ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 3, March 2017

Enhancing AES Algorithm Using Random Shuffle Method

Ashutosh Gupta, Vibhakar Mandal

B.Tech Final year Student, Dept. of Computer Science and Engineering, Institute of Technology, Guru Ghasidas Vishwavidyalaya, Koni, Bilaspur, Chhattisgarh, India

ABSTRACT: Today network security plays an important role in our life. It not only help us in providing a secure channel to our network , but also helps us in secure transaction for banking, shopping, sending email,filing tax returns, etc. And to provide a good secured network, we require a good encryption technique.
So in this advance encryption we are using randomised key for enhancing AES for security purpose. The input data to the encryption process i.e. Plain text (128 bit) is fed to the random shuffle method .The random shuffle method divides the obtained Plain text (128 bit) into blocks where each block contain 16 characters.Now according to the random number (one of 2^16 possible permutations)generated by Random Number Generator Method, each block is permuted and arranged and passed to the AES encryption process.

KEYWORDS: randkey, Mod Ciphertext, Random No Generator(), Random Shuffle(), Inv Random Shuffle(), Advance Encrypt(), Advance Decrypt().

I. INTRODUCTION

The Advance Encryption Standard (AES) was published by National Institute of Standards and Technology (NIST) in February 2001 to replace DES as the approved standard for a wide range of applications .The drawbacks of 3DES was that it is very slow and also it uses 64-bit block size same as DES. The two researchers who developed and submitted Rijndael for the AES were **Dr. Joan Daemen and Dr. Vincent Rijmen.** The algorithm is flexible in supporting any combination of data and key size of 128, 192, and 256 bits. The number of rounds in AES varies as given below:-

a) For AES-128 ,number of rounds = 10
b) For AES-192 ,number of rounds = 12
c) For AES-256 ,number of rounds = 14

Each round consist of following four steps in order:-

a) SubByte: It is also known as byte Substitution step.It is the first step in each round .Itis a non-linear byte Substitution, using a substitution table (S-box), which is constructed by multiplicative inverse and affine transformation. It provides nonlinearity and confusion in the cipher text.
b) ShiftRows: It is a simple byte transposition. There is no changes in the first row, but for the 2nd, 3rd and 4th row 1, 2 and 3 bytes are rotated respectively.
c) MixColumn: In this step, each column vactor is XORed by a fixed matrix.
d) AddRoundKey:The key generated for each round through key expansion process is XORed with the working states matrix.

The encryption starts with an AddRoundKey operation. After that 9 round are performed for 128 bit key. And in the 10th round MixColumn step is skipped.

II. RELATED WORK

In [1] authors have changed the form of plain text and encryption key given to AES algorithm.They have mapped input plain text and encryption key in various binary codes before applying as input to the AES algorithm. The performance is evaluated using avalanche effect. They found that when plain text and key both are mapped in 8421 binary code, maximum avalanche effect occurred. In [2] authors have modified each step involved in rounds of the AES. They claim to have improved the algorithm with strong diffusion and confusion by doing such modification. In [3] authors has added an extra stage named as Rotate S-box at the beginning if round function. The decryption process contain four stages,but the InvSubBytes operation is modified in a way to nullify the effect of Rotate S-box operation of encryption. This is followed by a description of key expansion and generation of shift offset-matrix. They have improved the security of AES by making its S-box key dependent. In [4] authors have proposed modification in which the bytes are rotated using 2-Dimensional rotation of D block after AddRoundKey. They have claimed that the proposed scheme will have improved complexity that increases the security. In [5] authors have presented the implementation of AES algorithm and also have studied the avalanche effect on AES they have concluded that AES is very strong cipher and impossible to break without knowing the key.

III. PROPOSED ALGORITHM

In this proposed encryption technique we are mainly using additional function apart from AES encryption Method and AES decryption method and these method are:-

1. RandomNoGenerator( ) Method :-RandomNoGenerator( ) Method is known as Random Number Generator which takes 1,2,3,…..16 number as arguments and generate a random number(one among 2^16 possible permutations) as a result .figure 3.1 describe the working of RandomNoGenarator( ) Method

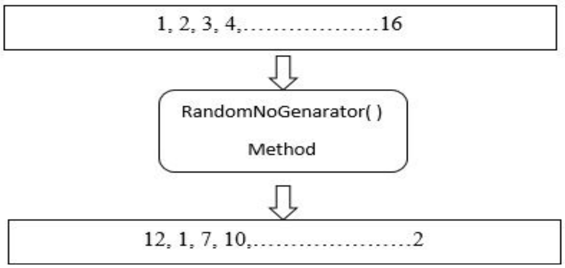


Fig 3.1: RandomNoGenarator( ) Method

2. RandomShuffle ( ) Method: -First of all we add padding to make the length of the plain text a multiple of 16. The padding is added in order to convert the text into complete block where each block comprises of 16 chracters.

