International Journal of Engineering Research and General Science Volume 3, Issue 1, January-February, 2015
ISSN 2091-2730

Analysis of modified Blowfish Algorithm in different cases with various parameters

Vaibhav Poonia, Dr. Narendra Singh Yadav

Computer Science

Sri Balaji College Of Engineering & Technology

Jaipur, India

vaibhavpoonia007@gmail.com

narensinghyadav@yahoo.com

Contact No.: +91-9549327490

Abstract— Security has always been a great concern whenever there is communication between sender and receiver. To overcome the issues of security breaches many cryptographic algorithms are used like: AES, DES, Triple DES, Blowfish, etc. The objective of this paper is to enhance and evaluate the Blowfish algorithm on the basis of different parameters like Encryption Quality, Correlation Coefficients, Key Sensitivity Test and Size of Output File. The 'f' function is modified by mixing the XOR and addition used in the original algorithm. Four cases are created and analyzed. The results of all the tests conducted on these cases lead to a common conclusion that the security of the modified algorithm with different cases makes the original Blowfish algorithm more compact and more secure than the earlier.

Keywords— Blowfish algorithm; Encryption Quality; Correlation coefficient; Key Sensitivity; Size of output file; 'f' function; XOR

INTRODUCTION

Due to the swift increase in the digital communication and exchange of electronic data, the security of information has become an important issue in business, industry, and administration. In modern era security is the major issue for every communication between sender and receiver. If there are any security breaches in between communication then there will be major loss to both of them, sender and receiver. The cryptography used today gives many essential techniques for protecting data and securing information.

Cryptography

Cryptography is an essential part for the Information Security System (ISS). It plays an important role in the security of data between sender and receiver. Cryptography provides us confidentiality, accuracy, fairness, along with data integrity. Now the cryptography is used routinely to secure data, which must be communicated and/or saved over long periods, to protect electronic fund transfers and classified communications.

Modern cryptographic techniques are based on number theoretical or algebraic concepts. Before going on our main topic we need to know at least brief information about security trends in cryptography, what are the various security attacks could be possible, what are the various security services and what are security mechanisms should be applied to achieve those services.

Types of Cryptography

Mainly two types of cryptography are known: Asymmetric key cryptography and Symmetric key cryptography.

- Asymmetric Key Cryptography

In this type of cryptography, there are two keys used: public key and private key, one for encryption and one for decryption purpose. Popular examples of asymmetric key cryptography are: RSA, ElGamal, Merkle's Puzzles, Elliptic Curve Cryptography (ECC) [2]. An Asymmetric key cryptography is also known as public key cryptography. This algorithm don't need a secured beginning exchange of

one or more keys between the sender and receiver. The algorithm used for encryption and decryption was designed in such a way that, it makes easy for the receiver to produce the public and private keys and to decrypt the message by private key. It is also easy for the sender to encrypt the message by utilizing public key, and it is very difficult for anyone to find out the private key based on the knowledge of the public key.

- Symmetric Key Cryptography

In this type of cryptography, same key is used for both encryption and decryption purpose. Symmetric algorithms can be divided into two type-stream cipher and block cipher. Stream cipher encrypt one bit of plaintext at a time as compared to block cipher which takes a number of bits (typically 64 bits), and encrypt them as one unit in whole. Symmetric ciphers are likely to be harmed by the known plaintext and chosen text attacks, as well as differential and linear. Some examples of popular symmetric algorithms are: Serpent, Twofish, AES (Rijndael), Blowfish, CAST5, RC4, RC6, DES, 3DES, and IDEA. Symmetric key algorithms are less computationally intensive as compared to asymmetric key algorithms. But in practice, asymmetric key algorithms are much slower as compared to the symmetric key algorithms. Asymmetric algorithms(also known as public-key algorithms) requires at least a 3,000-bit key to reach at the same level of security as that of a 128-bit symmetric algorithm.

Blowfish

Blowfish algorithm is a symmetric block cipher which can be used as a drop-in replacement for IDEA or DES. It takes a changeable-length key, from 32 bits to 448 bits, which makes it perfect for both exportable and domestic use. Blowfish was designed by Bruce Schneier in 1993 as a free alternative to the present encryption algorithms.

