String and Image Encryption using DNA Encryption, Randomly Generated Moore Machine and Hyperchaotic System

Presented by:

Saai Sudarsanan D
(123003212)

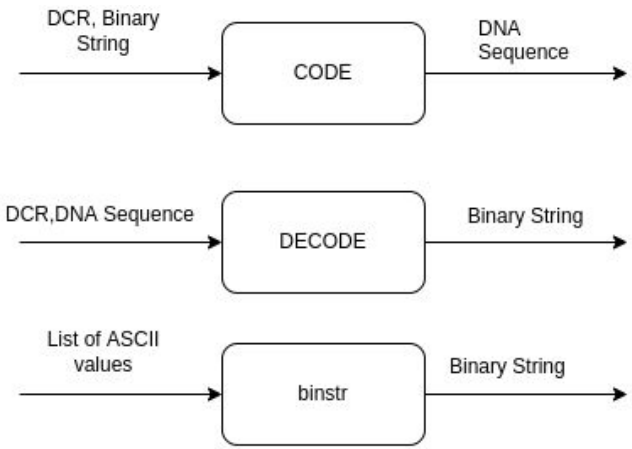
Aravind M (123003022)

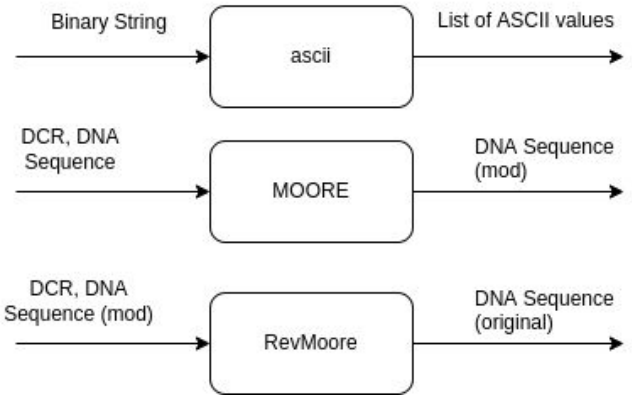
Guided by:
Dr. Kannan Balasubramanian,

Professor, SOC,

SASTRA Deemed University,
Thanjavur

Components


DCR, Binary String
CODE
DNA Sequence
DCR,DNA Sequence
DECODE
Binary String
List of ASCII values
binstr
Binary String

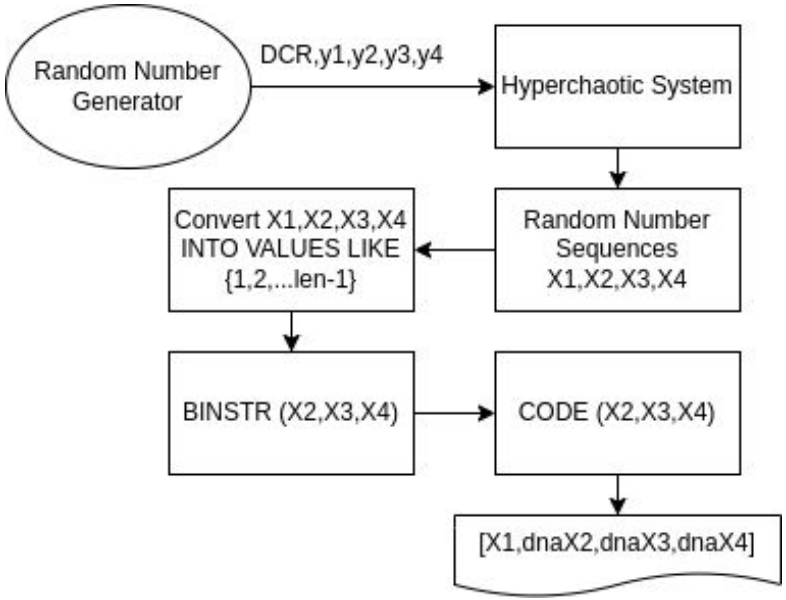

Binary String
ascii
List of ASCII values
DCR, DNA Sequence
MOORE
DNA Sequence (mod)
DCR, DNA Sequence (mod)
RevMoore
DNA Sequence (original)

