Genetic Algorithm and Random number Generation for Symmetric Encryption

Ceira Sara Cherian[1], Dr. Rasmi P S[2]

[1]Asst. Prof, Department of Computer Science &Engineering,
Toc H Institute of Science and Technology, Arakunnam, Kerala, India
[2]HOD, Department of Computer Science &Engineering,
Toc H Institute of Science and Technology, Arakunnam, Kerala, India

Abstract- Data security is important for various day to day activities of humans. Personal and business applications require security of data. The amount of information that companies must keep secure is increasing. As a result of technological advances, companies are constantly gaining more data about their clients and customers. They must ensure that data security and privacy remain a priority to protect against costly breaches. Genetic Algorithm (GA) is a search-based optimization technique based on the principles of Genetics and Natural Selection. It is frequently used to find optimal or near-optimal solutions to difficult problems which otherwise would take a lifetime to solve. In this paper a symmetric encryption technique using Genetic Algorithm and pseudo random number generation is used to encrypt text files.

Keywords – Encryption, Decryption, Genetic algorithm, Random number generation, mutation, crossover.

I. INTRODUCTION

Genetic algorithms have been for the most part techniques applied by computer scientists and engineers to solve practical problems. The genetic algorithm is a method for solving optimization problems that is based on natural selection. The genetic algorithm modifies a population of individual solutionsrepeatedly. At each step, the genetic algorithm selects individuals at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution.

Cryptography converts data into a format that is unreadable for an intruder, allowing it to be transmitted without unauthorized decoding it back into a readable format, thus compromising the data. Information security uses cryptography on several levels. The information cannot be read without a key to decrypt it. The information maintains its integrity during transit and while being stored. Public-key cryptography, or asymmetric cryptography, is an encryption scheme that uses two mathematically related, but not identical, keys - a public key and a private key. Unlike symmetric key algorithms that rely on one key to both encrypt and decrypt, each key performs a unique function. The public key is used to encrypt and the private key is used to decrypt.

In this paper a new method is proposed for encryption which uses Genetic Algorithm for encryption and decryption. The new method proposed is applied to text data

II. PROPOSED ALGORITHM

2.1 Encryption algorithm using GA

Encryption Algorithm works as follows:

1. Read the text file to be encrypted and calculate the length of the string, l. If the length of string is not even then add a space to the end.
2. Convert each character of the string to 8 bit binary equivalent of its ASCII value. Each 8 bit value is considered as a block.
3. The blocks are paired in such format that the end blocks are paired. This process is repeated till the all blocks are paired.
4. Generate l/2 random numbers using Pseudo random generator algorithm. For each block a separate key value is used for encryption. Append the binary equivalent of the random number to the key file.
5. Generate the crossover point.
   r = Key % Length
   cp = r % 7
   cp is the crossover point. The bits after cp is exchange between the block pairs.
6. Generate the mutation point. 2 mutation points are selected for each block pair.
   mp1 = key % 7
   mp2 = length % 7
   The bits of mp1 and mp2 are flipped.

7. Convert the binary value to decimal. Convert the translated block pair to its character equivalent of ASCII value. Append the translated block into the encrypted file.
8. Repeat the crossover and mutation process for each block pair, using a different key value that is generated.

