort Bits (SB): Sort Bits is the process of arranging the bits into an order. Here the 0s and 1s of all bytes of DDC PT into an order, usually ascending order. This king of order helps the next section to drop the digits for the MCA. Here too find out Index of the bits before dropping the bits. This index is act as Index-2. It is another sub-key to be made core-key for this algorithm. The generated bits are the above example is as follows.

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Figure 3 Drop Dupe Digits

These repeated bits are dropped in the following section.

Drop Dupe Digits (3D): Like Drop Dupe Characters, the digits are dropped. In the above generated bits, there are only two kinds of digits: 0s and 1s. These digits are counted and dropped because there are lot of repeating of digits. Finally, we get the counted numbers for 0s and 1s. The Drop Dupe Digits (3D) code for the said above example is as follows

0s = 64

1s = 64

Figure 4 . Core-Key

'Sai' is 3 1 2. These key values 3 and 1 (usually, sequence of character entry) only used for the above data for generating the code.

To made Core-Key as a symmetric key for this algorithm, we combined all these three (Index-1, Index-2, and Entry key) key values as key. The following image shows the Core-Key process.

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
*DOI: https://doi.org/10.17762/ijritcc.v11i10.8485*
***Article Received:** 28 July 2023 **Revised:** 15 September 2023 **Accepted:** 04 October 2023*

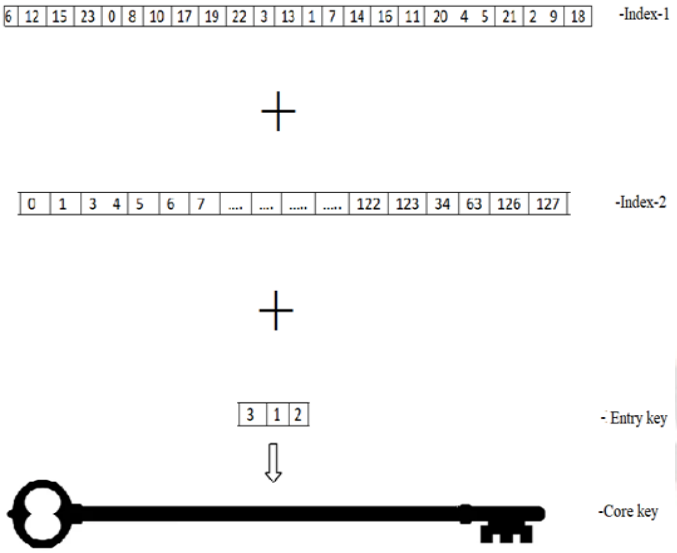


Figure 5. Core-Key process

To made Core-Key as a symmetric key for this algorithm, we combined all these three (Index-1, Index-2, and Entry key) key values as key. The following image shows the Core-Key process.

*Code Generation (CG):* Code Generation is the process of generating the related code for the above generated code. Here, first we perform XOR operation on the two kinds of data with the key position values. And finally, convert the data into character to generate the code. This code is act as cipher code (cipher text). Simply, this code is appeared as two character code for any length of string of characters. So that the cipher text is generated by this algorithm is two characters code only.

It is one of the unique features in cryptography till today. And it appears as micro cipher for any length of string of characters. The code for the above plaintext is as follows.

**CA**

Figure 6. String of characters

The pictorial representation of the MCA is a follows

## III . PROPOSED ALGORITHMS:

In our proposed technique, we introduce set of sub-algorithms as an algorithm. All the sub-algorithms are called in a main algorithm to done the job perfectly.

a. ALGORITHM: MCA

---

INPUT: PLAIN TEXT, MAIN KEY.

Output: Cipher text, Core key.