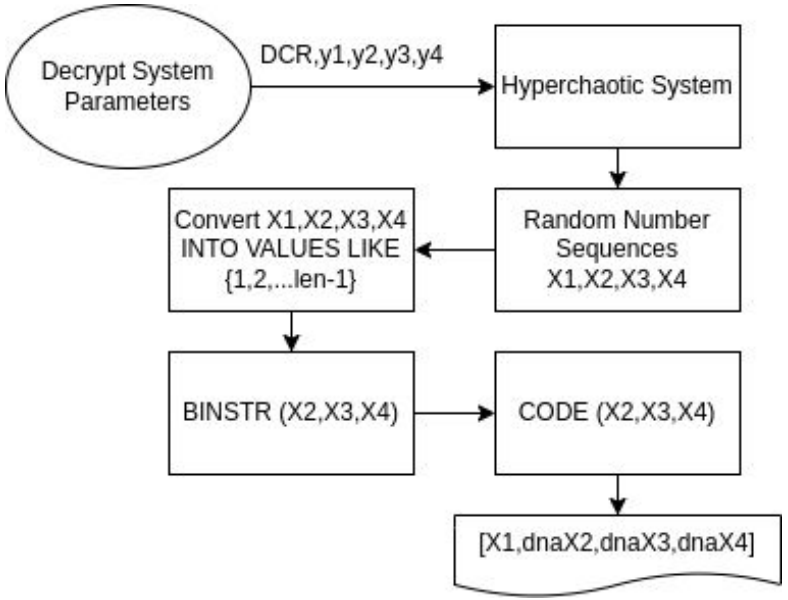
String Encryption

The string encryption algorithm has been adapted directly from the paper and no novelty has been introduced by us when making it.

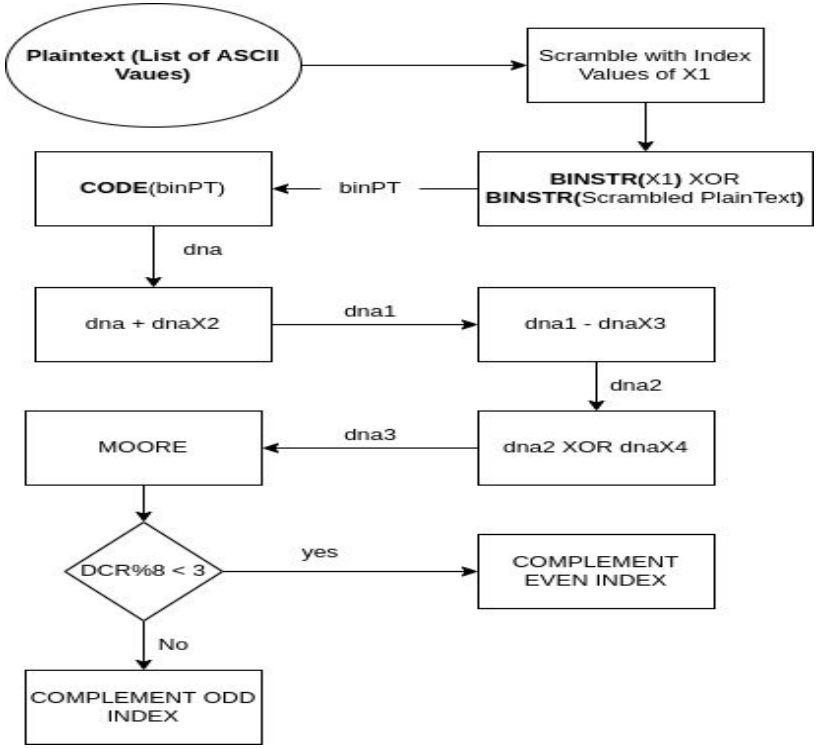
- The algorithm involves the use of the Rossler's fourth order hyperchaotic system and a randomly generated moore machine.
- The algorithm also uses DNA Encoding techniques and some basic DNA operations further strengthen the encryption.

Key Generation and Retrieval





Encryption


Plaintext (List of ASCII Vaues)
Scramble with Index Values of X1
BINSTR(X1) XOR BINSTR(Scrambled PlainText)
binPT
CODE(binPT)
dna
dna + dnaX2
dna1
dna1 - dnaX3
dna2
dna2 XOR dnaX4
dna3
MOORE
DCR%8 < 3
yes
COMPLEMENT EVEN INDEX
No
COMPLEMENT ODD INDEX