The flowchart showing the encryption process is shown in Figure 1

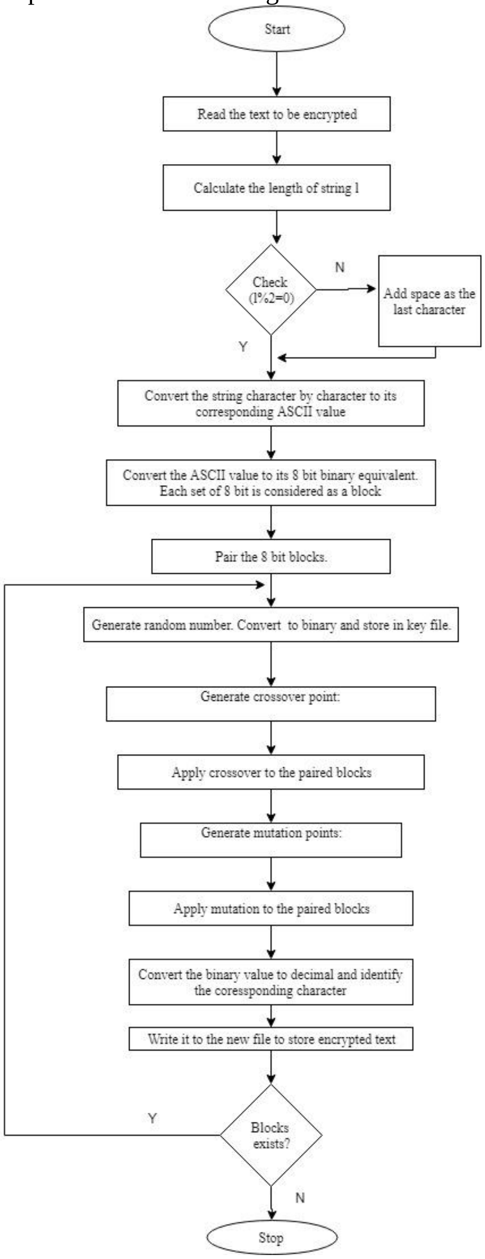

Start
Read the text to be encrypted
Calculate the length of string l
Check (l%2=0)
N
Add space as the last character
Y
Convert the string character by character to its corresponding ASCII value
Convert the ASCII value to its 8 bit binary equivalent. Each set of 8 bit is considered as a block
Pair the 8 bit blocks.
Generate random number. Convert to binary and store in key file.
Generate crossover point:
Apply crossover to the paired blocks
Generate mutation points:
Apply mutation to the paired blocks
Convert the binary value to decimal and identify the coressponding character
Write it to the new file to store encrypted text
Y
Blocks exists?
N
Stop

Figure 1: Flow chart for encryption

2.2 Decryption algorithm using GA

Decryption Algorithm works as follows:

1. Read the text file to be decrypted.
2. Convert each character of the string to 8 bit binary equivalent of its ASCII value.
3. The blocks are paired in such format that the adjacent blocks are paired. This process is repeated till the all blocks are paired.
4. Read the key value from the key file.
5. Generate the mutation point. 2 mutation points are selected for each block pair.

mp1 = key % 7
mp2 = length % 7
The bits of mp1 and mp2 are flipped.

6. Generate the crossover point.
   r = Key % Length
   cp = r % 7
   cp is the crossover point. The bits after cp is exchange between the block pairs
7. Convert the binary value to decimal. Convert the translated block pair to its character equivalent of ASCII value.
8. Append the first character of the pair to character in the left end. And place the second character to the left of the character in the right end.

