International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869 (O) 2454-4698 (P), Volume-5, Issue-2, June 2016

Halftone

Ranimol K K, Mrs. Divya Parameswaran

***Abstract*— Secure data transmissions prevent contact lists and personal e-mail from being read by someone other than the intended recipient, keep firmware upgrades out of devices they don't belong in, and verify that the sender of a piece of information is who he says he is. The sensibility of data security is even mandated by law in certain applications: in the U.S. electronic devices cannot exchange personal medical data without encrypting it first, and electronic engine controllers must not permit tampering with the data tables used to control engine emissions and performance. Due to growth of multimedia application, security becomes an important issue of communication and storage of images. This paper is about encryption and decryption of images using a secret-key block cipher called 64-bits Blowfish designed to increase security and to improve performance. This algorithm will be used as a variable key size up to 448 bits. It employs Feistel network which iterates simple function 16 times. Encryption algorithms can also provide authentication, the assurance that a message came from whom it says it came from. Halftone is the reprographic technique that simulates continuous tone imagery through the use of dots, varying either in size or in spacing, thus generating a gradient like effect.**

***Index Terms*—Reprographic technique, Blowfish Algorithm, Halftone, Secret-key block cipher, Cryptography, Image processing.**

I. INTRODUCTION

Halftone can also be used to refer specifically to the image that is produced by this process. Where continuous tone imagery contains an infinite range of colors or grays, the halftone process reduces visual reproductions to an image that is printed with only one color of ink, in dots of differing size (amplitude modulation) or spacing (frequency modulation). In this paper secret image and the visible image are first converted to halftone image using Halftone error diffusion method. Recovered image from the shares generated by is found to be of better quality.

Blowfish is a symmetric block cipher that can be used as a drop-in replacement for DES or IDEA. It will applied two shares then will get encrypted or decrypted forms. It takes a variable-length key, from 32 bits to 448 bits, making it ideal for both domestic and exportable use. Blowfish was designed in 1993 by Bruce Schneier as a fast, free alternative to existing encryption algorithms. Since then it has been analyzed considerably, and it is slowly gaining acceptance as a strong encryption algorithm. Blowfish is unpatented and license-free, and is available free for all uses.

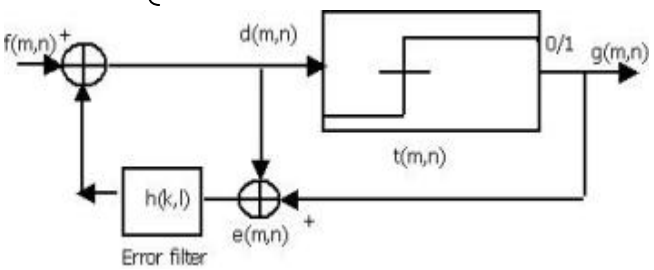
Many cryptographers have examined Blowfish, although there are few published results. Serge Vaudenay examined weak keys in Blowfish; there is a class of keys that can be detected--although not broken--in Blowfish variants of 14 rounds or less. Vincent Rijmen's Ph.D. thesis includes a second-order differential attack on 4-round Blowfish that cannot be extended to more rounds.

Image security is of almost concern as web attacks have become more and more serious. Image encryption decryption has applications in internet communication, multimedia systems, medical imaging, telemedicine, military communication, etc. To make the data secure from various attacks and for the integrity of data we must encrypt the data before it is transmitted or stored. Government, military, financial institution, hospitals and private business deals with confidential images about their patient (in Hospitals), geographical areas (in research), enemy positions (in defense), product, financial status. Most of this information is now collected and stored on electronic computers and transmitted across network to other computer. If these confidential images about enemy positions, patient and geographical areas fall into the wrong hands, than such a breach of security could lead to declination of war, wrong treatment etc. Protecting confidential images is an ethical and legal requirement.

1.1Half toning Process

Error Diffusion method is easy and simpler to other half toning technique . This is efficient algorithm for half-toning gray scale image. The quantization error is filtered at each pixel and fed back to set of future input samples. Figure represents a binary error diffusion where f(m, n) represents the (m, n)the pixel of the input grayscale image, d(m, n) is the sum of the input pixel value and the diffused past errors, and is the output quantized pixel value. The pixel f(m, n) passes through a quantizer to get the corresponding pixel of the halftone image g(m, n). The difference between these two pixels is diffused to the neighboring pixels by means of the error filter h(k, l). Error diffusion consists of two main components. The first component is the thresholding block where the output g(m, n) is given by

$$g(m,n) = \begin{cases} 1, \text{ if } d(m,n) <= t(m,n) \\ 0, \text{ Otherwise.} \end{cases}$$



The t(m, n) is the threshold position-dependent value. The second parameter is error filter h (k, l) whose input e(m, n) is the difference between d(m, n) and g(m, n).

