[NC-Rase 18]
DOI: 10.5281/zenodo.1483995

ISSN 2348 – 8034
Impact Factor- 5.070

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

PERFORMANCE ANALYSIS OF AES AND BLOWFISH CRYPTOGRAPHIC ALGORITHMS

Dr. Harshala B. Pethe[*1] ,Varsha C. Pande[2] & Dr. Subhash R. Pande[3]

[*1]Assistant Professor, Department of Computer Science, Dr. Ambedkar Collage, Nagpur(MH), [2]Lecturer SSESA's, Science College, Congress Nagar, Nagpur (MH),

[3]Associate Professor and Head, Department of Computer Science, Science College, Nagpur(MH).

ABSTRACT

Information security plays very important role in storing and transmitting the data through unsecured channel. Cryptography plays a very important role in the network security to maintain the CIA triad that is Confidentiality, Integrity, Authentication and non-repudiation of information. Due to that security of information is much important in data storage and transmission process. Using cryptography, the data is encoded before sending it and decoded after receiving, Cryptographic algorithms are broadly divided into two types, symmetric key and asymmetric key cryptographic algorithms.There are various symmetric key algorithms available such as DES, AES Blowfish, Two fish, SAFER etc. This paper deals with the performance analysis of symmetric key cryptographic algorithms AES and Blowfish.

Keywords: *Cryptography, AES, Blowfish, Symmetric key cryptography.*

I. INTRODUCTION

Cryptography is a technique which is used to secure transmitting information. The process includes encryption and decryption. Secret key is used in the process to convert the plaintext into encrypted format [1]. Symmetric key and asymmetric key cryptography are the two classifications of cryptography.

Various symmetric key cryptographic algorithms are available. In this paper we have considered only AES and Blowfish algorithms.

AES operates on a 128 bit data block at a time and uses 128, 192 or 256 bits key length and uses 10, 12 or 14 rounds. If both block length and key length are 128 bits, AES will perform 9 processing rounds. If the block and key are 192 bits, AES will perform 11 processing rounds. If the block and key are 256 bits [3], then it performs 13 processing rounds. Each processing rounds involves four steps.

Blowfish is a symmetric 64-bit block cipher invented by Bruce Schneier [4]; optimized for 32- bit processors with large data caches, it is significantly faster than DES on a Pentium or Power PC-class machine. Key lengths can vary from 32 to 448 bits in length [5]. Blowfish, available freely and intended as a substitute for IDEA, is in use in over 80 products. Blowfish is an algorithm of my own design, intended for implementation on large microprocessors. The algorithm is unpatented.

Blowfish is designed to meet the following design criteria.
1. **Fast.** Blowfish encrypts data on 32-bit microprocessors at a rate of 26 clock cycles per byte.
2. **Compact.** Blowfish can run in less than 5K of memory.
3. **Simple.** Blowfish uses only simple operations: addition, XORs, and table lookups on 32-bit operands. Its design is easy to analyze which makes it resistant to implementation errors.
4. **Variably Secure.** Blowfish's key length is variable and can be as long as 448 bits. Blowfish is optimized for applications where the key does not change often, like a communications link or an automatic file encryptor. It is significantly faster than DES when implemented on 32-bit microprocessors with large data caches, such as the Pentium and the PowerPC. Blowfish is not suitable for applications, such as packet switching, with frequent key

changes, or as a one-way hash function. Its large memory requirement makes it infeasible for smart card applications [6].

II. SYMMETRIC KEY ALGORITHMS

The following are the symmetric key cryptographic algorithm used in this paper.

Advanced Encryption Standard Algorithm
A data block is partitioned into an array of bytes. Such bytes are interpreted as a finite field elements using polynomial representation. The input is divided into 16 bytes and then arranged into a 4x4 matrix column wise [7]. This matrix is known as the state matrix. The original 128-bit key is also divided in to 16 bytes as like 128 bit data and arranged in the form of 4x4 matrixes. This matrix is called keyMatrix.

