# A New Combined Symmetric Key Cryptography CRDDBT Using - Relative Displacement (RDC) and Dynamic Base Transformation (DBTC)

Nehal Kandele, Shrikant Tiwari
*Department of Computer Science & Engineering (CSE)*
*Shri Shankaracharya Technical Campus (SSTC), Shri Shankaracharya Group of Institutions (SSGI)*
*Faculty of Engineering and Technology, Junwani, Bhilai, District-Durg, C.G., 490020*

## Abstract

*In this work the author has introduced a new cryptographic method for encryption and decryption of data. An innovative technique for the same is presented here which is based on division of string into odd and even ordered square matrices, operations using magic square matrices, rotation operation on matrices and also the base conversion by using the key derived from the string itself. Thus, this makes the encryption not only much safe but also act dynamically according to the accepted input. This algorithm not only overcomes the problem of repetition of characters, but also reduces, too many times, the probability of brute attack, that is determining the word formation by the structure of string. This algorithm can be used to accommodate all the characters having ASCII values from 0 to 255. This method has been tested for various input strings and the results derived were found to be remarkable. There was no pattern found in the output, proving it safe for the valuable data.*

## 1. Introduction

In past two decades, the usage of application of internet has increased tremendously in all the fields, may be medical science, research, commerce, education, communication and many more. Data security may not be essential for some of its applications but in many situations, the data security becomes the top priority. The crucial information cannot be communicated in an unprotected or bare format. This forms the basic reason for the generation of various data encryption and decryption algorithms, to protect the data from being stolen over the internet while being transmitted. We assume a situation where the security forces want to communicate from one location to another under the coverage of high level of security. If some intruder intercepts the message and deforms it, either by changing the contents or by the misuse of the information, there can be disasters. So, whenever we send a message we should try to encrypt it in such a way that, even after prolonged intercepting of encrypted message, the hacker could not easily find the encryption algorithm. Now a days, the hackers have become too smart and intelligent, to derive the pattern of string formation by brute attack. So, with the increasing attacks and levels of attacking, we need to generate a high security, pattern distorting encryption algorithm to preserve the confidential data.

In the present work, we present an encryption algorithm that not only changes the data but also changes the technique of encryption. This is not we end up with, we also change the relative position of some adjoining data items while keeping others constant and perform base conversion on another set of data items. The base conversion is dynamic in nature that is the value of base is decided on run time depending upon the input string. Thus making it too difficult for the hackers to recognize the encryption technique. Also, the key is being generated in a way that need not be sent, since the algorithm is designed in such a way that it can itself generate the key, thereby making it safer.

## 2. Basic terminology

### Base conversion

Base conversion is a technique of converting a given number from one number system to another by means of simple calculations.

### Magic square matrix

Magic square matrix is a square matrix in which the sum of all elements in each column and in each row is same. The sum can be calculated from the formula $(n*(n^2+1))/2$, where n is the size of square matrix.

### Pattern rotation

Pattern rotation involves a series of steps to shift the position of elements in a particular predefined pattern.

### A square matrix of even order

A square matrix of even order refers to a matrix with equal number of rows and columns and this number is even, that is divisible by 2. Example, matrix of order 2x2, 4x4, 6x6, etc.

### A square matrix of odd order

A square matrix of odd order refers to a matrix with equal number of rows and columns and this number is odd, that is not divisible by 2. Example, matrix of order 1x1, 3x3, 5x5, etc.

## 3. Proposed encryption algorithm

### Step-1

We calculate the length of the given input string and assign into the variable N. Each element of the input string is then converted into its corresponding ASCII value.

Consider that the entered input string is “COMPUTER IS A USEFUL ELECTRONIC DIGITAL DEVICE!!”
Here, length of string N = 48
So the ASCII equivalent of the string is = [67 79 77 80 85 84 69 82 32 73 83 32 65 32 85 83 69 70 85 76 32 69 76 69 67 84 82 79 78 73 67 32 68 73 71 73 84 65 76 32 68 69 86 73 67 69 33 33]

### Step-2

We break the input string into square matrices of maximum possible size and place the remaining elements into a variable REM. This step is repeated, using remainder REM of this step as input string, until there are 4 or more elements in REM.

So, in this example matrices are:

$$\begin{bmatrix} 67 & 79 & 77 & 80 & 85 & 84 \\ 69 & 82 & 32 & 73 & 83 & 32 \\ 65 & 32 & 85 & 83 & 69 & 70 \\ 85 & 76 & 32 & 69 & 76 & 69 \\ 67 & 84 & 82 & 79 & 78 & 73 \\ 67 & 32 & 68 & 73 & 71 & 73 \end{bmatrix}_{6\,x\,6} \quad \begin{bmatrix} 84 & 65 & 76 \\ 32 & 68 & 69 \\ 86 & 73 & 67 \end{bmatrix}_{3\,x\,3}$$

REM = [69 33 33]

### Step-3

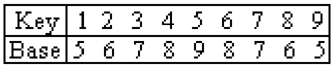
Calculating the KEY by KEY1 $\otimes$ KEY2, where, KEY1 can be calculated as sum of numbers of columns of all matrices and number of elements in REM. KEY2 can be calculated as the sum of magic square matrix of size same as that of last generated square matrix. Here, $\otimes$ refers to addition. If the KEY is greater than 9, then sum all the digits until a single digit KEY is obtained.