Ranimol K K, PG Student of MCA, KVM College of Engineering and Information Technology, Cherthala, Kerala

Mrs.Divya Parameswaran, Assistant Professor, KVM College of Engineering and Information Technology Cherthala, Kerala

Finally we calculate d(m, n) as

$$d(m,n)= f(m,n)-\sum h(k,l)e(m-k,n-l)$$

$1/16 *$

	•	7
3	5	1

Floyd Steinberg error filter[2].

Represent the current pixel. The weights are given by h(0, 1)=7/16, h(1, -1)=3/16, h(1, 0)=5/16, and h(1, 1)=1/16.

Constructing initial share

Using halftone image of both input secret image and visual image, we construct the meaningful shares. We have halftone visual image of 384*384 and halftone input secret image of 128*128 because each pixel of input secret image is represented by 3*3 of visual image.

1.2 Constructing Final Share

1. For each pixel of input halftone image we have block of 3*3 halftones visual
Image and block of 3*3 complementary halftone visual image.
2. Then if input halftone image pixel value is == 0
3. Then take block of 3*3 from halftone visual image and
4. Also take block of 3*3 from complementary halftone visual image.
5. Select four pixels of 3*3 block of both halftone visual image and complementary halftone visual image.
Block (2, 2)
Block (2, 3)
Block (3, 2)
Block (3, 3)

r/c	1	2	3
1			
2		*	*
3		*	*

Assign the value on place of * in 3*3 block

$$M1 = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \quad or \quad \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$$

$$and\ M0 = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} \quad or \quad \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$$

6. if input halftone image pixel value is == 1
Assign the value on place of * in 3*3 block

$$M1= \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \quad or \quad \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$$

$$and\ M0 = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \quad or \quad \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$$

7. After this process we will get two shares, one from the halftone visual image and other from the complementary halftone visual image.
8. These shares named as Share1 and Share2.
9. After applying Blowfish algorithm in these two shares we will get Recovered Secret Image.

1.3 Blowfish Encryption Algorithm

Blowfish was designed in 1993 by Bruce Scheier as a fast, alternative to existing encryption algorithms such AES, DES and 3 DES etc.

Blowfish is a symmetric block encryption algorithm designed in consideration with,

- **Fast :** It encrypts data on large 32-bit microprocessors at a rate of 26 clock cycles per byte.
- **Compact:** It can run in less than 5K of memory.
- **Simple:** It uses addition, XOR, lookup table with 32-bit operands.
- **Secure:** The key length is variable ,it can be in the range of 32~448 bits: default 128 bits key length.
- It is suitable for applications where the key does not change often, like communication link or an automatic file encryptor.
- Unpatentedandroyalty-free.

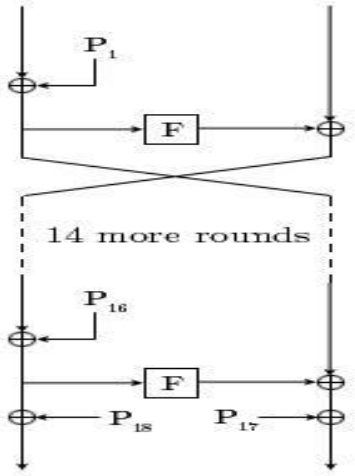


Fig 1: The Feistel structure of Blowfish

1.4 Blowfish Description of Algorithm

Blowfish symmetric block cipher algorithm encrypts block data of 64-bits at a time. It will follows the feistel network and this algorithm is divided into two parts.

1.4.1. Key-expansion

1.4.2 Data Encryption

1.4.1 Key-expansion:

It will converts a key of at most 448 bits into several sub key arrays totaling 4168 bytes. Blowfish uses large number of sub keys.

These keys are generate earlier to any data encryption or decryption.