---

Steps:

1. N ← size of (Plain text)
2. C[ ] ← Plain text // C is a Character Array to store plaintext characters
3. K[ ] ← SortText (C) // Array is passed to function and return Index-1 to K array
4. Str ← DropDupeCharacters(C) // Drop the duplicate Character from the plain text
5. Bg[ ] ← BitGeneration(Str) // Generate bits of the String Str
6. KN[ ] ← SortBits(Bg) // Array is passed to get Index-2 as well as arrange the bits into an order
7. N[ ] ← DropDupeDigits(Bg) // Drop the digits and count the number to drop.
8. CT ← N[ ] ⊕ Core Key. // CT is a variable which stores Cipher text.
9. END.

---

b. Algorithm – 3.2 : SortCipher(C[ ])

The proposed algorithm SortCipher is used to sort the input text. This is one of the technique which is used to arrange the characters into an order to drop easily. Through this algorithm, not only arrange the text into order, we may get index of the each character. This index is act as one of the partial key for the destiny.

---

Input: C[ ] // Array of characters

Output: Sorted Text, Index-1.

---

Steps:

1. n ← C. length
2. For ← 1 to n
3. K[i] ← i // K is an array to store index of the input text
4. For -← 1 to n-1
5. P ← i
6. For j ← i+1 to n
7. if C[p] < C[j] then
8. p ← j
9. End for
10. Swap K[p] ↔ K[i]
11. Swap C[p] ↔C[i]
12. End for
13. Return K
14. End

---

c. Algorithm – 3.3 **:** Drop Dupe Characters(C)

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
*DOI: https://doi.org/10.17762/ijritcc.v11i10.8485*
***Article Received:*** *28 July 2023* ***Revised:*** *15 September 2023* ***Accepted:*** *04 October 2023*

The proposed algorithm Drop Dupe Characters is used to drop the duplicate characters from the sorted plaintext. In this process, the dropped characters should be counted and added it to the generated output string of text. It is one of the core process to reduce the text.

---

Input: C[ ] // Array of characters
Output: RT //Reduced plaintext

---

Steps:

1. n ← C. length
2. k ← 1
3. for i ← 1 to n
4. p ← 0
5. if C[i+1]==C[i] then
6. for j ← i +1 to n
7. if C[i]==C[j] then
8. p ← p + 1
9. Continue
10. Else
11. EXIT
12. End if
13. End for
14. RT[k]=C[i] //RT is an array to store only the reduced characters
15. k ← k + 1
16. RT[k] ← p
17. Else
18. RT[k] ← C[i]
19. End if
20. End for
21. Return RT
22. End

d. Algorithm – 3.8: Decryption

The decryption algorithm DECTRYPTION is used to decrypt the data. Through this major algorithm, we call sub-algorithms which are related to the decryption. Simply it is the reverse processing algorithm for encryption

---

Input: CT //Cipher text
Output: PT //Plain text

---

Steps:

1. Split CT // split cipher text into two characters.
2. Split Core Key into three parts: 1. Index-1, 2. Index-2, 3. Key
3. CCT[0] ← C //C represents zero counted number character
4. CCT[1] ← T //T represented one counted number character
5. MC ← DXOR(CCT,Key) // performing XOR operation on cipher text
6. PN ← PickNums(MC) //Pick numbers from micro characters (MC)
7. US ← UnSortNums(PN,Index-2) //unsort the number using Index-2
8. C ← GetCharacters(PN) //get characters related to the bits of numbers
9. PD ← PickDupeCharacters(C) //pick dropped characters which were dropped
10. PT ← UnSortCharacters(PD, Index-1) // unsort the characters using Index-1 to //get the original string of characters
11. Print PT //PT is the plain text
12. End

---

The following frame work shown the entire process for the cipher text generation as well as key generation.