Here, KEY1 = 6 + 3 + 3 = 12, KEY2 = 3
Therefore KEY = 12 + 3 = 15 = 1 + 5 = 6

### Step-4

The derived KEY is then used to calculate the BASE using the key-base table which is as follows:

Table 1. Key-Base table



| Key | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| Base | 5 | 6 | 7 | 8 | 9 | 8 | 7 | 6 | 5 |

Here, Key = 6, So BASE = 8

### Step-5

For all the matrices of odd order, add the magic square matrix of size same as that of the matrix to it and then perform the base conversion using the derived BASE.

In this example, adding magic square matrix to the result obtained from step2:

$$\begin{bmatrix} 84 & 65 & 76 \\ 32 & 68 & 69 \\ 86 & 73 & 67 \end{bmatrix} + \begin{bmatrix} 8 & 1 & 6 \\ 3 & 5 & 7 \\ 4 & 9 & 2 \end{bmatrix} = \begin{bmatrix} 92 & 66 & 82 \\ 35 & 73 & 76 \\ 90 & 82 & 69 \end{bmatrix}$$

Now after performing base conversion to the result, the desired cipher is: [134 102 122 043 111 114 132 122 105]

### Step-6

For all the matrices of even order, first we apply the following rotation pattern and add magic square matrix of size same as that of the matrix to it and then finally subtract KEY from each element of the resultant matrix.
The sequence of the steps in rotation pattern can be listed as follows, assume that the matrix is:

| | | | | | |
|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 |
| 7 | 8 | 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 | 17 | 18 |
| 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | 29 | 30 |
| 31 | 32 | 33 | 34 | 35 | 36 |

**(i) In the first rotation** interchanging the even columns in the following manner:

| | | | | | |
|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 |
| 7 | 8 | 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 | 17 | 18 |
| 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | 29 | 30 |
| 31 | 32 | 33 | 34 | 35 | 36 |

| | | | | | |
|---|---|---|---|---|---|
| 1 | 6 | 3 | 2 | 5 | 4 |
| 7 | 12 | 9 | 8 | 11 | 10 |
| 13 | 18 | 15 | 14 | 17 | 16 |
| 19 | 24 | 21 | 20 | 23 | 22 |
| 25 | 30 | 27 | 26 | 29 | 28 |
| 31 | 36 | 33 | 32 | 35 | 34 |

**(ii) In second rotation** interchanging the even rows in the following manner:

| | | | | | |
|---|---|---|---|---|---|
| 1 | 6 | 3 | 2 | 5 | 4 |
| 7 | 12 | 9 | 8 | 11 | 10 |
| 13 | 18 | 15 | 14 | 17 | 16 |
| 19 | 24 | 21 | 20 | 23 | 22 |
| 25 | 30 | 27 | 26 | 29 | 28 |
| 31 | 36 | 33 | 32 | 35 | 34 |

| | | | | | |
|---|---|---|---|---|---|
| 1 | 6 | 3 | 2 | 5 | 4 |
| 31 | 36 | 33 | 32 | 35 | 34 |
| 13 | 18 | 15 | 14 | 17 | 16 |
| 7 | 12 | 9 | 8 | 11 | 10 |
| 25 | 30 | 27 | 26 | 29 | 28 |
| 19 | 24 | 21 | 20 | 23 | 22 |

**(iii) In third rotation** performing single-diagonal-left-up shift as shown below:

| | | | | | |
|---|---|---|---|---|---|
| 1 | 6 | 3 | 2 | 5 | 4 |
| 31 | 36 | 33 | 32 | 35 | 34 |
| 13 | 18 | 15 | 14 | 17 | 16 |
| 7 | 12 | 9 | 8 | 11 | 10 |
| 25 | 30 | 27 | 26 | 29 | 28 |
| 19 | 24 | 21 | 20 | 23 | 22 |

| | | | | | |
|---|---|---|---|---|---|
| 36 | 6 | 3 | 2 | 5 | 4 |
| 31 | 15 | 33 | 32 | 35 | 34 |
| 13 | 18 | 8 | 14 | 17 | 16 |
| 7 | 12 | 9 | 29 | 11 | 10 |
| 25 | 30 | 27 | 26 | 22 | 28 |
| 19 | 24 | 21 | 20 | 23 | 1 |