Now using the RandomNoGenarator ( ) Method randkey is generated. This randkey is used to permute the plaintext using the RandomShuffle( ) Method. .figure 3.2 describe the working of RandomShuffle( ) Method

3.InvRandomShuffle ( ) Method: -This method is inverse of RandomShuffle( ) Method .The randkey used by RandomShuffle( ) Method to permute the data is used hare to de-permute the data. Figure 3.3 describe the working of InvRandomShuffle( ) Method.This method also uses the same argument as the RandomShuffle( ) Method. The only difference is that in place of plain text. Cipher text is passed as an argument here.

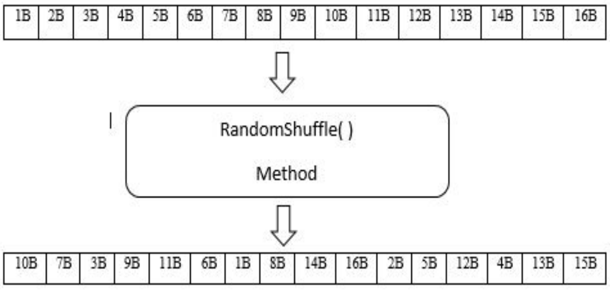

1B 2B 3B 4B 5B 6B 7B 8B 9B 10B 11B 12B 13B 14B 15B 16B
RandomShuffle( )
Method
10B 7B 3B 9B 11B 6B 1B 8B 14B 16B 2B 5B 12B 4B 13B 15B

Fig 3.3: RandomShuffle( ) Method

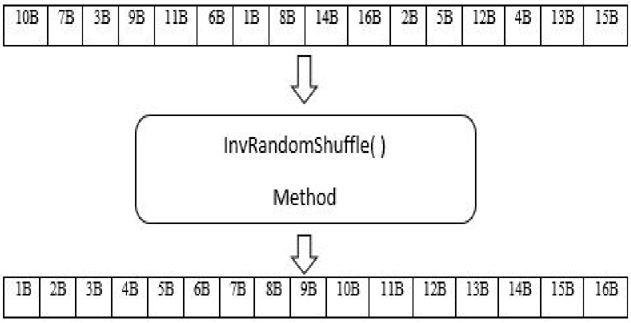

10B 7B 3B 9B 11B 6B 1B 8B 14B 16B 2B 5B 12B 4B 13B 15B
InvRandomShuffle( )
Method
1B 2B 3B 4B 5B 6B 7B 8B 9B 10B 11B 12B 13B 14B 15B 16B

Fig 3.3: InvRandomShuffle( ) Method

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 3, March 2017

4.AdvancedEcncrypt ( ) Method:-This method is hybrid of AES encryption method .Here the RandomShuffle( ) Method is called first .The plain text and the randkey (generated by RandomNoGenarator( ) Method is passed as the argument to this method. The RandomShuffle( ) Method returns the cipher text .This cipher text and the key(provided by the user) is then passed as the argument to normal AES Encryption Method .The output of this step is modified cipher text. .
5. AdvanceDecrypt () Method:The Cipher Text is passed to the Normal AES Decryption Method along with the key (provided by the user). After this step, the cipher text is obtained. This cipher text is then passed to the InvRandomShuffle ( ) Method. The randkey which is used in the AdvancedEncrypt () Method is also passed in InvRandomShuffle ( ) Method. At the end of this step, the Original Plain Text is obtained.
In proposed algorithm the plain text is passed to the RandomShuffle () method. RandomShuffle () Method uses randkey generated by RandomNoGenerator () Method. Using randkey the plain text is permuted for each block where 16 characters are present in a block. This permuted plain text along with the 128 bit key is passed to the original AES encryption method. The cipher text produced after this step is called modified cipher text and can be used for message passing. The randkey is also necessary to decrypt the data.
The decryption process is just the inverse of encryption process. First, the cipher text is decrypted using AES decryption method and then the decrypted text is passed to InvRandomShuffle () Method along with randkey.
Figure 3.4 describe the working of proposed algorithm.