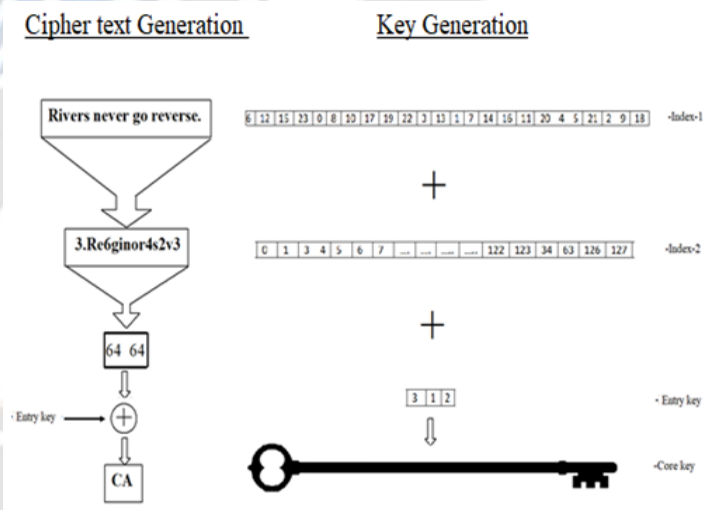


Figure 7. Cipher text generation

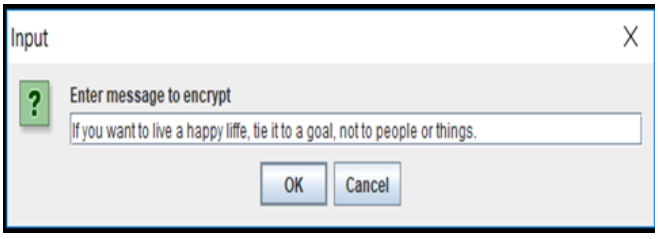


Figure 8. Input: *Plaintext*

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
*DOI: https://doi.org/10.17762/ijritcc.v11i10.8485*
*Article Received: 28 July 2023 Revised: 15 September 2023 Accepted: 04 October 2023*

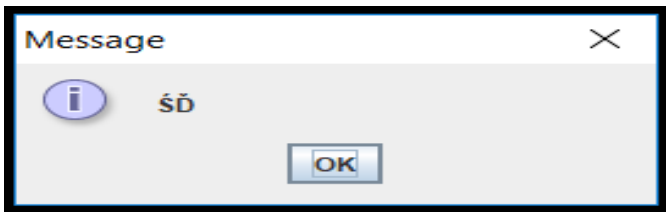


*Figure 9 .* INPUT Key

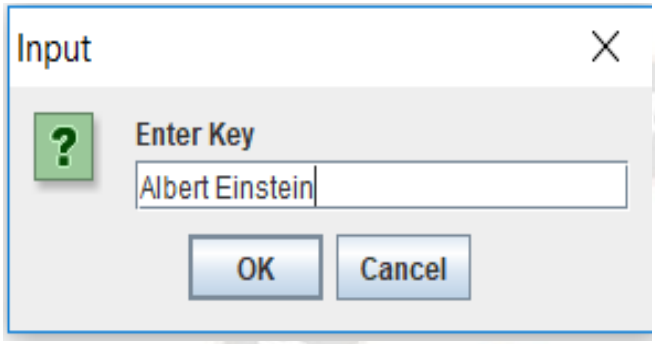


*Figure 10. OUTPUT:CIPHER TEXT*

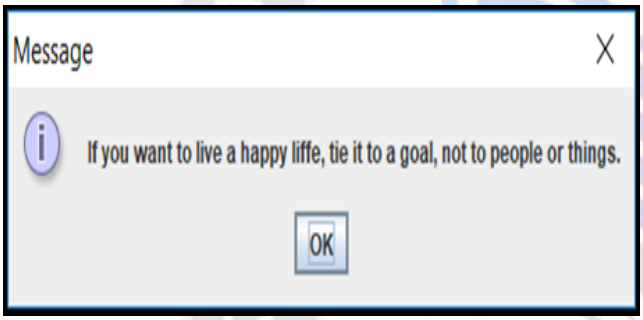


***FIGURE . OUTPUT:* PLAINTEXT**

## IV . CONCLUSION

Micro cipher generation is a unique process. In this process, the index positions of both plaintext and generated code plays an important role. These two index values are preserve and merge to make a large key (index1 + index2 + key). At destination end, split these values to decrypt the text.