**(iv) In fourth rotation** performing single-diagonal-right-up shift as shown below:

| | | | | | |
|---|---|---|---|---|---|
| 36 | 6 | 3 | 2 | 5 | 4 |
| 31 | 15 | 33 | 32 | 35 | 34 |
| 13 | 18 | 8 | 14 | 17 | 16 |
| 7 | 12 | 9 | 29 | 11 | 10 |
| 25 | 30 | 27 | 26 | 22 | 28 |
| 19 | 24 | 21 | 20 | 23 | 1 |

| | | | | | |
|---|---|---|---|---|---|
| 36 | 6 | 3 | 2 | 5 | 35 |
| 31 | 15 | 33 | 32 | 14 | 34 |
| 13 | 18 | 8 | 9 | 17 | 16 |
| 7 | 12 | 30 | 29 | 11 | 10 |
| 25 | 19 | 27 | 26 | 22 | 28 |
| 4 | 24 | 21 | 20 | 23 | 1 |

**(v) In fifth rotation** applying single-up shift to the every even column as follows:

| | | | | | |
|---|---|---|---|---|---|
| 36 | 6 | 3 | 2 | 5 | 35 |
| 31 | 15 | 33 | 32 | 14 | 34 |
| 13 | 18 | 8 | 9 | 17 | 16 |
| 7 | 12 | 30 | 29 | 11 | 10 |
| 25 | 19 | 27 | 26 | 22 | 28 |
| 4 | 24 | 21 | 20 | 23 | 1 |

| | | | | | |
|---|---|---|---|---|---|
| 36 | 15 | 3 | 32 | 5 | 34 |
| 31 | 18 | 33 | 9 | 14 | 16 |
| 13 | 12 | 8 | 29 | 17 | 10 |
| 7 | 19 | 30 | 26 | 11 | 28 |
| 25 | 24 | 27 | 20 | 22 | 1 |
| 4 | 6 | 21 | 2 | 23 | 35 |

**(vi) In the sixth and last rotation** rotating once the outer most cycle in clock wise direction, its inner circle in anti-clock wise direction, and so on as shown below:

| | | | | | |
|---|---|---|---|---|---|
| 36→ | 15→ | 3→ | 32→ | 5→ | 34 ↓ |
| ↑ 31 | 18 ↓ | ←33 | ←9 | ←14 | 16 ↓ |
| ↑ 13 | 12 ↓ | 8→ | 29 ↓ | ↑ 17 | 10 ↓ |
| ↑ 7 | 19 ↓ | ↑ 30 | ←26 | ↑ 11 | 28 ↓ |
| ↑ 25 | 24→ | 27→ | 20→ | ↑ 22 | 1 ↓ |
| ↑ 4 | ←6 | ←21 | ←2 | ←23 | ←35 |

| | | | | | |
|---|---|---|---|---|---|
| 31 | 36 | 15 | 3 | 32 | 5 |
| 13 | 33 | 9 | 14 | 17 | 34 |
| 7 | 18 | 30 | 8 | 11 | 16 |
| 25 | 12 | 26 | 29 | 22 | 10 |
| 4 | 19 | 24 | 27 | 20 | 28 |
| 6 | 21 | 2 | 23 | 35 | 1 |

So, the resultant matrix of above example after these rotation operations is:

$$\begin{bmatrix} 67 & 73 & 85 & 77 & 32 & 85 \\ 65 & 68 & 32 & 32 & 69 & 73 \\ 69 & 70 & 73 & 82 & 83 & 83 \\ 67 & 32 & 84 & 78 & 69 & 73 \\ 80 & 85 & 69 & 82 & 76 & 79 \\ 84 & 32 & 79 & 76 & 71 & 67 \end{bmatrix}$$

Adding magic square matrix of size 6 x 6

$$\begin{bmatrix} 67 & 73 & 85 & 77 & 32 & 85 \\ 65 & 68 & 32 & 32 & 69 & 73 \\ 69 & 70 & 73 & 82 & 83 & 83 \\ 67 & 32 & 84 & 78 & 69 & 73 \\ 80 & 85 & 69 & 82 & 76 & 79 \\ 84 & 32 & 79 & 76 & 71 & 67 \end{bmatrix} + \begin{bmatrix} 35 & 1 & 6 & 26 & 19 & 24 \\ 3 & 32 & 7 & 21 & 23 & 25 \\ 31 & 9 & 2 & 22 & 27 & 20 \\ 8 & 28 & 33 & 17 & 10 & 15 \\ 30 & 5 & 34 & 12 & 14 & 16 \\ 4 & 36 & 29 & 13 & 18 & 11 \end{bmatrix} =$$

$$\begin{bmatrix} 102 & 74 & 91 & 103 & 51 & 109 \\ 68 & 100 & 39 & 53 & 92 & 98 \\ 100 & 79 & 75 & 104 & 110 & 103 \\ 75 & 60 & 117 & 95 & 79 & 88 \\ 110 & 90 & 103 & 94 & 90 & 95 \\ 88 & 68 & 108 & 89 & 89 & 78 \end{bmatrix}$$

Now subtracting KEY from each element of resultant matrix.

String = [102 74 91 103 51 109 68 100 39 53 92 98 100 79 75 104 110 103 75 60 117 95 79 88 110 90 103 94 90 95 88 68 108 89 89 78]

After subtracting KEY = [96 68 85 97 45 103 62 94 33 47 86 92 94 73 69 98 104 97 69 54 111 89 73 82 104 84 97 88 84 89 82 62 102 83 83 72]

## Step-7

For all the elements in REM adding the sum of magic square matrix of size same as that of the last square matrix generated and then performing base conversion according to the derived BASE.