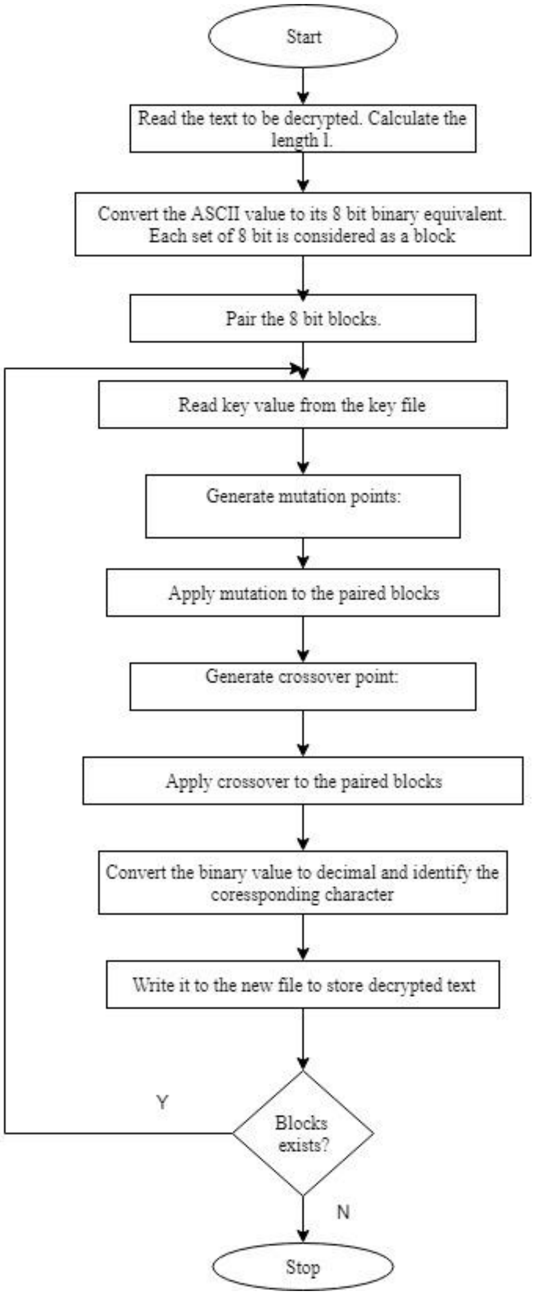
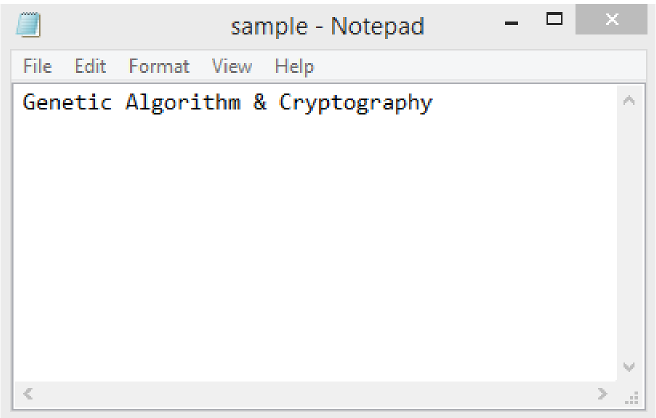

Start
Read the text to be decrypted. Calculate the length l.
Convert the ASCII value to its 8 bit binary equivalent. Each set of 8 bit is considered as a block
Pair the 8 bit blocks.
Read key value from the key file
Generate mutation points:
Apply mutation to the paired blocks
Generate crossover point:
Apply crossover to the paired blocks
Convert the binary value to decimal and identify the coressponding character
Write it to the new file to store decrypted text
Blocks exists?
Y
N
Stop

Figure 2: Flow chart for decryption

III. EXPERIMENT AND RESULT

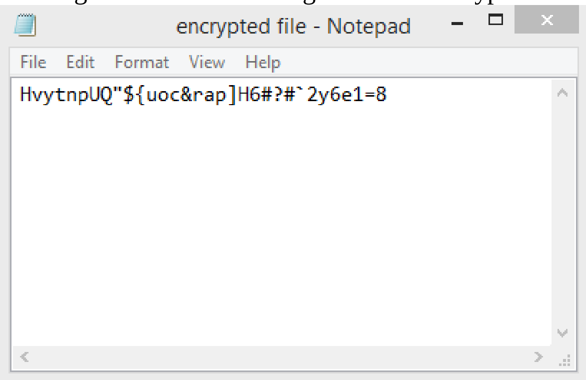
Files used in the program are:

File Name	Description
Sample.txt	Contains the text to be encrypted
Encrypted file.txt	Contains the encrypted text
Decrypted file.txt	Contains the encrypted text
Keys.txt	Contains randomly generated keys



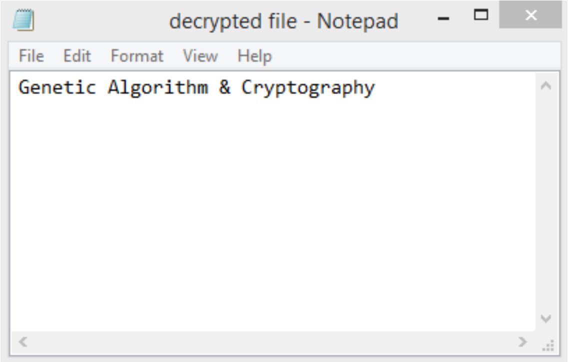
```
Genetic Algorithm & Cryptography
```