It is a revolutionary technique in the computing power, throughput, cloud computing etc. Especially, it is most notorious cipher text for the Crackers. It is 100 per cent unbreakable code for the cryptanalysis because the key and cipher text not at all matched. So that this a most trustable and secure code for the ciphers.

If the input text is larger, split the input text into multiple part then perform the operation through the proposed system.

## REFERENCES

[1] Yahia Alemami, Mohamad Afendee Mohamed, Saleh Atiewi, "Research on Various Cryptography Techniques", International Journal of Resent Technology and engineering (IJRTE), Volume-8, Issue-283, July 2019.

[2] Isnar Sumartono and Andysah Putera Utama Siahaan, "Encryption of DES Algorithm in Information Security", International Journal for Innovative Research in Multidisciplinary Field, Volume – 4, Issue – 10, Oct – 2018.

[3] Sarita Kumari, "A Research Paper on Cryptography Encryption and Compression Techniques", International Journal of Engineering And Computer Science, Volume 6, Issue 4, April 2017, Page No. 20915-20919.

[4] Indumathi Saikumar, "DES-Data Encryption Standard", International Research Journal of Engineering and technology (IRJET), Volume: 04, Issue: 03, Mar-2017.

[5] Isnar Sumartono, Andysah Putera Utama Siahaan, Nova Mayasari, "An Overview of RC4 Algorithm", IOSR Journal of Computer Engineering (IOSR-JCE), Volume 18, Issue 6, Ver. IV (Nov – Dec. 2016). PP. 67-73.

[6] "Security", in Wikipedia, Wikimedia Foundation, 2016,

[7] S. Artheeswari and Dr. RM. Chandrasekaran, "International Data Encryption Algorithm (Idea) For Data Security In Cloud", International Journal of Technology and Engineering System (IJTES)", Vol. 8, No.1 – Jan-March 2016, pp.06-11.

[8] Rajdeep Bhanot and Rahul Hans, "A Review and Comparative Analysis of Various Encryption Algorithms", International Journal of Security and its Applications (SCOPUS), Vol. 9. No. 4 (2015, pp. 389 – 306.

[9] Benni Purnama, Hetty Rohayani.AH, "A New Modified Caesar Cipher Cryptography Method With Legible Ciphertext From A Message To Be Encrypted", International Conference on Computer Science and Computational Intelligence (ICCSCI 2015), Procedia Computer Science, Vol. 59, pp. 195–204, 2015.

[10] Poonam Jindal, Brahmjit Singh, "Performance Analysis of Modified RC4 Encryption Algorithm", IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), May 09-11, 2014.

[11] Ashwak Alabaichi, Faudziah Ahmad and Ramlan Mahmod, "Security Analysis of Blowfish algorithm", IEEE Conference Paper – September 2013.

[12] Osama Almasri and Hajar Mat Jani, "Introducing an Encryption Algorithm based on IDEA", International Journal of Science and Research (IJSR), Volume 2, Issue 9, pp. 334-339, September 2013.

[13] NIST, "ADVANCED ENCRYPTION STANDARD (AES)", Federal Information Processing Standards Publication (FIPS) 197, November 26, 2001.

[14] Aamer Nadeem, Dr M. Younus Javed, "A Performance Comparison of Data Encryption Algorithms", International Conference on Information and Communication Technology", IEEE, pp. 84-89, 2005.

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
*DOI: https://doi.org/10.17762/ijritcc.v11i10.8485*
*Article Received: 28 July 2023 Revised: 15 September 2023 Accepted: 04 October 2023*

[15] Chandra Kanth, P., & Anbarasi, M. S. (2020). A generic framework for data analysis in privacy-preserving data mining. In Computational Intelligence in Data Mining: Proceedings of the International Conference on ICCIDM 2018 (pp. 653-661). Springer Singapore.