Blowfish is a 16 rounds Feistal Structure as shown in fig 1.2 and fig 1.3. Every round is made up of a key- and data-dependent substitution and a key-dependent permutation. All operations are additions on 32-bit words and XOR. The only additional operations, for every round are performed in the following way:

1. Split each block into halves
2. Right half becomes new left half
3. The right half is made when XOR is done on the left half and the result we get after applying 'f' to the right half and the key.
4. The rounds which are prior can be obtained even if the function 'f' is not turned upside down.

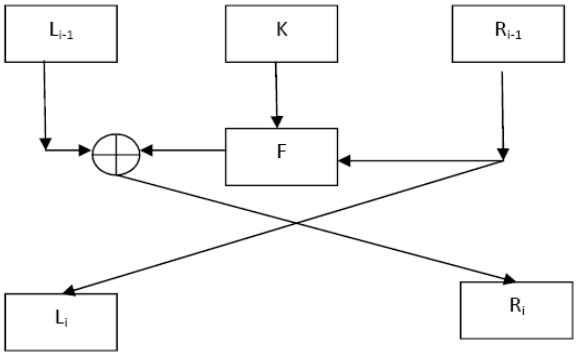

Fig 1.1: Feistal Network

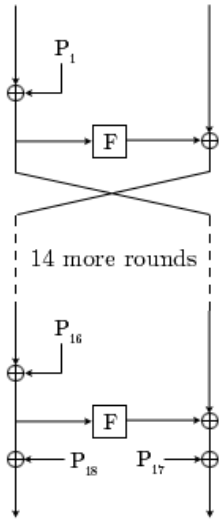

Fig 1.2: Blowfish Feistal Structure of 16 rounds.

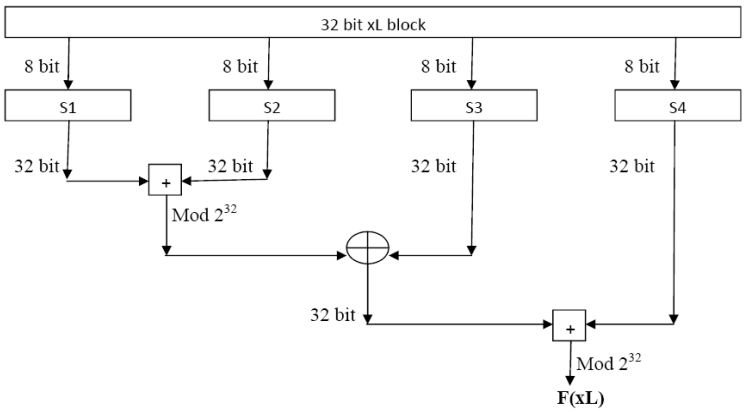

Fig 1.3: S-Box operation (F function) of Blowfish algorithm

REVIEW OF RELATED WORK

Blowfish is one of the fastest block ciphers used by many people, except when changing keys [3].Many researchers have tried to test the security provided by Blowfish algorithm and they have concluded that it is a secure algorithm to use[1][5][6][4]. In this paper we have improved the original Blowfish Algorithm by changing its F function to different cases. After analyzing those changes to F function of Blowfish algorithm, we will see that most of the changes makes original Blowfish Algorithm most secure. One more thing to add here is the capability of compression and decompression to the encrypted files. This additional feature makes it more compact than the earlier. The comparison is based on the basis of Encryption quality, Correlation analysis, Key Sensitivity test and size of the encrypted file.

METHODOLOGY

I have made the four cases of F function with two ADD and one XOR or with two XOR and one ADD operation. The followings are the four cases. These are shown in fig 3.1, fig 3.2, fig 3.3 and fig 3.4 respectively for Case 1, Case 2, case 3 and Case 4.

Case 1:

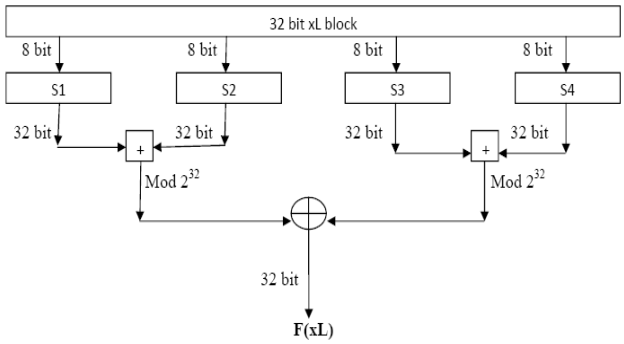

Fig 3.1: Modified Blowfish with case 1.

In this case F(xL) can be calculated as:

$F(xL)=(((S1 + S2) \bmod 2^{32}))XOR((S3+S4) \bmod 2^{32}))$.

