Designing a Secure Scheme using AES For Key Expansion By Using The Fuzzy Sets

Ankita Sachdeva, Rajiv Kumar Nath
Dept of Computer Science, Galgotia's college Greater Noida

Abstract - This paper presents a modified scheme for the key expansion module of Advance Encryption Standard (AES). Depending upon the frequencies of input characters in the message, this key calculates the indices for various categories. These indices are used by a Fuzzy system to decide whether normal key expansion will be used or modified key expansion will be used in aes encryption. A new technique for finding the factors used in the modified key expansion algorithm is presented in this paper ,which is based on FFT .the key expansion algorithm uses the factor for generating the key of each round of aes encryption. Sample output of various modules are also presented in the paper. A brief overview of AES is also provided

Keywords: Modified key expansion ,count frequency module, fuzzy sets, FFt scheme, AES

I. INTRODUCTION

In this paper we focus on the key expansion module in AES using the fuzzy sets. An AES is a symmetric block cipher that was replacement of DES(Data Encryption Standard). An AES uses the block size of 128 bits also called the plaintext, and key size of 128 bits, 192 bits and 256 bits. It uses 10 rounds for 128 bit key size ,12 rounds for 192 bit key size and 14 rounds for 256 bits key size. Each round uses a round key derived from original key .This key is generated in key Expansion using the fuzzy sets.

In this it comprise these four basic steps for each round but the final round mixing is not done.
Each round in AES there is a structure i.e given by Rijndael. So there are four basic rounds :

1. Byte Substitution : This is used non linear structure i.e differential and cryptanalytic attacks.
2. Shift Row Transformation: In this we do the permutation on bytes.
3. Mix Columns: In this do some diffusion of bits over multiple rounds. This step is not done on the last round.
4. Add Round key: In this the ex-or of bits is performed on the previous value

In this paper we find the modifying factor by using the fuzzy sets. Fuzzy sets is a problem-solving control system methodology that lends itself to implementation in systems ranging from simple, small, embedded micro-controllers to large, networked, multi-channel pc (or) workstation-based data acquisition and control systems. It can be implemented in hardware, software (or) a combination of both. Fuzzy logic provides a simple way to arrive at a definite conclusion based upon vague, ambiguous, imprecise, and noisy (or) missing input information. Fuzzy set and Fuzzy logic is the powerful technique for modelling and controlling the uncertain systems. Fuzzy sets are different from the crisp set because In crisp sets can have only two possible values either 0 or 1. 0 implies it is not a member of fuzzy set and 1 implies the member of fuzzy set. While in fuzzy set every member is assigned a membership value between 0 and 1. Fuzzy sets are used in many engineering and industrial applications. In this we first take an input text i.e classify into seven different categories on the basis of frequency e.g Type A- 2-7 Then use this information to generate a modifying factor and tis modifying factor further use to expand key routine of AES. Then Encrypt using AES and sends the message .On the other hand Reciever's side Decryption of message is performed.

Network Security is become more and more crucial as the huge amount of data being exchanged on the internet access [1]. Based on these, the security involves four important parameters: Confidentiality, message authentication, integrity and non – repudiation. In these parameters we check the data is confidential or not. The message is sent by sender is altered by attacker and attacker removes the authenticity of the message. so we make the scheme more encrypted so the attackers cant breaks that easily. So in this we use the key expansion by fuzzy so that improve the authenticity of the message. In cryptography, public-key cryptosystems are convenient in that they do not require the sender and receiver to share a common secret in order to communicate securely [2]. However, they often rely on complicated mathematical computations and are thus generally much more inefficient than comparable symmetric-key cryptosystems.

Fuzzy Logic Approach

Fuzzy Logic is a simple but strong methodology in logic building. Fuzzy set theory is an extension of classical set theory where elements have different degrees of membership. A logic based on the two truth values True and False is sometimes inadequate when describing

human reasoning [2]. Fuzzy logic uses thewhole interval between 0 (false) and 1 (True) to describe human reasoning. A fuzzy set is any set that allows its members to have different degree of membership function in the interval [0,1].The degree of membership (or) truth is not same as probability[3].Fuzzy Sets are introduce by LOTFIZADEH in 1965 Berkley university of California Now a days it has been used in many Engineering and Industrial application[6].