The p-array consists of 18, 32-bit sub keys:

P1,P2,……,P18

Four 32-bit S-Boxes consists 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

Generating the Subkeys :

The subkeys are calculated using the Blowfish algorithm:

1. Initialize first the P-array and then the four S-boxes, in order, with a fixed string. This string consists of the hexadecimal digits of pi (less the initial 3): P1 = 0x243f6a88, P2 = 0x85a308d3, P3 = 0x13198a2e, P4 = 0x03707344, etc.
2. XOR P1 with the first 32 bits of the key, XOR P2 with the second 32-bits of the key, and so on for all bits of the key (possibly up to P14). Repeatedly cycle through the key bits until the entire P-array has been XORed with key bits. (For every short key, there is at least one equivalent longer key; for example, if A is a 64-bit key, then AA, AAA, etc., are equivalent keys.)
3. Encrypt the all-zero string with the Blowfish algorithm, using the subkeys described in steps (1) and (2).
4. Replace P1 and P2 with the output of step (3).
5. Encrypt the output of step (3) using the Blowfish algorithm with the modified subkeys.
6. Replace P3 and P4 with the output of step (5).
7. Continue the process, replacing all entries of the P array, and then all four S-boxes in order, with the output of the continuously changing Blowfish algorithm.

In total, 521 iterations are required to generate all required subkeys. Applications can store the subkeys rather than execute this derivation process multiple times.

1.4.2 Data Encryption

It is having a function to iterate 16 times of network. Each round consists of key-dependent permutation and a key and data-dependent substitution. All operations are XORs and additions on 32-bit words. The only additional operations are four indexed array data lookup tables for each round.

Algorithm:Blowfish Encryption

Divide x into two 32-bit halves: xL, xR

For i = 1to 16:

xL = XL XOR Pi

xR = F(XL) XOR xR

Swap XL and xR

Swap XL and xR (Undo the last swap.)