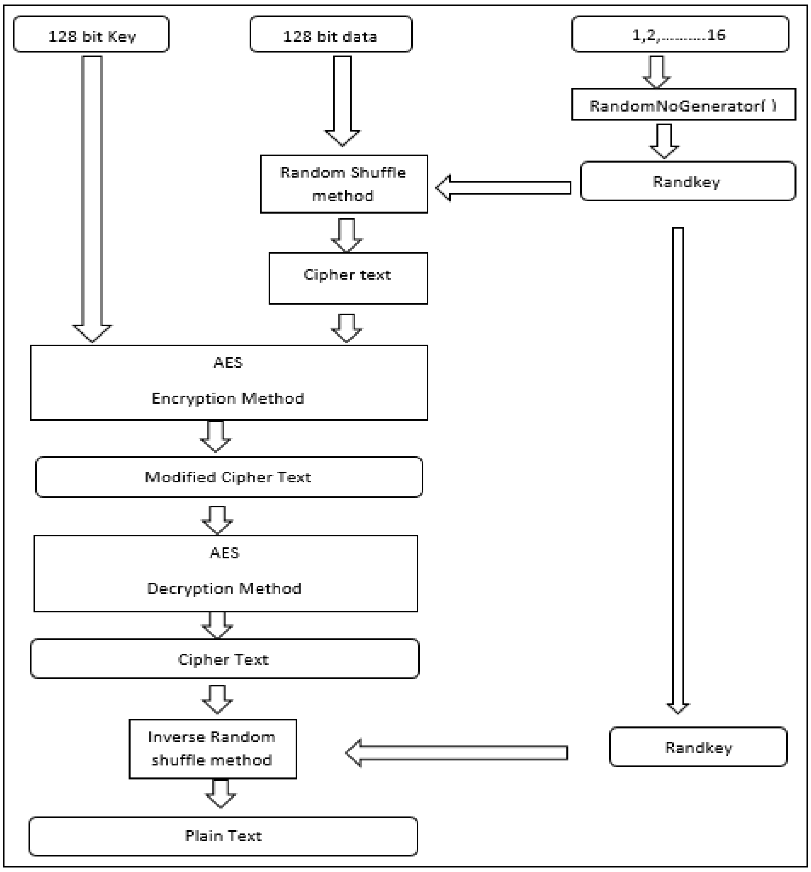


Fig 3.4: working of proposed algorithm.

DOI: 10.15680/IJIRCCE.2017. 0503040

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 3, March 2017

IV. PSEUDO CODE

```
AdvancedEncrypt(plaintext,key,randkey)
{
        ModCiphertext=RandomShuffle(plaintext,randkey)
        ciphertext =AES(ModCiphertext, key)  // normal AES Encryption
        return ciphertext
}


AdvanceDecrypt(ciphertext, key, randkey)
{
        Ciphertext=AES(ciphertext,key)  //normal AES
        plaintext =RandomShuffle(ciphertext, randkey) //normal AES Decryption
        return plaintext
}

RandomShuffle(plaintext, randkey)
{
    b is an array of char of length 16
    add padding to plaintext to make its length multiple of 16
    blocks = text.length / 16
    for i = 0 to blocks – 1
    {
        for k = 0 to 15
        {
                c = randkey.charAt(k)
                x = c – ‘A’
                b[k] = plaintext.charAt(i * 16 + x)
        }
        for k = 0 to 15
        {
         temp = temp + b[k]
        }
    }
    return temp
}

InvRandomShuffle(plaintext, randkey)
{
    b is an array of char of length 16
    blocks = ciphertext.length / 16
    for i = 0 to blocks – 1
    {
        for k = 0 to 15
        {
           c = randkey.charAt(k)
           x = c – ‘A’
           b[x] = plaintext.charAt(i * 16 + k)
        }
        for k = 0 to 15
```

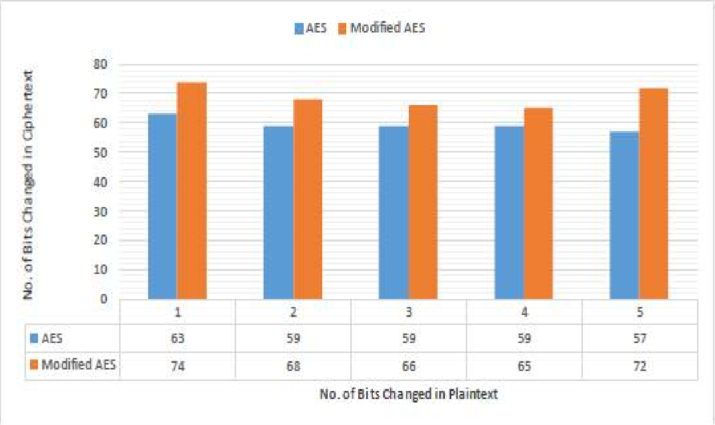
```
            {
             ciphertext [i * 16 + k] = b[k]
            }
        }
        return ciphertext
    }

    ModifiedAES()
    {
        user input = plain text and key
        randkey = RandomNoGenerator( )
        ciphertext = AdvanceEncrypt(plaintext, key, randkey) //to encrypt message
        plaintext = AdvanceDecrypt(ciphertext, key, randkey) //to decrypt message
    }
```

V. SIMULATION RESULTS

The simulation studies involve the study of the Avalanche Effect on the proposed Algorithm. An enviable property of any encryption algorithm is that for a small change in either plaintext or key should result in a significant change in cipher text [5]. This property is known as the Avalanche Effect.