Case 2:

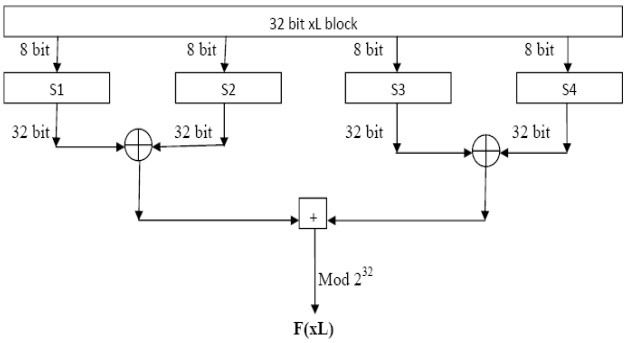

Fig 3.2: Modified Blowfish with case 2.

In this case F(xL) can be calculated as:

$F(xL)= ((S1 \text{ XOR } S2)+(S3 \text{ XOR } S4) \bmod 2^{32})$.

Case 3:

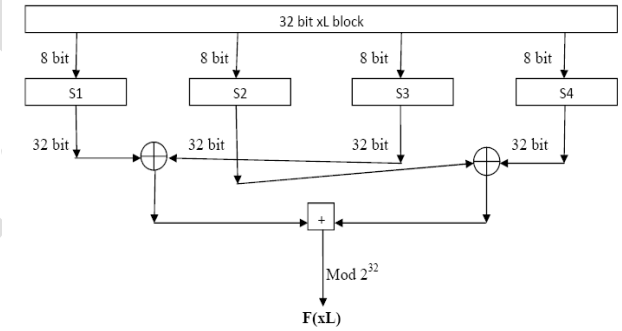

Fig 3.3: Modified Blowfish with case 3.

In this case F(xL) can be calculated as:

$F(xL)= ((S1 \text{ XOR } S3)+(S2 \text{ XOR } S4) \bmod 2^{32})$.

Case 4:

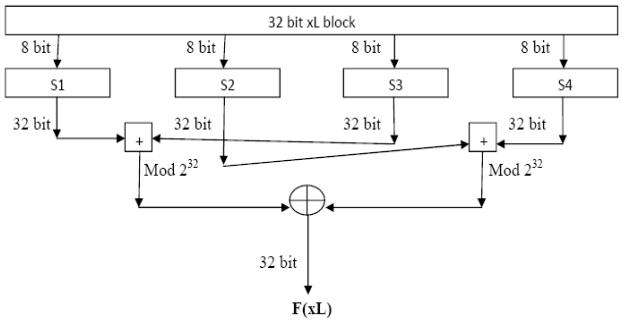

Fig 3.4: Modified Blowfish with case 4.

In this case F(xL) can be calculated as:

$F(xL)=(((S1 + S3) \bmod 2^{32}))XOR((S2+S4) \bmod 2^{32}))$.

EXPERIMENTAL RESULTS AND PERFORMANCE ANALYSIS

The Modified Blowfish Algorithm was successfully implemented in Java. In all experiments a grey scale (0-255) bitmap image is used as the original image (plain image) of size 512x512. The above four cases are analyzed on the basis of following parameters:

Encryption quality, Correlation coefficient analysis, Key sensitivity test and Size of data file after encryption.

i. Encryption Quality:

To evaluate the quality of encryption [7], [8],[11],[12] of modified Blowfish cipher with that of original Blowfish [2], [3], [9], [10] the ciphers are applied to several digital images. Before encryption/decryption, we must first extract the image header for the image to be encrypted/ decrypted. So, we must study the file format for image to determine all parts of the file header and to determine the beginning of the data stream to be encrypted. Then, the ciphers are applied to the image.

How the total number of round (r) affects the encryption quality for blowfish and modified blowfish algorithm is investigated. Both the block size and key length are kept persistent. The encryption quality (EQ) is calculated by using the number of rounds and the result is obtained for the image which is mentioned above in *table 4.1*. These results are also in column-chart as shown in *fig 4.1*.