Here, size of the last square matrix generated = 3,
So sum of magic square matrix of size 3 = 15,
BASE = 8, REM = [69 33 33]
So, REM = [69+15 33+15 33+15] = [84 48 48]
After base conversion of the above result, the obtained resultant string is = [124 060 060]

### Step-8

In the last step, we merge all the square matrices and REM in the order they were derived, to form the cipher text.

Here, matrices are:

$$\begin{bmatrix} 96 & 68 & 85 & 97 & 45 & 103 \\ 62 & 94 & 33 & 47 & 86 & 92 \\ 94 & 73 & 69 & 98 & 104 & 97 \\ 69 & 54 & 111 & 89 & 73 & 82 \\ 104 & 84 & 97 & 88 & 84 & 89 \\ 82 & 62 & 102 & 83 & 83 & 72 \end{bmatrix} \begin{bmatrix} 134 & 102 & 122 \\ 043 & 111 & 114 \\ 132 & 122 & 105 \end{bmatrix}$$

[124 060 060]

So, Cipher text is = [96 68 85 97 45 103 62 94 33 47 86 92 94 73 69 98 104 97 69 54 111 89 73 82 104 84 97 88 84 89 82 62 102 83 83 72 134 102 122 043 111 114 132 122 105 124 060 060]

## 4. Proposed decryption algorithm

### Step-1

We calculate the length of the given input string and assign into the variable N.

Consider that the entered input string is = [96 68 85 97 45 103 62 94 33 47 86 92 94 73 69 98 104 97 69 54 111 89 73 82 104 84 97 88 84 89 82 62 102 83 83 72 134 102 122 043 111 114 132 122 105 124 060 060]
Here, length of string N=48

### Step-2

We break the input string into square matrices of maximum possible size and place the remaining elements into a variable REM. This step is repeated, using remainder REM of this step as input string, until there are 4 or more elements in REM.
So, in this example matrices are:

$$\begin{bmatrix} 96 & 68 & 85 & 97 & 45 & 103 \\ 62 & 94 & 33 & 47 & 86 & 92 \\ 94 & 73 & 69 & 98 & 104 & 97 \\ 69 & 54 & 111 & 89 & 73 & 82 \\ 104 & 84 & 97 & 88 & 84 & 89 \\ 82 & 62 & 102 & 83 & 83 & 72 \end{bmatrix}_{6 \times 6} \begin{bmatrix} 134 & 102 & 122 \\ 043 & 111 & 114 \\ 132 & 122 & 105 \end{bmatrix}_{3 \times 3}$$

REM = [124 060 060]

### Step-3

Calculating the KEY by KEY1 $\otimes$ KEY2, where, KEY1 can be calculated as sum of numbers of columns of all matrices and number of elements in REM. KEY2 can be calculated as the sum of magic square matrix of size same as that of last generated square matrix. Here, $\otimes$ refers to addition. If the KEY is greater than 9, then sum all the digits until a single digit KEY is obtained.

Here, KEY1 = 6 + 3 + 3 = 12, KEY2 = 3
Therefore KEY = 12 + 3 = 15 = 1 + 5 = 6

### Step-4

The derived KEY is then used to calculate the BASE using the key-base table which is as follows:

Table 2. Key-Base table

| Key | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| Base | 5 | 6 | 7 | 8 | 9 | 8 | 7 | 6 | 5 |

Here, Key = 6, so BASE = 8

### Step-5

For all the matrices of odd order, perform the base to decimal conversion using the derived BASE and then subtract the magic square matrix of size same as that of the matrix to it.

In this example, matrix is:

$$\begin{bmatrix} 134 & 102 & 122 \\ 043 & 111 & 114 \\ 132 & 122 & 105 \end{bmatrix}$$

After base to decimal conversion matrix is:

$$\begin{bmatrix} 92 & 66 & 82 \\ 35 & 73 & 76 \\ 90 & 82 & 69 \end{bmatrix}$$

Subtracting magic square matrix:

$$\begin{bmatrix} 92 & 66 & 82 \\ 35 & 73 & 76 \\ 90 & 82 & 69 \end{bmatrix} - \begin{bmatrix} 8 & 1 & 6 \\ 3 & 5 & 7 \\ 4 & 9 & 2 \end{bmatrix} = \begin{bmatrix} 84 & 65 & 76 \\ 32 & 68 & 69 \\ 86 & 73 & 67 \end{bmatrix}$$

### Step-6

For all the matrices of even order, first add KEY to each element of the matrix, then subtract magic square matrix of size same as that of the matrix from it and then apply the following rotation pattern.
The sequence of steps in rotation pattern can be listed as follows, assume that the matrix is:

| | | | | | |
|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 |
| 7 | 8 | 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 | 17 | 18 |
| 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | 29 | 30 |
| 31 | 32 | 33 | 34 | 35 | 36 |