We have implemented the proposed algorithm using Java. Input to the algorithm is 128 bit plain text and 128 bit encryption key. After implementation, Avalanche effect of both the AES and the proposed algorithm is calculated. Figure 5.1 and Figure 5.2 shows the Avalanche effect due 1, 2, 3, 4 and 5 bit variation in plaintext "GKAHEFOIRYNPEMAC" and "RGEDIJOCAWXESIGA" respectively, keeping the key "0f1571c947d9e859" constant in both figures.Figure 5.3 and Figure 5.4 shows the Avalanche effect due to 1, 2, 3, 4 and 5 bit variation in AES Key "KHIROPVYMEIOUXZQ" and "VRCGOMQWSXPALFTC" respectively, keepingthe plaintext "12468aceeca86420" constant in both figures.

The randkey used in Figure 5.1, Figure 5.2, Figure 5.3 and Figure 5.4 is "EAICONMDLKHBFJGP", "ECNDMBKAJOIPGHFL", "JNBDGFKLMOPCHEAI" and "ECNDMBKAJOIPGHFL" respectively. The randkey is kept constant in all the simulations.The table in all the figures is showing the values of avalanche effect in AES and Modified AES. It is clear from all the figures that the Modified AES is producing better avalanche effect than AES.



	1	2	3	4	5
AES	63	59	59	59	57
Modified AES	74	68	66	65	72

Figure 5.1: Avalanche Effect on changing plaintext "GKAHEFOIRYNPEMAC"

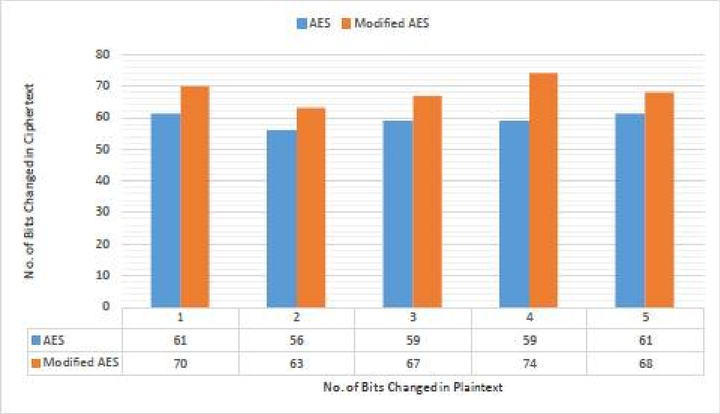


Figure 5.2: Avalanche Effect on changing plaintext “RGEDIJOCAWXESIGA”

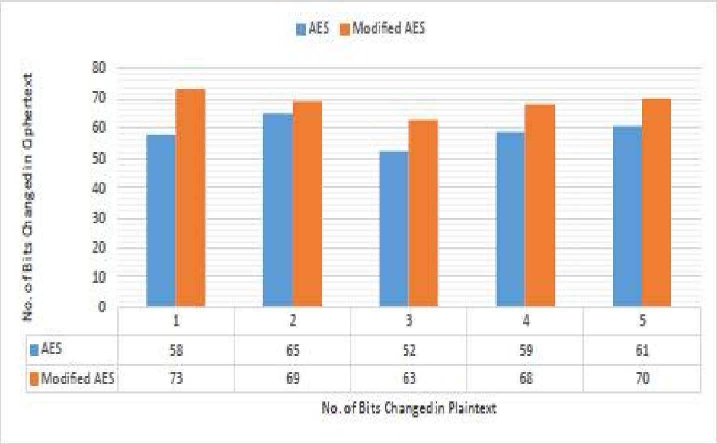


Figure 5.3: Avalanche Effect on changing key “KHIROPVYMEIOUXZQ”