Table 4.1 : *Encryption quality for image.*

No. Of rounds	Original Blowfish	Mod. Blowfish Case 1	Mod. Blowfish Case 2	Mod. Blowfish Case 3	Mod. Blowfish Case 4
2	809.123	820.279	827.176	839.813	842.105
4	815.236	821.923	830.275	839.998	847.341
6	819.991	828.227	833.702	838.513	848.385
8	821.276	829.769	832.993	839.095	849.113
10	829.458	831.093	835.621	841.387	850.111
12	832.734	836.776	840.789	845.219	853.876
14	839.986	839.997	845.453	849.001	859.176
16	842.669	847.886	849.361	854.639	865.659

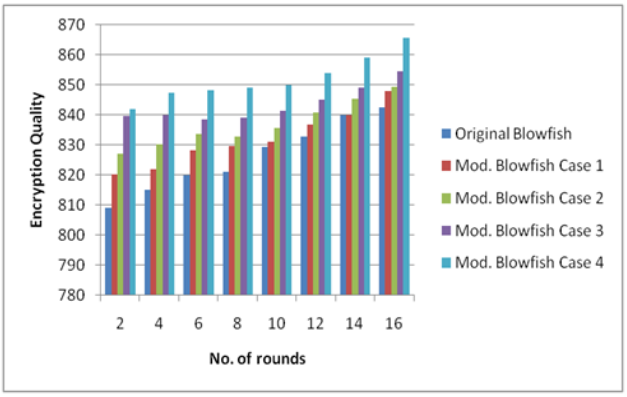

Fig 4.1: *Column chart for comparison of encryption quality.*

ii. Correlation analysis:

To find the correlation between two adjoining pixels [7], [10] of an image, the following steps are taken: First, select 'n' pairs of adjoining pixels from an image randomly. Now calculate the correlation coefficient using the following formula:

r = cov(x,y)/D(x)^0.5*D(y)^0.5

where x and y shows the grey-scale value of adjoining pixels of the image. D(x) and D(y) shows the difference of x and y values, cov(x,y) shows the covariance of x and y; and r shows the correlation coefficient. To identify the correlation between two adjoining pixels we have selected 1000 pixels randomly and pixels adjoining to them from the original picture and the encrypted images. After that we have computed the correlation coefficient using the above equations. *Table 4.2* and *fig 4.2* shows the correlation coefficients in original image and encrypted images encrypted by various Blowfish algorithms.

Table 4.2: *Comparison of correlation coefficients between pixels using different form of blowfish algorithm for the image.*

Algorithm used	Correlation coefficient
Original image	0.9984
Original Blowfish	0.0414
Modified Blowfish with case 1	0.0186
Modified Blowfish with case 2	0.0123
Modified Blowfish with case 3	0.0101
Modified Blowfish with case 4	0.0099

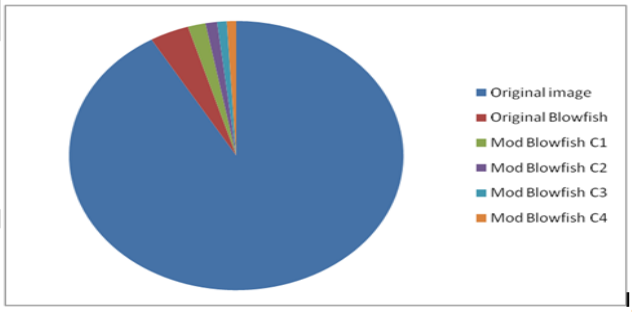

Fig 4.2: *Pie chart for comparison of correlation coefficient for image.*

iii. Key Sensitivity Test:

Assume that a 16-character ciphering key is used. This means that the key has 128 bits. To test the key sensitivity [7], [3], [8] following steps are taken:

Image is encrypted by using the test key 1234567890098765432112345678900**9**(Hex). After that one bit from key is randomly selected and it is changed. Here we have modified one bit of the key, that is; 1234567890098765432112345678900**1**.

The same image is then encrypted by using this modified key. The bit changed is shown in bold in test and the modified key. Ultimately, we have compared the images which are encrypted by the two slightly different keys in Table 4.3.