Figure 3: File containing text to be encrypted



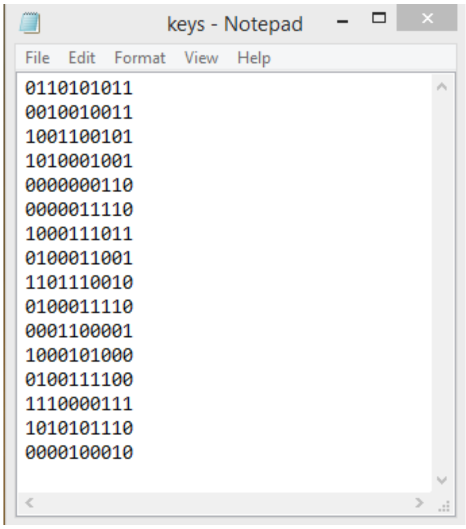
```
HvytnpUQ"${uoc&rap]H6#?#`2y6e1=8
```

Figure 4: File containing encrypted text



```
Genetic Algorithm & Cryptography
```

Figure 5: File containing decrypted text



```
0110101011
0010010011
1001100101
1010001001
0000000110
0000011110
1000111011
0100011001
1101110010
0100011110
0001100001
1000101000
0100111100
1110000111
1010101110
0000100010
```

Figure 6: File containing key values.

IV.CONCLUSION

In this paper we have proposed a symmetric algorithm for encryption and decryption of text file using genetic algorithm. The blocks of code were paired and crossover and mutation operators were applied. The crossover point and mutation points were identified based on random key values were generated. The algorithm is tested and results are produced.

The algorithm can be further improved by using different types of crossover methods based on the key generated. The pseudo random number generation can also be modified to get improved results.

V. REFERENCES

[1] Stallings, W.,Cryptography and network security principles and practice (5th ed.), Boston: Pearson, (2014).

[2] Deo, N., System simulation with digital computer, Prentice Hall of India Publications, (2011).

[3] SuvajitDutta andTanumay Das, "a Cryptography Algorithm Using the Operations of Genetic Algorithm & Pseudo Random Sequence Generating Functions",International Journal of Advances in Computer Science and Technology, Volume 3, (2014), ISSN 2320 – 2602.

[4] David E Goldberg, "Genetic algorithms in search, optimization and machine learning". (1989).

[5] Goyat, S.,"GENETIC KEY GENERATION FOR PUBLIC KEY CRYPTOGRAPHY", International Journal of Soft Computing and Engineering (IJSCE), Volume 2(3), (2012).

[6] L.M.R.J Lobo, Suhas B. Chavan, Use of Genetic Algorithm in Network Security, International Journal of Computer Applications (0975 8887)Volume 53– No.8, September 2012

[7] Dr. Poornima G. Naik, Girish R. Naik, Asymmetric Key Encryption using Genetic Algorithm, International Journal of Latest Trends in Engineering and Technology (IJLTET), Vol. 3 Issue 3 January 2014

[8] A Abdali Rashed, Using Modified Genetic Algorithm in Private Key Cryptosystems: Key Generation and Expansion

[9] Dilbag Singh, Pooja Rani, Rajesh Kumar, To Design a Genetic Algorithm for Cryptography to Enhance the Security, international journal of innovations in engineering and technology(IJIET) ,ISSN:2319-1058, Volume 2, Issue 2 ,2013,pp. 380-385.

[10] Farhat Ullah Khan, Surbhi Bhatia, A Novel Approach To Genetic Algorithm Based Cryptography, International Journal of Research in Computer Science (IJORCS), ISSN 2249-8265 Volume 2 Issue 3 ,2012,pp. 7-10.