xR = xR XOR P17

xL = xL XOR P18

Recombine xL and xR

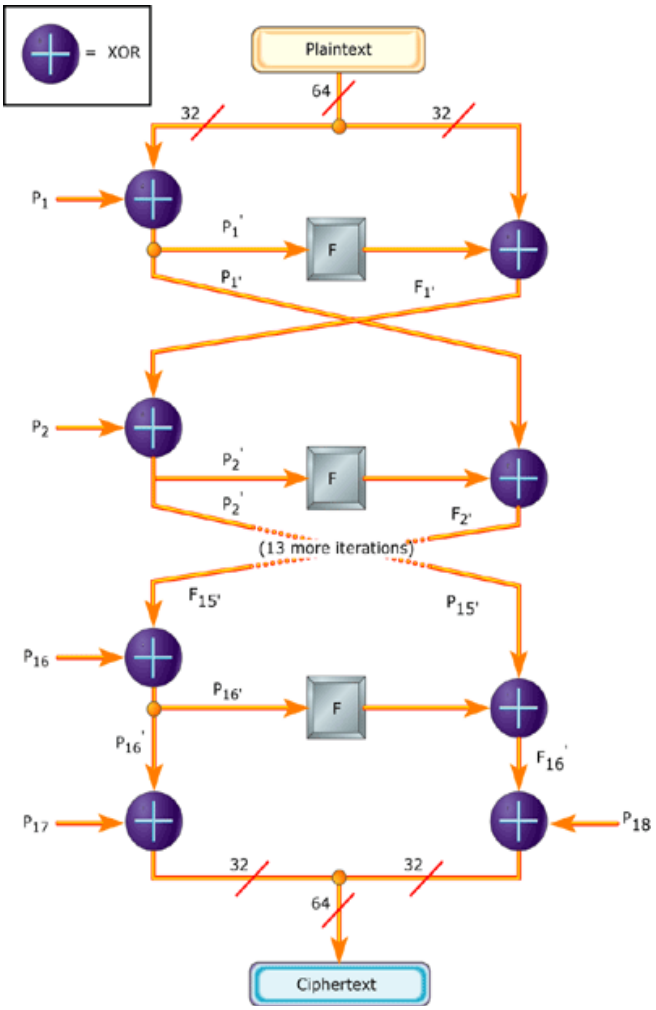


Fig 2: Blowfish Encryption

1.5 Blowfish in pseudocode

```
uint32_t P[18];
uint32_t S[4][256];

uint32_t f (uint32_t x) {
  uint32_t h = S[0][x >> 24] + S[1][x >> 16 & 0xff];
  return ( h ^ S[2][x >> 8 & 0xff] ) + S[3][x & 0xff];
}

void encrypt (uint32_t & L, uint32_t & R) {
  for (int i=0 ; i<16 ; i += 2) {
    L ^= P[i];
    R ^= f(L);
    R ^= P[i+1];
    L ^= f(R);
  }
  L ^= P[16];
  R ^= P[17];
  swap (L, R);
}

void decrypt (uint32_t & L, uint32_t & R) {
  for (int i=16 ; i > 0 ; i -= 2) {
    L ^= P[i+1];
    R ^= f(L);
    R ^= P[i];
    L ^= f(R);
  }
  L ^= P[1];
  R ^= P[0];
  swap (L, R);
}

{
  // ...
  // initializing the P-array and S-boxes with values derived
from pi; omitted in the example
  // ...
  for (int i=0 ; i<18 ; ++i)
    P[i] ^= key[i % keylen];
  uint32_t L = 0, R = 0;
  for (int i=0 ; i<18 ; i+=2) {
    encrypt (L, R);
    P[i] = L; P[i+1] = R;
  }
  for (int i=0 ; i<4 ; ++i)
    for (int j=0 ; j<256; j+=2) {
      encrypt (L, R);
      S[i][j] = L; S[i][j+1] = R;
    }
}
```

II. RELATED WORKS

In this section we are going to analyze different algorithms for Encryption and Decryption.

A) DES algorithm using Transportation Cryptography Techniques:-Data encryption standard (DES) is a private key cryptography system that provides the security in communication system but now a days the advancement in the computational power the DES seems to be weak against the brute force attacks. To improve the security of DES algorithm the transposition technique is added before the DES algorithm to perform its process. If the transposition technique is used before the original DES algorithm then the intruder required first to break the original DES algorithm and then transposition technique. So the security is approximately double as compared to a simple DES algorithm.

B)Image Encryption Using Block-Based Transformation Algorithm:-Here a block-based transformation algorithm based on the combination of image transformation and a well known encryption and decryption algorithm called Blowfish is used. The original image was divided into blocks, which were rearranged into a transformed image using a transformation algorithm presented here, and then the transformed image was encrypted using the Blowfish algorithm. The results showed that the correlation between image elements was significantly decreased by using the proposed technique.

BLOCK DIAGRAM

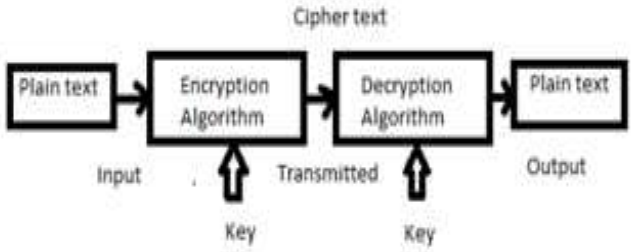


Fig: Encryption / Decryption Process

III. SIMULATION RESULT

In this paper we have simulated the image processing part of Encryption and decryption in Visual Basic software. Here we would be taking an image & obtaining the matrix and pixels of the chosen image & then we would be encrypting the image matrix using blowfish algorithm. The result shows the original image, encrypted image and the decrypted image. The text in the image will be hidden using a specific key and image hidden with a data is encrypted and decrypted by a 32 bit iteration loop.

ALOGRITHM	CREATED BY	KEY SIZE(BITS)	BLOCK SIZE(BITS)
DES	IBM IN 1975	56	64
3DES	IBM IN 1978	112 OR 168	64
RIJNDAEL	JOAN DAEMEN & VINCENT RIJMEN IN 1998	256	128
BLOWFISH	BRUCE SCHNEIER IN 1993	32-448	64