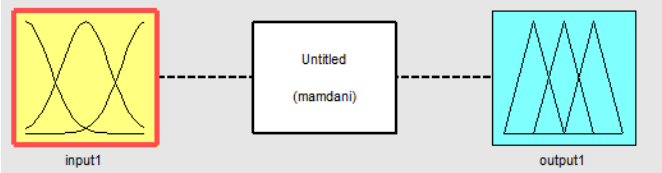

Fig1: Basic Of Fuzzy Logic

II. PREVIOUS WORK

In On March 17,1975, the United States of government proposed that the Data Encryption Standard (DES),as originally specified in FIPS-42.The DES using the 56 bit key that was too small length, so 2 56 unique keys only made so, for attacker its very Simple to break this key. Then the First Public crack of the DES cryptosystem occurred in 1997 by Rockie Verser. The concept of triple DES introduced but there is also some certain limitations So, triple DES was accepted as a temporary replacement to DES until a replacement is establish through the AES development process[3].

In August 1998 the first AES conference held in Ventura, California, NIST. In march of 1999 a second conference was held in Rome. In October 2000 NIST announced that it had selected Rijndael as the algorithm for AES is the development Project. Rijndael decide the key length can be defined as 128, 192 and 256 bits.[4]

III. PROPOSED METHODOLOGY

.Encryption Algorithm

At, the start of encryption in0, in1,….in15 is copied into the state array of 4*4 matrix . These are stated as $S_{0,0}$,$S_{0,1}$…$S_{3,3}$.In this we work on the Galois field GF($2^{8)}$. The elements in the Galois field are in bytes.

$$\begin{vmatrix} a_{0,0} & a_{0,1} & a_{0,2} & a_{0,3} \\ a_{1,0} & a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,0} & a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,0} & a_{3,1} & a_{3,2} & a_{3,3} \end{vmatrix}$$

In the initial round these five operations are performed:

1. Sub bytes
2. Shift Rows
3. Mixing Columns
4. Key Expansion
5. Add Round Key

The First nine rounds these operations are performed but in the Final round Mix Column transformation is not used.

Details of Single Round

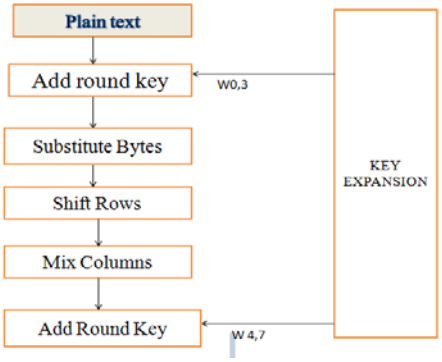

Fig2. Single round details

1 Sub Byte Transformation

Each byte of 4*4 input matrix is substituted by another byte using a substitution table called the s-box. This module takes a Hex value as input It finds the index I from this hex value which is used for row number in the substitution table. Also it finds the index j from the hex value which is used for column number in the substitution table. This substitution also forms the output of 4*4 matrix is given as:

$$\begin{vmatrix} b_{0,0} & b_{0,1} & b_{0,2} & b_{0,3} \\ b_{1,0} & b_{1,1} & b_{1,2} & b_{1,3} \\ b_{2,0} & b_{2,1} & b_{2,2} & b_{2,3} \\ b_{3,0} & b_{3,1} & b_{3,2} & b_{3,3} \end{vmatrix}$$

2 Shift rows Transformation

In this operation the bytes are shifted circular left one by one . In first row there is 0 shift .In second row shifted all values by 1 and so on.

$$\begin{vmatrix} c_{0,0} & c_{0,1} & c_{0,2} & c_{0,3} \\ c_{1,0} & c_{1,1} & c_{1,2} & c_{1,3} \\ c_{2,0} & c_{2,1} & c_{2,2} & c_{2,3} \\ c_{3,0} & c_{3,1} & c_{3,2} & c_{3,3} \end{vmatrix} =$$

$$\begin{vmatrix} x_{0,0} & x_{0,1} & x_{0,2} & x_{0,3} \\ x_{1,1} & x_{1,2} & x_{1,3} & x_{1,0} \\ x_{2,2} & x_{2,3} & x_{2,0} & x_{2,1} \\ x_{3,3} & x_{3,0} & x_{3,1} & x_{3,2} \end{vmatrix}$$

3.Mixing Columns

In this take the output of the previous step in Galois Field GF(2^8). This is multiplied by another 4*4 matrix modulo of a degree 4 polynomial i.e x^4+1, So the term is: $a(x)=\{03\}x^3+\{01\}x^2+\{01\}x+\{02\}$