**(i) In the first rotation** rotating once the outer most cycle in anti clock wise direction, its inner circle in clock wise direction, and so on as shown below:

| | | | | | |
|---|---|---|---|---|---|
| 1↓ | ←2 | ←3 | ←4 | ←5 | ←6 |
| 7↓ | 8→ | 9→ | 10→ | 11↓ | ↑12 |
| 13↓ | ↑14 | 15↓ | ←16 | 17↓ | ↑18 |
| 19↓ | ↑20 | 21→ | ↑22 | 23↓ | ↑24 |
| 25↓ | ↑26 | ←27 | ←28 | ←29 | ↑30 |
| 31→ | 32→ | 33→ | 34→ | 35→ | ↑36 |

| | | | | | |
|---|---|---|---|---|---|
| 2 | 3 | 4 | 5 | 6 | 12 |
| 1 | 14 | 8 | 9 | 10 | 18 |
| 7 | 20 | 16 | 22 | 11 | 24 |
| 13 | 26 | 15 | 21 | 17 | 30 |
| 19 | 27 | 28 | 29 | 23 | 36 |
| 25 | 31 | 32 | 33 | 34 | 35 |

**(ii) In second rotation** applying single-down shift to every even column as follows:

| | | | | | |
|---|---|---|---|---|---|
| 2 | 3 | 4 | 5 | 6 | 12 |
| 1 | 14 | 8 | 9 | 10 | 18 |
| 7 | 20 | 16 | 22 | 11 | 24 |
| 13 | 26 | 15 | 21 | 17 | 30 |
| 19 | 27 | 28 | 29 | 23 | 36 |
| 25 | 31 | 32 | 33 | 34 | 35 |

| | | | | | |
|---|---|---|---|---|---|
| 2 | 31 | 4 | 33 | 6 | 35 |
| 1 | 3 | 8 | 5 | 10 | 12 |
| 7 | 14 | 16 | 9 | 11 | 18 |
| 13 | 20 | 15 | 22 | 17 | 24 |
| 19 | 26 | 28 | 21 | 23 | 30 |
| 25 | 27 | 32 | 29 | 34 | 36 |

**(iii) In third rotation** performing single-diagonal-left-down shift as shown below:

| | | | | | |
|---|---|---|---|---|---|
| 2 | 31 | 4 | 33 | 6 | 35 |
| 1 | 3 | 8 | 5 | 10 | 12 |
| 7 | 14 | 16 | 9 | 11 | 18 |
| 13 | 20 | 15 | 22 | 17 | 24 |
| 19 | 26 | 28 | 21 | 23 | 30 |
| 25 | 27 | 32 | 29 | 34 | 36 |

| | | | | | |
|---|---|---|---|---|---|
| 2 | 31 | 4 | 33 | 6 | 25 |
| 1 | 3 | 8 | 5 | 35 | 12 |
| 7 | 14 | 16 | 10 | 11 | 18 |
| 13 | 20 | 9 | 22 | 17 | 24 |
| 19 | 15 | 28 | 21 | 23 | 30 |
| 26 | 27 | 32 | 29 | 34 | 36 |

**(iv) In fourth rotation** performing single-diagonal-right-down shift as shown below:

| | | | | | |
|---|---|---|---|---|---|
| 2 | 31 | 4 | 33 | 6 | 25 |
| 1 | 3 | 8 | 5 | 35 | 12 |
| 7 | 14 | 16 | 10 | 11 | 18 |
| 13 | 20 | 9 | 22 | 17 | 24 |
| 19 | 15 | 28 | 21 | 23 | 30 |
| 26 | 27 | 32 | 29 | 34 | 36 |

| | | | | | |
|---|---|---|---|---|---|
| 36 | 31 | 4 | 33 | 6 | 25 |
| 1 | 2 | 8 | 5 | 35 | 12 |
| 7 | 14 | 3 | 10 | 11 | 18 |
| 13 | 20 | 9 | 16 | 17 | 24 |
| 19 | 15 | 28 | 21 | 22 | 30 |
| 26 | 27 | 32 | 29 | 34 | 23 |

**(v) In fifth rotation** interchanging the even rows in the following manner:

| | | | | | |
|---|---|---|---|---|---|
| 36 | 31 | 4 | 33 | 6 | 25 |
| 1 | 2 | 8 | 5 | 35 | 12 |
| 7 | 14 | 3 | 10 | 11 | 18 |
| 13 | 20 | 9 | 16 | 17 | 24 |
| 19 | 15 | 28 | 21 | 22 | 30 |
| 26 | 27 | 32 | 29 | 34 | 23 |

| | | | | | |
|---|---|---|---|---|---|
| 36 | 31 | 4 | 33 | 6 | 25 |
| 13 | 20 | 9 | 16 | 17 | 24 |
| 7 | 14 | 3 | 10 | 11 | 18 |
| 26 | 27 | 32 | 29 | 34 | 23 |
| 19 | 15 | 28 | 21 | 22 | 30 |
| 1 | 2 | 8 | 5 | 35 | 12 |