Table 4.3: *Comparison of pixel difference.*

Original Blowfish	99.565292%
Modified Blowfish Case 1	99.623383%
Modified Blowfish Case 2	99.651372%
Modified Blowfish Case 3	99.671372%
Modified Blowfish Case 4	99.700732%

iv. Size of Data File:

The Entered image file was of size 800.0 KB, when we encrypt it by original Blowfish and by modified Blowfish with all suggested cases, we got the following differences in the file size, as shown in *Table 4.4* and in *fig 4.3*:

Table 4.4: *Size comparison by modified blowfish algorithms.*

Encryption Algorithm Used	Output File size(KB)
Original Blowfish	800
Modified Blowfish(case 1)	485
Modified Blowfish(case 2)	485
Modified Blowfish(case 3)	485
Modified Blowfish(case 4)	485

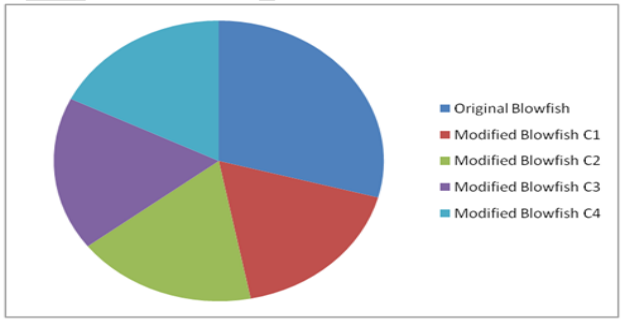

Fig 4.3: *Pie chart for comparison of encrypted file size.*

ACKNOWLEDGMENT

I would like to express my deep sense of gratitude and indebtedness to "Dr. Narendra Singh Yadav"**,** Head Of Computer Science Department, Sri Balaji College Of Engineering and Technology, Jaipur , India , for his invaluable encouragement, suggestions and support throughout the work. Above all, his priceless and meticulous supervision at each and every phase of work inspired me in innumerable ways.

CONCLUSION

The main objective of this thesis is to evaluate the performance of modified Blowfish algorithm in four different cases with different parameters like Encryption Quality, Correlation Coefficients, Key Sensitivity Test and Size of Output File. In all those case we find that we have improved the Original Blowfish algorithm to some extents. The results of all the tests conducted above lead to common conclusion that the security of the modified algorithm with different cases makes the original Blowfish algorithm more compact and more secure than the earlier. Here one more thing we want to conclude that case 4 as suggested is most secured than others.

REFERENCES:

[1] Joan Daemen, Vincent Rijmen, editors, Fast Software Encryption, 9th International Workshop, FSE 2002, Leuven, Belgium, February 4-6, 2002, Revised Papers. Springer,2002. ISBN 3-540-44009-7.

[2] William Stallings, "Cryptography and Network Security", Third Edition, Pearson Education, 2003.

[3] B. Schneier, "Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish)", Fast Software Encryption, Cambridge Security Workshop proceedings (December 1993), Springer-Verlag, pp. 191-204, 1994.

[4] Ashwaq T Hashim" Type-3 Feistel Network of The 128-bits Block Size Improved Blowfish Cryptographic Encryption " IJCSNS International Journal of Computer Science and Network Security, vol.8 No.12, pp. 280-286, December 2008.

[5] G. Chen, Y. Mao, C.K. Chui, "A symmetric image encryption based on 3D chaotic maps", Chaos Solutions and Fractals, vol. 21, pp. 749–761, 2004.

[6] Vincent Rijmen, Bart Preneel, Erik De Win, "On Weaknesses of Non-surjective Round Functions, Designs, Codes and Cryptography", Springer,Volume 12, pp. 253 - 266, 1997.

[7] Hossam El-din H. Ahmed, Hamdy M. Kalash, and Osama S. Farag Allah, "Encryption Quality Analysis of RC5 Block Cipher Algorithm for Digital Images", Journal of Optical Engineering, vol. 45, 2006.

[8] Hossam El-din H. Ahmed, Hamdy M. Kalash. And Osama S. Farang Allah, "Encryption Effeciency Analysis and Security Evaluation of RC6 Block Cipher for Digital Images",International Journal of Computer, Information , and System Science, and Engineering vol. 1, no. 1, ISSN 1307-2331, pp 33-38, 2007.

[9] B. Schneier, "Applied Cryptography – Protocols, algorithms, and source code in C", John Wiley & Sons, Inc., New York, second edition, 1996.

[10] "Blowfish—One Year Later", Dr. Dobb's Journal, September 1995.

[11] Krishnamurthy G.N, Dr. V Ramaswamy, "Performance Enhancement of Blowfish algorithm by modifying its function", Proceedings of International Conference in CISSE, University of Bridgeport, Bridgeport, CT, USA, pp 244-249, 2006.

[12] Krishnamurthy G N, Dr. V Ramaswamy "Encryption quality analysis and Security Evaluation of CAST 128 algorithm and its modified version using digital images", International Journal of Network Security and its Application(IJNSA), vol. 1, No. 1, pp28-33, April 2009.