Both these matrices form the necessary inputs to the algorithm.
AES encryption includes,

1) An initial round (0)
2) Nine general rounds (1 to 9) and
3) A final round (10)

In round(0) the two matrices are simply XORed under AddRoundKey transformation. The output of Round0 is given as the input to Round 1. Each round composed of four distinct, uniform and invertible transformations: Subbytes, ShiftRows, MixColumn and AddRoundKey[8].

i) Subbytes
This stage also known as Substitute Bytes, it is simply a table lookup using a 16x16 matrix of byte values called s-box. This matrix consists of all the possible combinations of an 8 bit sequence (2^8 =16x16 = 256). It is a non-linear byte substitution operation designed to give the required amount of Confusion. It operates independently on each byte of the State matrix using a substitution table S-box. Each byte is substituted by corresponding byte in the S-box.

ii) ShiftRows
This stage is also known as Shift Row Transformation. It works as follows:

1) The first row of state is not altered.
2) The second row is shifted 1 bytes to the left in a circular manner.
3) The third row is shifted 2 bytes to the left in circular manner.
4) The fourth row is shifted 3 bytes to the left in a circular manner

It is a transposition step that gives the required amount of Inter – word Diffusion and operates individually on each of the last three rows of state matrix shifting cyclically a certain number of bytes. The first row is left unchanged. The second row is left rotated by one byte, third row by two bytes and fourth row by three bytes as shown in fig 1.

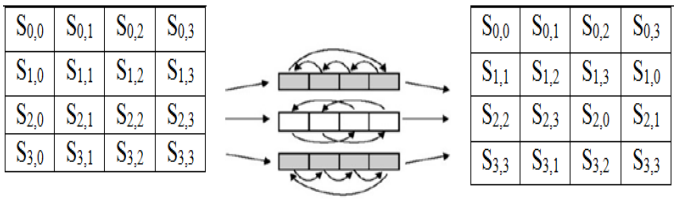


Figure 1 : ShiftRows stage

iii) MixColumn
This operation gives intra-word Diffusion and operates on each column of the state matrix individually, combining the four bytes in each column using multiplications and additions in GF (2^8). Each byte of a column is mapped into a

[NC-Rase 18] ISSN 2348 – 8034
DOI: 10.5281/zenodo.1483995 Impact Factor- 5.070

new value that is a function of all four bytes in the column. The transformation can be determined by the following matrix multiplication on state.

The MixColumn transformation of a single column j (0<=j<=3) of state can be expressed as follows:

$$\begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} S_{0,0} & S_{0,1} & S_{0,2} & S_{0,3} \\ S_{1,0} & S_{1,1} & S_{1,2} & S_{1,3} \\ S_{2,0} & S_{2,1} & S_{2,2} & S_{2,3} \\ S_{3,0} & S_{3,1} & S_{3,2} & S_{3,3} \end{pmatrix} = \begin{pmatrix} S'_{0,0} & S'_{0,1} & S'_{0,2} & S'_{0,3} \\ S'_{1,0} & S'_{1,1} & S'_{1,2} & S'_{1,3} \\ S'_{2,0} & S'_{2,1} & S'_{2,2} & S'_{2,3} \\ S'_{3,0} & S'_{3,1} & S'_{3,2} & S'_{3,3} \end{pmatrix}$$

$$S'0,j = (2 \bullet S0,j) \oplus (3 \bullet S1,j) \oplus S2,j \oplus S3,j$$
$$S'1,j = S0,j \oplus (2 \bullet S1,j) \oplus (3 \bullet S2,j) \oplus S3,j$$
$$S'2,j = S0,j \oplus S1,j \oplus (2 \bullet S2,j) \oplus (3 \bullet S3,j)$$
$$S'3,j = (3 \bullet S0,j) \oplus S1,j \oplus S2,j \oplus (2 \bullet S3,j)$$

Where • denotes multiplication over the finite field GF(2^8).