Decryption

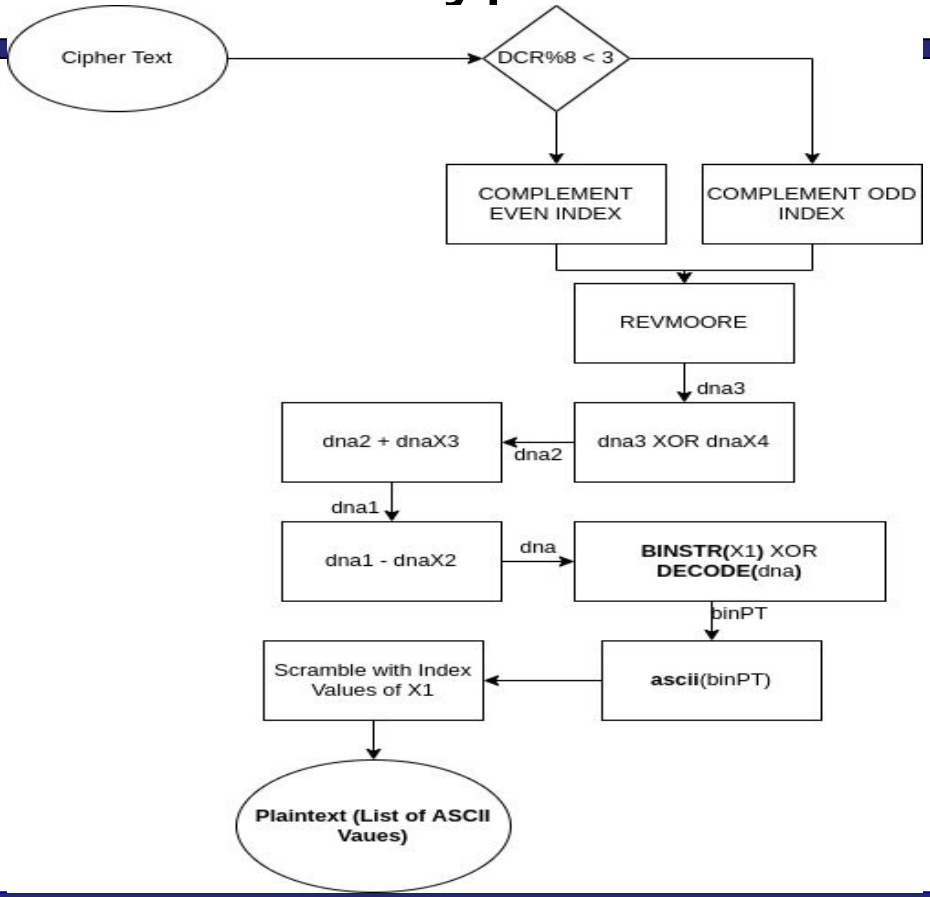

Cipher Text
DCR%8 < 3
COMPLEMENT EVEN INDEX
COMPLEMENT ODD INDEX
REVMOORE
dna3
dna2 + dnaX3
dna2
dna3 XOR dnaX4
dna1
dna1 - dnaX2
dna
BINSTR(X1) XOR DECODE(dna)
binPT
Scramble with Index Values of X1
ascii(binPT)
Plaintext (List of ASCII Vaues)

Image Encryption

The Image encryption algorithm has been made solely by us after taking a few references from the base paper (the DNA operations and Moore Machine).

- The image encryption algorithm involves Cipher Block Chaining, wherein the ciphertext output of one block is the input to the next block.
- The algorithm also uses the DNA operations and Moore Machine by implementing the DNA Encryption technique proposed in the paper as an intermediate form during the plaintext to ciphertext conversion.
- The algorithm uses the AES SBox, and the encrypted image is of the same size as the input image.
- A Novel scrambling algorithm has also been proposed by us in this implementation.

Key Generation and Retrieval

Algorithm 1: Key Generation Algorithm

Input: None

Output: [SR,y1,y2,y3],[IV, k1, k2, k3]

1 **GENERATE** 4 Random 32-bit numbers S4, y1, y2, y3

2 **EXECUTE** SCRAMBLE(SR,(y1 XOR y2 XOR y3)) to generate IV

3 **EXECUTE** SCRAMBLE(y1,(SR XOR y2 XOR y3)) to generate k1

4 **EXECUTE** SCRAMBLE(y2,(y1 XOR SR XOR y3)) to generate k2

5 **EXECUTE** SCRAMBLE(y3,(y1 XOR y2 XOR SR)) to generate k3

6 **RETURN**

Algorithm 3: Key Retrieval Algorithm

INPUT: SR, y1, y2, y3

OUTPUT: IV, k1, k2, k3

1 **EXECUTE** SCRAMBLE(SR,(y1 XOR y2 XOR y3)) to generate IV

2 **EXECUTE** SCRAMBLE(y1,(SR XOR y2 XOR y3)) to generate k1

3 **EXECUTE** SCRAMBLE(y2,(y1 XOR SR XOR y3)) to generate k2

4 **EXECUTE** SCRAMBLE(y3,(y1 XOR y2 XOR SR)) to generate k3

5 **RETURN**

Encryption and Encipher Block

Algorithm 2: Encryption Algorithm

Input: IV, k1, k2, k3, 4-channels Image
Output: Encrypted Image Array
1 **SAVE** Image shape into tshape
2 **CONVERT** Image Matrix to Vector
3 **CONVERT** image vector into binary string binimg
4 **SCRAMBLE** binimg with k1
5 **SAVE** length of binimg into l
6 **SPLIT** binimg into 32-bit blocks
7 **FOR** block in blocks
8 cblock = ENCIPHERBLOCK(IV, [k1,k2,k3], block)
9 IV = cblock
10 **APPEND** cblock to cblocks
11 **SCRAMBLE** cblocks with k3
12 **MAKEIMG** using cblocks and set shape = tshape
13 **STOP**