**(vi) In the sixth and last rotation** interchanging the even columns in the following manner:

| | | | | | |
|---|---|---|---|---|---|
| 36 | 31 | 4 | 33 | 6 | 25 |
| 13 | 20 | 9 | 16 | 17 | 24 |
| 7 | 14 | 3 | 10 | 11 | 18 |
| 26 | 27 | 32 | 29 | 34 | 23 |
| 19 | 15 | 28 | 21 | 22 | 30 |
| 1 | 2 | 8 | 5 | 35 | 12 |

| | | | | | |
|---|---|---|---|---|---|
| 36 | 33 | 4 | 25 | 6 | 31 |
| 13 | 16 | 9 | 24 | 17 | 20 |
| 7 | 10 | 3 | 18 | 11 | 14 |
| 26 | 29 | 32 | 23 | 34 | 27 |
| 19 | 21 | 28 | 30 | 22 | 15 |
| 1 | 5 | 8 | 12 | 35 | 2 |

Here in this example, matrix is:

$$\begin{bmatrix} 96 & 68 & 85 & 97 & 45 & 103 \\ 62 & 94 & 33 & 47 & 86 & 92 \\ 94 & 73 & 69 & 98 & 104 & 97 \\ 69 & 54 & 111 & 89 & 73 & 82 \\ 104 & 84 & 97 & 88 & 84 & 89 \\ 82 & 62 & 102 & 83 & 83 & 72 \end{bmatrix}$$

Adding KEY to each element of matrix:

$$\begin{bmatrix} 102 & 74 & 91 & 103 & 51 & 109 \\ 68 & 100 & 39 & 53 & 92 & 98 \\ 100 & 79 & 75 & 104 & 110 & 103 \\ 75 & 60 & 117 & 95 & 79 & 88 \\ 110 & 90 & 103 & 94 & 90 & 95 \\ 88 & 68 & 108 & 89 & 89 & 78 \end{bmatrix}$$

Subtracting magic square matrix

$$\begin{bmatrix} 102 & 74 & 91 & 103 & 51 & 109 \\ 68 & 100 & 39 & 53 & 92 & 98 \\ 100 & 79 & 75 & 104 & 110 & 103 \\ 75 & 60 & 117 & 95 & 79 & 88 \\ 110 & 90 & 103 & 94 & 90 & 95 \\ 88 & 68 & 108 & 89 & 89 & 78 \end{bmatrix} -$$

$$\begin{bmatrix} 35 & 1 & 6 & 26 & 19 & 24 \\ 3 & 32 & 7 & 21 & 23 & 25 \\ 31 & 9 & 2 & 22 & 27 & 20 \\ 8 & 28 & 33 & 17 & 10 & 15 \\ 30 & 5 & 34 & 12 & 14 & 16 \\ 4 & 36 & 29 & 13 & 18 & 11 \end{bmatrix} = \begin{bmatrix} 67 & 73 & 85 & 77 & 32 & 85 \\ 65 & 68 & 32 & 32 & 69 & 73 \\ 69 & 70 & 73 & 82 & 83 & 83 \\ 67 & 32 & 84 & 78 & 69 & 73 \\ 80 & 85 & 69 & 82 & 76 & 79 \\ 84 & 32 & 79 & 76 & 71 & 67 \end{bmatrix}$$

Then performing rotation operation, so resultant matrix is:

$$\begin{bmatrix} 67 & 79 & 77 & 80 & 85 & 84 \\ 69 & 82 & 32 & 73 & 83 & 32 \\ 65 & 32 & 85 & 83 & 69 & 70 \\ 85 & 76 & 32 & 69 & 76 & 69 \\ 67 & 84 & 82 & 79 & 78 & 73 \\ 67 & 32 & 68 & 73 & 71 & 73 \end{bmatrix}$$

## Step-7

For the entire element in REM perform base to decimal conversion according to the derived BASE and then subtract the sum of magic square matrix of size same as that of the last square matrix generated.

Here, size of last square matrix generated = 3,
So sum of magic square matrix of size 3 = 15,
BASE = 8, REM = [124 060 060]
After base to decimal conversion, REM = [84 48 48]

Now subtracting sum of magic square matrix of size 3, so REM = [84-15 48-15 48-15]
= [69 33 33]

### Step-8

In the last step, we merge all the square matrices and REM in the order they were derived, convert into its corresponding ASCII value to form the plain text.

Here, matrices are:

$$\begin{bmatrix} 67 & 79 & 77 & 80 & 85 & 84 \\ 69 & 82 & 32 & 73 & 83 & 32 \\ 65 & 32 & 85 & 83 & 69 & 70 \\ 85 & 76 & 32 & 69 & 76 & 69 \\ 67 & 84 & 82 & 79 & 78 & 73 \\ 67 & 32 & 68 & 73 & 71 & 73 \end{bmatrix} \begin{bmatrix} 84 & 65 & 76 \\ 32 & 68 & 69 \\ 86 & 73 & 67 \end{bmatrix} [69\ 33\ 33]$$

String is = [67 79 77 80 85 84 69 82 32 73 83 32 65 32 85 83 69 70 85 76 32 69 76 69 67 84 82 79 78 73 67 32 68 73 71 73 84 65 76 32 68 69 86 73 67 69 33 33]
So, ASCII equivalent and plain text is:
COMPUTER IS A USEFUL ELECTRONIC
DIGITAL DEVICE!!