Fig. Comparison of Algorithms on the Basis of Block Size

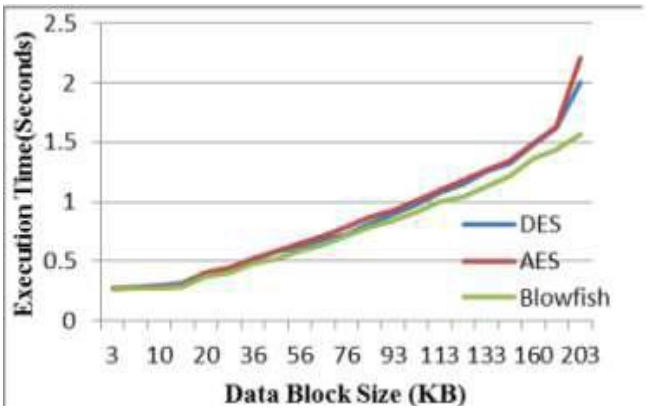


Fig. Comparison of Execution Time

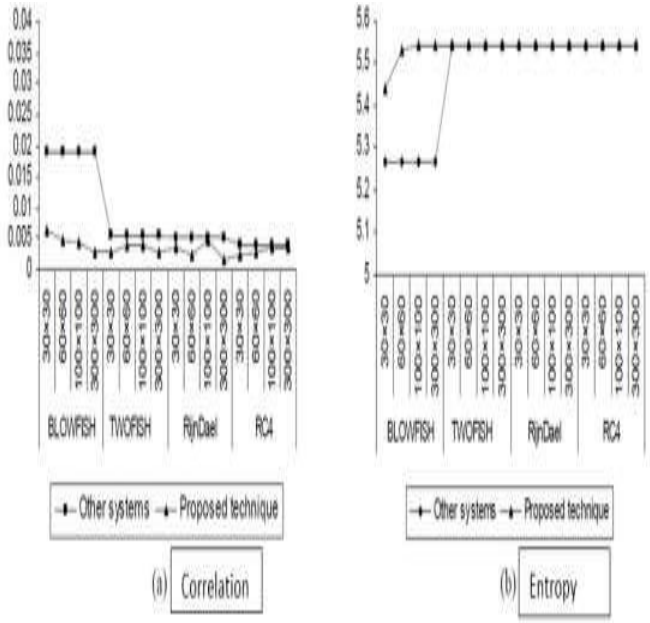


Fig. Correlation & Entropy

In this paper we have simulated the image processing part of Encryption and decryption in JAVA software. Here we would be taking an image. Firstly we would be obtaining the matrix and pixels of the chosen image & then we would be encrypting the image matrix using blowfish algorithm. The result shows the original image, encrypted image and the decrypted image. The text in the image will be hidden using a specific key and image hidden with a data is encrypted and decrypted by a 32 bit iteration loop and display in JAVA.

IV. CONCLUSION

In visual secret sharing schemes, a secret image can be encoded into halftone shares taking meaningful visual information. Then applied the blowfish algorithm into these halftone images and construct the share image, in reverse order will get the recovered image. Histogram of encrypted image is less dynamic and significantly different from the respective histograms of the original image. Blowfish cannot be broken until an attacker tries 28r+1 combinations where r is the number of rounds. Hence if the number of rounds are been increased then the blowfish algorithm becomes stronger. Since Blowfish has not any known security weak points so far it can be considered as an excellent standard encryption algorithm.

REFERENCES

[1]Zhongmin Wang, Student Member, IEEE, Gonzalo R. Arce, Fellow, IEEE, and Giovanni Di Crescenzo. *Halftone Visual Cryptography Via Error Diffusion*. IEEE Transactions on Information Forensics and Security; VOL. 4,NO. 3, SEPTEMBER 2009.

[2]N. Askari, H. M. Heys, and C. R. Moloney. *An Extended Visual Cryptography Scheme Without Pixel Expansion For Halftone Images.* 26th IEEE Canadian Conference Of Electrical And Computer Engineering; (CCECE)-2013.

[3] Zhongmin Wang, Student Member, IEEE, Gonzalo R. Arce, Fellow, IEEE, and Giovanni Di Crescenzo. *Halftone Visual Cryptography*. IEEE Transactions on Image Processing; VOL. 15, NO. 8. AUGUST 2006.

[4] Aloha Sinha, Kehar Singh, "A technique for image encryption using digital signature", Optics Communications, Vol-2 I 8 (2203), 229-234.

[5]DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis, International Journal of Emerging Technology and Advanced Engineering.

[6] Mohammad Ali Bani Younes and Aman Jantan ,Image Encryption Using Block-Based Transformation Algorithm, IAENG International Journal of Computer Science, 35:1, IJCS_35_1_03.

[7] Nadeem, Aamer; "A Performance Comparison of Data Encryption Algorithms", IEEE 2005.

[8] C. C. Lin and W. H. Tsai. *Visual cryptography for gray-level images by dithering techniques*. Pattern Recognit. Lett. vol. 24; pp. 349358, Jan. 2003.

[9] N. D. Venkata and B. L. Evansi. *Adaptive threshold modulation for error diffusion halftoning*. IEEE Trans. Image Process; vol. 10, no. 1, pp. 104116,Jan. 2001.

[10] P. Li and J. P. Allebach. *Tone-dependent error diffusion"*, ", IEEE Trans. Image Process; vol. 13, no. 2, pp. 2012 , Feb. 2004.