Algorithm 5: Encipher Block Algorithm

INPUT: IV, k1, k2, k3, block
OUTPUT: cblock
1 **EXOR** IV and block to get cblock
2 **SCRAMBLE** cblock with k3
3 **SBOX** cblock
4 **EXOR** cblock with k2
5 **CONVERT** cblock to DNA Sequence dna with DCR = IV
6 **CONVERT** k1 to DNA Sequence dnak1 with DCR = IV
7 **CONVERT** k2 to DNA Sequence dnak2 with DCR = IV
8 **CONVERT** k3 to DNA Sequence dnak3 with DCR = IV
9 **EXECUTE** DNA ADD dnak1 with dna to get dna1
10 **EXECUTE** DNA SUB dnak2 with dna1 to get dna2
11 **EXECUTE** DNA EXOR dnak3 with dna2 to get dna3
12 **INPUT** dna3 to MOORE Machine-generated using k2 to get dna4
13 **IF** k3 < 3
14 COMPLEMENT DNA bases in dna4 having an even index
15 **IF** k3 >= 3
16 COMPLEMENT DNA bases in dna4 having an odd index
17 **DECODE** dna4 to get binary string cblock
18 **RETURN**

Decryption and Decipher Block

Algorithm 4: Decryption Algorithm

INPUT: IV, k1, k2, k3 and Encrypted Image
OUTPUT: Decrypted Image
1 **SAVE** image shape in tshape
2 **CONVERT** Image Matrix to Vector
3 **CONVERT** Image Vector to binary string binimg
4 **SAVE** length on binimg in l
5 **UNSCRAMBLE** binimg with k3
6 **SPLIT** binimg into 32-bit blocks
7 **FOR** cblock in cblocks
8 block = ENCIPHERBLOCK(IV, [k1,k2,k3], block)
9 IV = cblock
10 **APPEND** block to blocks
11 **UNSCRAMBLE** blocks with k1
12 **MAKEIMG** with blocks and set shape = tshape
13 **STOP**

Algorithm 6: Decipher Block Algorithm

Input: IV, k1, k2, k3 and cblock
Output: block
1 **CONVERT** cblock to dna
2 **CONVERT** k1 to DNA Sequence dnak1
3 **CONVERT** k2 to DNA Sequence dnak2
4 **CONVERT** k3 to DNA Sequence dnak3
5 **CONVERT** cblock to DNA Sequence dna4
5 **IF** k3 < 3
6 COMPLEMENT DNA bases in dna4 having an even index
7 **IF** k3 >= 3
8 COMPLEMENT DNA bases in dna4 having an odd index
9 **INPUT** dna4 to MOORE MACHINE generated with k2 and get dna3
10 **EXECUTE** DNA EXOR dna3 with dnak3 to get dna2
11 **EXECUTE** DNA ADD dna2 with dnak2 to get dna1
12 **EXECUTE** DNA SUB dna1 with dnak1 to get dna
13 **DECODE** dna with DCR to get cblock
14 **ISBOX** cblock
15 **EXOR** cblock with k2
16 **UNSCRAMBLE** cblock with k3
16 **EXOR** cblock with IV to get block
17 **RETURN**

Analysis

- We conducted analysis on the image encryption algorithm using Mean Squared Error loss metric, to calculate the MSE between the original and decrypted image and also in a set of test case scenarios,that were as follows,

1. Removal of SRand all others intact.
2. Removal of y1 and all others intact.
3. Removal of y2 and all others intact.
4. Removal of y3 and all others intact.
5. Change MSB of SR
6. Change MSB of y1
7. Change MSB of y2
8. Change MSB of y3
9. Change LSB of SR
10. Change LSB of y1
11. Change LSB of y2
12. Change LSB of y3
13. Removal of keys SR,y1 and all other intact
14. Removal of keys SR,y2 and all other intact
15. Removal of keys SR,y3 and all other intact
16. Removal of keys y1.y2 and all other intact
17. Removal of keys y1,y3 and all other intact
18. Removal of keys y2,y3 and all others intact

Conclusion and Future Works

- The string encryption algorithm has been successfully implemented and demonstrated.
- The string encryption algorithm has successfully been extended to accommodate file encryption.
- The future works in this algorithm includes setting higher level of precision for the output of the hyperchaotic system to accommodate large input sizes.
- The image encryption algorithm has been successfully implemented and demonstrated.
- The image encryption algorithm was analysed using MSE similarity metric and its strength was proven.
- Minor imbalance in key contribution in the algorithm and a better algorithm can be made for key generation.

Thank You