## 5. Result

On analyzing, using strings of various lengths, we find that the time taken for encryption is not dependent on the length of the string. There are critical points for the length of the string, where the time taken for encryption becomes too less and also becomes too high. This variation is represented in the following String-Time graph.

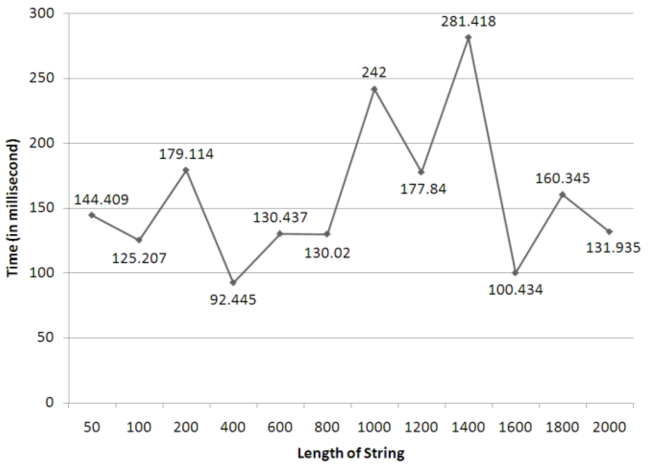


Figure 1. String-Time graph

On calculating, the average time taken per character in the string of varying length, we obtain the following Character-Time graph. This graph shows that, on increasing the length of the string, the average time taken per character for encryption is decreasing. This is one of the most important advantages of this algorithm.

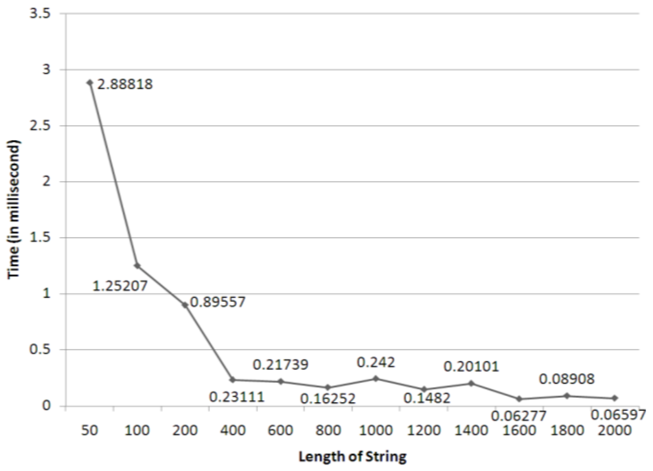


Figure 2. Character-Time graph

## 6. Conclusion

In the present work, we come up with a new technique of encryption without a predefined key. The input string is fragmented into several parts, with each part encrypted using a different algorithm. On the whole, three unique algorithms have been applied to encrypt the fragmented string on the basis of its orientation. For higher security levels, the key is derived from the two differently determined keys. The salient feature of this algorithm is that, a part of string is manipulated using base conversion, second part of string is deformed by interchanging position and increasing number of repetitions, and in the remaining elements, we perform simple operations. So, this algorithm is a complex combination without involving any complex calculation.

## 7. Acknowledgement

I am highly indebted to Dr. R. N. Patel sir for his invaluable guidance and suggestions. I owe much of thanks to my brother, Suyash Kandele for his support.

## 8. References

[1] Ya-Ping Zhang, Jizhou Sun, and Xu Zhang, “A Stream Cipher Algorithm Based on Conventional Encryption Techniques”, *IEEE*, 0-7803-8253-6/04, 2004.

[2] A. Chandra Sekhar, K.R. Sudha, and Prasad Reddy P V G D, "Data Encryption technique using Random number generator", *IEEE Computer Society*, 0-7695-3032-X/07, DOI 10.1109/GrC.2007.73, 2007.

[3] Albert H. Carlson, Robert E. Hiromoto, and Richard B. Wells, "Breaking Block and Product Ciphers Applied Across Byte Boundaries", *IEEE*, 978-1-4577-1425-2/11, 2011.

[4] Dripto Chatterjee, Suvadeep Dasgupta, Joyshree Nath, and Asoke Nath, "A new Symmetric key Cryptography Algorithm using extended MSA method: DJSA symmetric key algorithm", *IEEE*, 978-0-7695-4437-3/11, DOI 10.1109/CSNT.2011.25, 2011.

[5] Zhang Yunpeng, Zhu Yu, Wang Zhong, and Richard O.Sinnott, "Index-Based Symmetric DNA Encryption Algorithm", *IEEE*, 978-1-4244-9306-7/11, 2011.