Figure 2 : Matrix Multiplication on state

iv) AddRoundKey

It is designed to provide Key Dependency and Asymmetry. It operates independently on each byte of the State matrix by adding it with the corresponding byte of the Subkey using bitwise XOR. For each round, a Subkey is derived from the main key using the keyexpansion function. Each subkey has the same size as the state matrix .

The final round includes all the transformations except MixColumn. After completing all the ten rounds the output is 128 bits in encrypted format called cipher text.

Key Expansion

Key expansion is an important for both encryption and decryption. The AES key expansion algorithm takes as input a 4-word (16 bytes) key and produces a linear array of 44 words (176 bytes).This is sufficient to provide a 4-word round key for the initial Add Round Key stage and each of the 10 rounds of the cipher. The following figure shows pseudo code for generating the expanded key from the actual key.

```
KeyExpansion (byte key [16], word w [44])
{
Word temp
for(i=0;i<4;i++)
w[i]=(key[4*i], key[4*i+1], key[4*i+2], key[4*i+3]);
for(i=4;i<44;i++)
{
temp=w[i-1];
if(i mod 4=0)
temp=SubWord(RotWord(temp)) ⊕ Rcon[i/4];
w[i]=w[i-4] ⊕ temp;
}
}
```

The key is copied into the first four words of the expanded key. The remainder of the expanded key is filled in four words at a time. Each added
word w[i] depends on the immediately preceding word, w[i-1], and the word four positions back w[i-4].

Blowfish

[NC-Rase 18]
DOI: 10.5281/zenodo.1483995
ISSN 2348 – 8034
Impact Factor- 5.070

Pseudo Code of Blowfish Algorithm

begin itemize
Blowfish has 16 rounds.
The input is a 64-bit data element, x.
Divide x into two 32-bit halves: xL, xR.
Then, for i = 1 to 16:
xL = xL XOR Pi
xR = F(xL) XOR xR
Swap xL and xR

After the sixteenth round, swap xL and xR again to undo the last swap.
Then, xR = xR XOR P17 and xL = xL XOR P18.
Finally, recombine xL and xR to get the ciphertext. Decryption is exactly the same as encryption, except that P1, P2,..., P18 are used in the reverse order. Implementations of Blowfish that require the fastest speeds should unroll the loop and ensure that all subkeys are stored in cache [9, 10].

Key generation

• Large number of sub keys is used in blowfish.
• The p-array consists of 18, 32-bit sub keys.
P1,P2,…….,P18
• S-Boxes consist of 256 entries each,
S1, 0, S1,1,……. S1, 255
S2, 0, S2,1,…….. S2, 255
S3, 0, S3,1,…….. S3, 255
S4, 0, S4,1,……... S4, 255

Steps to Generate Sub Keys
1) Initialize first the P-array and then the four S-boxes.
2) The first 32 bits of the key is with XOR P1, the second 32-bits of the key is with XOR P2.
3) Encrypt the all-zero string with the Blowfish algorithm, using the sub keys described in steps (1) and (2).
4) This new output is now P1 and P2.
5) Encrypt the output of step (3) using the Blowfish algorithm with the modified sub keys.
6) This new output is now P3 and P4.
7) Repeat 521 times in order to calculate the new sub keys for the P- array and Four S- boxes.

III. IMPLEMENTATION

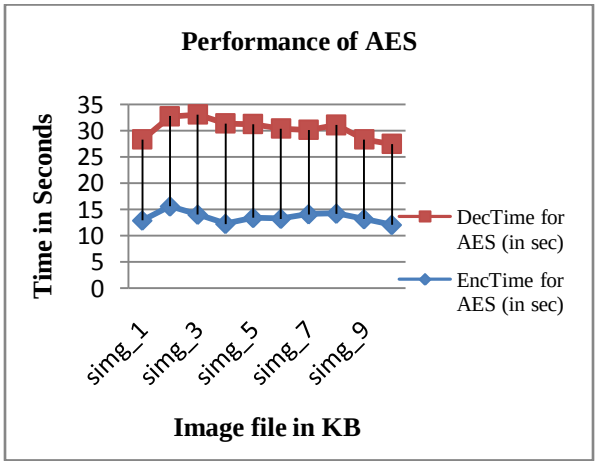
Both the algorithms AES and Blowfish are implemented in MATLAB 2011B on Street View Text (SVT) dataset. The following table shows the results for AES algorithm.