The matrix multiplication is given as:

$$\begin{vmatrix} 0*02 & 0*03 & 0*01 & 0*01 \\ 0*01 & 0*02 & 0*03 & 0*01 \\ 0*01 & 0*01 & 0*02 & 0*03 \\ 0*03 & 0*01 & 0*01 & 0*02 \end{vmatrix} \text{ ex-or } \begin{vmatrix} c_{0,0} & c_{0,1} & c_{0,2} & c_{0,3} \\ c_{1,0} & c_{1,1} & c_{1,2} & c_{1,3} \\ c_{2,0} & c_{2,1} & c_{2,2} & c_{2,3} \\ c_{3,0} & c_{3,1} & c_{3,2} & c_{3,3} \end{vmatrix}$$

$$= \begin{vmatrix} d_{0,0} & d_{0,1} & d_{0,2} & d_{0,3} \\ d_{1,0} & d_{1,1} & d_{1,2} & d_{1,3} \\ d_{2,0} & d_{2,1} & d_{2,2} & d_{2,3} \\ d_{3,0} & d_{3,1} & d_{3,2} & d_{3,3} \end{vmatrix}$$

4.Key Expansion

In this we first count the frequency to find the mean value and standard deviation on the basis we categorized the frequency. Then apply the fuzzy logic and then make the rules:

S.N	Index1	Index2	Index3	Index4	Index5	S.D
1	large	Large	large	large	large	Very high
2	medium	Medium	medium	medium	medium	high
3	small	Small	small	medium	medium	high
4	small	Small	small	small	small	small

Table1:Rules set

On the basis of these rules we decide in which set it lies whether normal expansion or modified expansion. And then apply this algorithm:

Expanded key for each round

y [1:44]

for round 0 = y [1:4]

for round 1= y[5:8]

.....

for round 10= y[41:44]

for i=1:Nk

Assign the values of y from input parameter x as given below:

y(1,i) = x(4*(i-1)+1)

y(1,i) = x(4*(i-1)+1)

y(1,i) = x(4*(i-1)+1)

y(1,i) = x(4*(i-1)+1)

end of for statement

for i=5:44

t1=y(1,(i-1));

t2=y(2,(i-1));

t3=y(3,(i-1));

t4=y(4,(i-1));

if (modulus(i, Nk) == 0) then

calculate rcons as given below

rcons(4)=0

rcons(3)=0

rcons(2)=0

rcons(1) = 2 raise to (i/Nk)

if rcons(1)== '0100' then rcons(1)='1b'

if rcons(1) == '0200' then rcons(1) = '1b' , shift rcons(1) left by 1 position.

```
  /* rotate word  */
   t0=t1
    t1=t2
    t2=t3
    t3=t4
    t4=t0
/* substitute word from s box
     t1=sub word(t1)
     t2=sub word(t2)
     t3=sub word(t3)
     t4=sub word(t4  /*  Modified steps in key expansion
    t1= t1 or '70'
    t2= t2 or '50'
    t3= t3 or '30'
    t4= t4 or '10'
/* Other steps being same as in original key expansion
   t1 = t1 xor rcons(1)
   t2=  t2 xor rcons(2)
   t3 = t3 xor rcons(3)
   t4=  t4 xor rcons(4)
 end if
else if (N k>6) and (modulus(i,Nk)==4) then
     t1=sub word(t1)
     t2=sub word(t2)
     t3=sub word(t3)
     t4=sub word(t4)
 end if
y(1,i)= y(1,i-Nk) xor t1
 y(2,i)= y(2,i-Nk) xor t2
 y(3,i)= y(3,i-Nk) xor t3
 y(4,i)= y(4,i-Nk) xor t4
end of for
```

5.Add Round Key

In the round key it is derived from key expansion using that algorithm. This key is arranged in 4*4 matrix[Ki,j] is ex-or with the output of Mix column step Mathematically:

$$\begin{vmatrix} d_{0,0} & d_{0,1} & d_{0,2} & d_{0,3} \\ d_{1,0} & d_{1,1} & d_{1,1} & d_{1,3} \\ d_{2,0} & d_{2,1} & d_{2,2} & d_{2,3} \\ d_{3,0} & d_{3,1} & d_{3,2} & d_{3,3} \end{vmatrix} \text{ ex-or } \begin{vmatrix} k_{0,0} & k_{0,1} & k_{0,2} & k_{0,3} \\ k_{1,0} & k_{1,1} & k_{1,2} & k_{1,3} \\ k_{2,0} & k_{2,1} & k_{2,2} & k_{2,3} \\ k_{3,0} & k_{3,1} & k_{3,2} & k_{3,3} \end{vmatrix} = \begin{vmatrix} e_{0,0} & e_{0,1} & e_{0,2} & e_{0,3} \\ e_{1,0} & e_{1,1} & e_{1,2} & e_{1,3} \\ e_{2,0} & e_{2,1} & e_{2,2} & e_{2,3} \\ e_{3,0} & e_{3,1} & e_{3,2} & e_{3,3} \end{vmatrix}$$