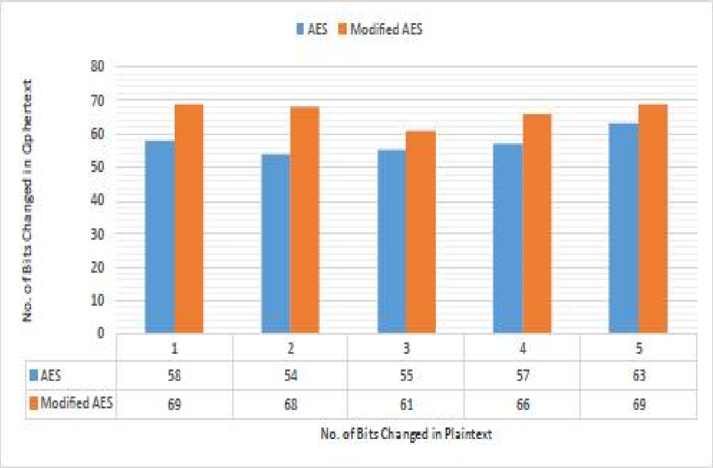


Figure 5.4: Avalanche Effect on changing key “VRCGOMQWSXPALFTC”

VI. CONCLUSION AND FUTURE WORK

The simulation results showed that the proposed algorithm has much more avalanche effect than the existing AES.Avalanche effect due to change in 1, 2, 3, 4, and 5 bits of plain text resulted in much better the existing AES. Also, when the same process was applied with the change in key bits, it also resulted that the proposed algorithm function better. Therefore, it increases the complexity of the algorithm. It helps to encrypt the data by making strong diffusion and confusion.

Our future work will include the pipeline operation of the proposed algorithm to reduce the effective running time of proposed algorithm and space complexity. Our focus will also be to increase the security level of proposed algorithm.

REFERENCES

1. Chandra Prakash Dewangan and Shashikant Agrawal , “A Novel Approach to Improve Avalanche Effect of AES Algorithm”, International Journal of Advanced Research in Computer Engineering& Technology, Volume 1,Issue 8, October 2012
2. Priyanka Pimpale, Rohan Rayarikar and Sanket Upadhyay, “Modifications to AES Algorithm for Complex Encryption”, IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.10, October 2011
3. Krishnamurthy G N, V Ramaswamy, “Making AES Stronger: AES with Key Dependent S-Box”, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.9, September 2008
4. Pushpa R. Suri, Sukhvinder Singh Deora, “Design of a Modified Rijndael Algorithm Using 2D Rotations”, IJCSNS International Journal of Computer Science and Network Security, Vol .11 No.9, September 2011
5. Amish Kumar and Mrs. Namita Tiwari, “Effective Implementation And Avalanche Effect of AES”, International Journal of Security, Privacy and Trust Management (IJSPTM), Vol. 1, No ¾, August 2012
6. Nidhi Singhal and J.P.S.Raina,“Comparative Analysis of AES and RC4 Algorithms for Better Utilization”, International Journal of ComputerTrends and Technology- July to Aug Issue 2011
7. Himani Agrawal and Monisha Sharma, “Implementation and analysis of various symmetric cryptosystems”, Indian Journal of Science andTechnology Vol. 3 No. 12 (Dec 2010)
8. A.Nadeem, “A performance comparison of data encryption algorithms”, IEEE information and communication technologies,pp.84-89, 2006.Bn
9. Ahmed Bashir Abugharsa,Abd Samad Bin Hasan Basari,Hamid Almangush, “A new encryption approach Using the integration of a shifting Technique and the AES algorithm”, International Journal ofComputer Application,Volume42-No.9 March 2012
10. Diaa Salama Abd Elminaam, Hatem Mohamad Adbual Kader, Mohiy Mohamed Hadhoud, “Evaluation of the Performance of Symmetric Encryption Algorithms”, International Journal of Network Security, Volume 10, No. 3, pp. 216-222, May 2010
11. Kazlauskas, Kazys, and Jaunius Kazlauskas, "Key-dependent S-box generation in AES block cipher system”, Informatica 20.1 (2009): 23-34.
12. Chaudhary Saifurrab and Saqlain Mirza, “AES algorithm using advance key implementation in MATLAB”, International Research Journal of Engineering and Technology, Volume: 03 Issue: 09, Sep -2016

BIOGRAPHY

Ashutosh Guptais currently pursuing his B.Tech in Computer Science and Engineering at Guru Ghasidas Vishwavidyalaya, Bilaspur, Chhattisgarh, India, 495009.

Vibhakar Mandalis currently pursuing his B.Tech in Computer Science and Engineering at Guru Ghasidas Vishwavidyalaya, Bilaspur, Chhattisgarh, India, 495009.

DOI: 10.15680/IJIRCCE.2017. 0503040