Table 1:Encryption and Decryption time using AES algorithm

Image File	EncTime for AES (in sec)	DecTime for AES (in sec)
simg_1	12.8963	15.4335
simg_2	15.5877	17.1697
simg_3	14.0168	19.049
simg_4	12.2666	19.1165
simg_5	13.3995	17.8594
simg_6	13.2348	17.1037
simg_7	14.0978	16.0071

ISSN 2348 – 8034
Impact Factor- 5.070

simg_8	14.2068	16.851
simg_9	13.1588	15.0954
simg_10	12.0649	15.393

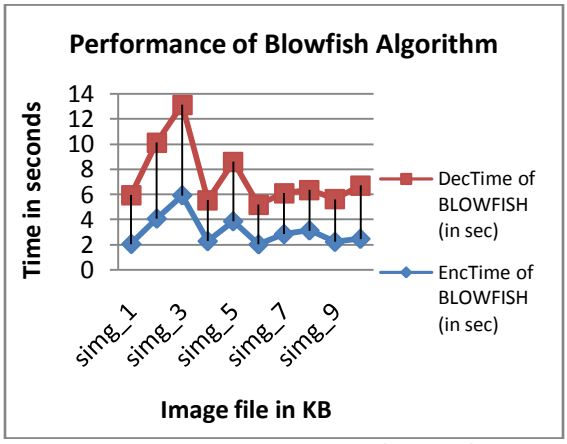


Graph 1: Encryption and Decryption time of AES algorithm

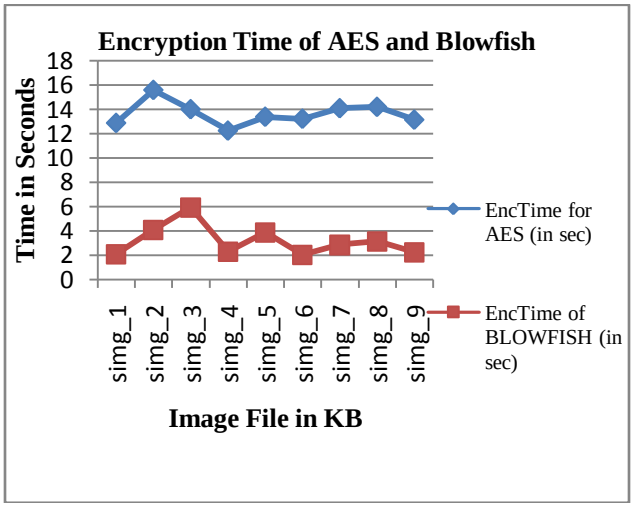
Table 2:Encryption and Decryption time using BLOWFISH algorithm

Image File	EncTime of BLOWFISH (in sec)	DecTime of BLOWFISH (in sec)
simg_1	2.0741	3.8855
simg_2	4.0829	6.0387
simg_3	5.9221	7.2156
simg_4	2.3013	3.2498
simg_5	3.8681	4.7308
simg_6	2.0435	3.1608
simg_7	2.8564	3.2649
simg_8	3.1485	3.2146
simg_9	2.2426	3.4008
simg_10	2.4819	4.2392