[6] Neeraj Khanna, Joyshree Nath, Joel James, Amlan Chakrabarti, Sayantan Chakraborty, and Asoke Nath, “New Symmetric Key Cryptographic algorithm using combined bit manipulation and MSA encryption algorithm: NJJSAA symmetric key algorithm”, *IEEE Computer Society*, 978-0-7695-4437-3/11, DOI 10.1109/ CSNT.2011.33, 2011.

[7] D. Rajavel, and S. P. Shantharajah, “Cubical Key Generation and Encryption Algorithm Based on Hybrid Cube’s Rotation”, *IEEE*, 978-1-4673-1039-0/12, March 21-23, 2012.

[8] Marcin Niemiec, and Lukasz Machowski, “A new symmetric block cipher based on key-dependent S-boxes”, *IEEE*, 978-1-4673-2015-3/12, 2012.

[9] Hai Cheng, and Qun Ding “Overview of the Block Cipher”, *IEEE Computer Society*, 978-0-7695-4935-4/12, DOI 10.1109/IMCCC.2012.379, 2012.

[10] Gaurav Bhadra, Tanya Bala, Samik Banik, Asoke Nath, and Joyshree Nath,"Bit Level Encryption Standard (BLES): Version-II", *IEEE*, 978-1-4673-4805-8/12, 2012.

[11] Rishav Ray, Jeeyan Sanyal, Debanjan Das, and Asoke Nath, “A new Challenge of hiding any encrypted secret message inside any Text/ASCII file or in MS word file: RJDA Algorithm”, *IEEE*, 978-0-7695-4692-6/12, DOI 10.1109/CSNT.2012.191, 2012.

[12] Somdip Dey, "SD-C1BBR: SD-Count-1-Byte-Bit Randomization: A New Advanced Cryptographic Randomization Technique", *IEEE*, 978-1-4673-4805-8/12, 2012.

[13] Akanksha Mathur, “A Research paper: An ASCII value based data encryption algorithm and its comparison with other symmetric data encryption algorithms”, *International Journal on Computer Science and Engineering*, ISSN: 0975-3397, vol. 4, no. 09, 2012, pp. 1650-1657.

[14] Sayak Guha, Tamodeep Das, Saima Ghosh, Joyshree Nath, Sankar Das, and Asoke Nath, "A New Data Hiding Algorithm With Encrypted Secret Message Using TTJSA Symmetric Key Crypto System", *Journal of Global Research in Computer Science*,ISSN-2229-371X, vol. 3, no. 4, April 2012.

[15] Somdip Dey, "SD-AREE: An Advanced Modified Caesar Cipher Method to Exclude Repetition from a Message", *International Journal of Information and Network Security*, vol. 1, no. 2, ISSN: 2089-3299, June 2012.

[16] Rajavel D, and Shantharajah S. P, "Cryptography Based on Combination of Hybridization and Cube’s Rotation", *International Journal of Computational Intelligence and Informatics*, vol. 1: no. 4, ISSN: 2231-0258, March 2012.

[17] Somdip Dey, Joyshree Nath, and Asoke Nath, "An Integrated Symmetric Key Cryptographic Method-Amalgamation of TTJSA Algorithm, Advanced Caesar Cipher Algorithm, Bit Rotation and Reversal Method: SJA Algorithm", *I. J. Modern Education and Computer Science*, DOI: 10.5815/ijmecs.2012.05.01, 2012.

[18] Somdip Dey, Kalyan Mondal, Joyshree Nath, and Asoke Nath, "Advanced Steganography Algorithm Using Randomized Intermediate QR Host Embedded With Any Encrypts Secret Message: ASA_QR Algorithm", *I. J. Modern Education and Computer Science*, DOI: 10.5815/ijmecs.2012.06.08, 2012.

[19] Mr. Rangaswamy D. A., and Mr. Punithkumar M. B., "New Symmetric Key Cryptographic Algorithm Using Combined Bit Manipulation and MSA Encryption Algorithm: NJJSAA Symmetric Key Algorithm", *International Journal of Innovative Research and Development*, vol. 2, Issue 6, ISSN: 2278-0211, June 2013.

[20] Thanapal P, Muthamil Selvan T, and Pratheeba T, "An Integrated Cryptography Approach Using MSA Symmetric Key", *International Journal of Engineering Research and Technology*, vol. 2, Issue 3, ISSN:2278-0181, March 2013.

**Nehal Kandele**, B.E., M.E. Scholar in Computer Technology & Application from Shri Shankaracharya Technical Campus, Bhilai, Chhattisgarh, India. Research areas are Computer Network and Cryptography.

**Dr. Shrikant Tiwari**, currently working as Assistant Professor in Department of Computer Science & Engineering at Shri Shankaracharya Technical Campus, Bhilai, Chhattisgarh, India. He has received his Ph.D in Department of Computer Science and Engineering from the Indian Institute of Technology (Banaras Hindu University), Varanasi. He has published more than 20 papers in international journal and conference and also published 5 Book Chapters.