DECRYPTION

Each of the steps Sub Byte, Shift Row ,Mix Column and Add Round Key are invertible:

1.The inverse of sub byte transformation is given by another table called inverse of sub byte .
2.The inverse of shift row is obtained by shifting of bytes in the rows to the right instead of left shift.
3.The inverse of mix column is obtained by multiplying 4*4 matrix with inverse of the 4*4 matrix (when we use mix columns) modulo $x^4 + 1$.Thus the multiplying matrix is:

$$\begin{vmatrix} 0*0E & 0*0B & 0*0D & 0*09 \\ 0*09 & 0*0E & 0*0B & 0*0D \\ 0*0D & 0*09 & 0*0E & 0*0B \\ 0*0B & 0*0D & 0*09 & 0*0E \end{vmatrix} =$$

4.Add round key is inverse of itself.

5.At the time of decryption the receiver knows the private Key.

IV. RESULTS ANALYSIS

In the modified algorithm we use the Fast Fourier Transform(FFT). Taking output of these parameters we calculated the frequency and find the standard deviation. By this we know that either normal or modified expansion is used. Then after performing these mentioned operations we get the encrypted text called cipher text. Again we apply the inverse then we obtained the plain text.

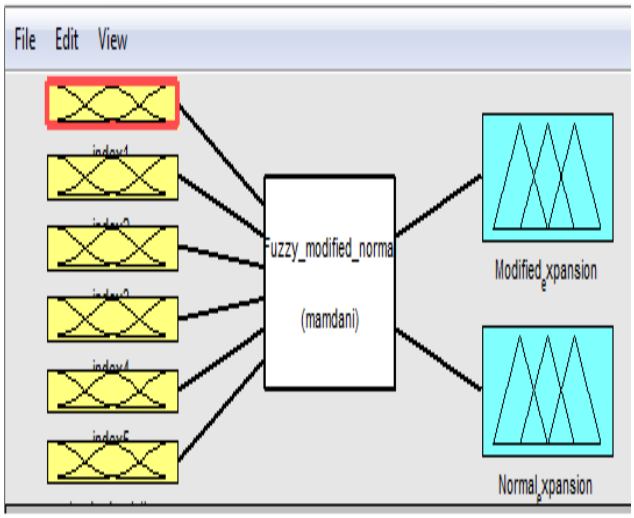

Fig2: This shows parameters are used to calculate normal or modified expansion.

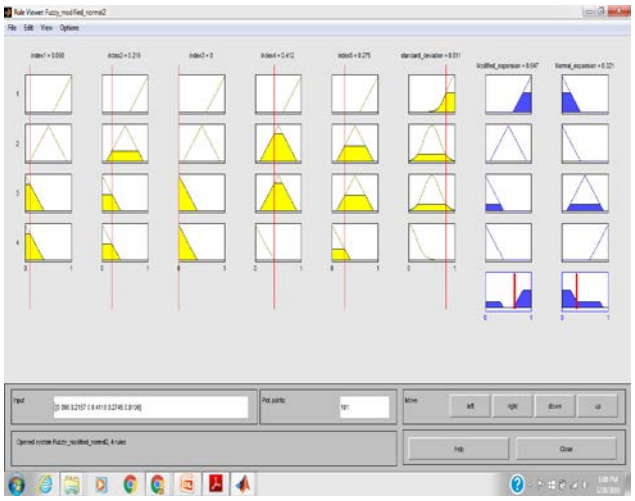

Fig2. This shows the output of rules of normal and modified expansion through which calculate the mean and standard deviation

V. CONCLUSION

In this paper a symmetric cryptology is introduced. This algorithm works as a modify algorithm of Rijndael concept .In this we use the fuzzy logic to make more secure the data so ,that we protect our message from the intruders.

VI. FUTURE SCOPES

In this we also use the other mathematical scheme like FFT to make the secure algorithm and We also use the other ideas like Fuzzy.

REFERENCES

[1] "Advanced Encryption Standard (AES)", Federal Information Processing Standards Publication 197, November 26, 2001.

[2] Dr. Brian Gladman, Rijndeal (by Joan Daeman & vince VincentRijmen), "A Specification for the AES Algorithm", 15 April 2003.

[3] Shafi Golgwasser Mihir Bellare, "Lecture Notes on Cryptography", July 2008.

[4] William Stallings, "Cryptography and Network Security", Fourth Edition, June 3, 2010"U.S.Loses Focus onFuzzy Logic" (Machine Design,June 221, 1990).

[5] Fuzzy Sets and Applications: Selected Paprsby L.A Zadeh", ed. R.R Yager et al. (John Wiley, New York, 1987).