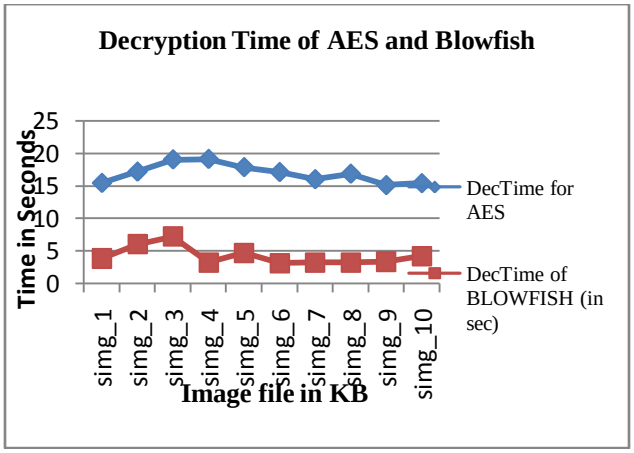
ISSN 2348 – 8034
Impact Factor- 5.070



Graph 2:Encryption and Decryption time for Blowfish algorithm



Graph 3: Encryption Time required for AES and BLOWFISH



Graph 4: Decryption Time required for AES and BLOWFISH

[NC-Rase 18]
DOI: 10.5281/zenodo.1483995
ISSN 2348 – 8034
Impact Factor- 5.070

IV. CONCLUSION

Cryptography deals with the actual securing of digital data. It refers to the design of mechanisms based on mathematical algorithms that provide fundamental information security services. It is an establishment of a large toolkit containing different techniques in security applications.

The encryption and decryption time of symmetric key algorithms AES and Blowfish is compared and analyzed. The evaluation result shows that, the execution time required for Blowfish algorithm is less than AES algorithm therefore the performance of Blowfish algorithm is found to be better than AES.

REFERENCES

1. Atul Kahte, "Cryptography and Network Security", Tata Mcgraw Hill, 2007.
2. NIST - FIPS Standard, n.197, 2001, November 26, Announcing the ADVANCED ENCRYPTION STANDARD (AES), in Federal Information Processing Standards Publication.
3. A. A. Zaidan, "An overview: Theoretical and Mathematical Perspectives for Advance Enrcyption Standard/Rijndael Journal of Applied Sciences 10 (18): 2161-2167, 2010, ISSN 1812-5654.
4. Manisha S. Mahindrakar, "Evaluation of Blowfish Algorithm based on Avalanche Effect" International Journal of Innovations in Engineering and Technology (IJIET) Vol. 4 Issue 1 June 2014 ISSN: 2319 – 1058.
5. PratapChnadraMandal, "Superiority of Blowfish Algorithm" International Journal of Advanced Research in Computer Science and Software Engineering Volume 2, Issue 9, September 2012 ISSN: 2277 128X
6. Anjaneyulu GSGN, Pawan Kumar Kurmi, Rahul Jain" Image Encryption And Decryption Using Blowfish Algorithm With Randomnumber Generator" International Journal Of Pharmacy &Technology(IJPT) Jan-2015 Vol. 6 Issue No.3 pp 7164-7170 ISSN: 0975-766X.
7. T. Saravanan, V. Srinivasan, R, Udayakumar "MATLAB-Simulink Implementation of AES Algorithm for Image Transfer" Middle-East Journal of Scientific Research 18(12) 1709-1712, 2013.
8. Hamdan.O.Alanazi, B. B. Zaidan, A. A. Zaidan, Hamid A. Jalab, M.Shabbir and Y. Al-Nabhani "New Comparative Study Between DES, 3DES and AES within Nine Factors" VOLUME 2, ISSUE 3, MARCH 2010.
9. Tanjyot Aurora, ParulArora, "Blowfish Algorithm" International Journal of Computer Science and Communication Engineering IJCSCE Special issue on "Recent Advances in Engineering & Technology" NCRAET-2013 ISSN 2319-7080.
10. Chaitali Haldankar, Sonia Kuwelkar, "IMPLEMENTATION OF AES AND BLOWFISH ALGORITHM" IJRET: International Journal of Research in Engineering and Technology Volume: 03 Special Issue: 03 May-2014 eISSN: 2319-1163 pISSN: 2321-7308..