International Journal of Scientific & Engineering Research, Volume 4, Issue 12, December-2013
ISSN 2229-5518



# Design and Implementation of Hybrid Encryption Algorithm

Ali E. Taki El_Deen

*IEEE Senior Member,* Alexandria University, Egypt.

A_takieldeen@yahoo.com

***Abstract*** – In today's world 99% people are more interested in sending and receiving data through internet and mobile data storage devices. But among those people don't encrypt their data though they know that data contains personal information and the chances of data lose or hacking is very high. Information security has always been important in all aspects of life. It can be all the more important as technology continues to control various operations in our day-to-day life. Cryptography provides a layer of security in cases, where the medium of transmission is susceptible to interception, by translating a message into a form that cannot be read by an unauthorized third party. The ultimate objective of the research presented in this paper is to develop both AES and Blowfish to be low power, high-throughput, real-time, reliable and extremely secure cryptography algorithm and in addition to making estimation of both AES and Blowfish more difficult seems impossible.

***Keywords***: Blowfish, AES, DES, RSA

IJSER

**TABLE OF CONTENTS**



## 1. Introduction

Hybrid Encryption Algorithm is a keyed, symmetric block cipher, designed in 2012. It is a combination of two known algorithms ( Blowfish & AES 128 ) .

Hybrid Encryption Algorithm takes the advantages of blowfish algorithm and Advanced-Encryption-Standard (AES) algorithm makes it harder for any attacker to try to decrypt the cipher text.

Hybrid Encryption Algorithm requires fast processing techniques. Hybrid Encryption Algorithm is a high encryption security.

## 2. The Blowfish Algorithm

Blowfish has a 64-bit block size and a key length of anywhere from 32 bits to 448 bits (32-448 bits in steps of 8 bits; default 128 bits).

It is a 16-round Feistel cipher and uses large key-dependent S-boxes. It is similar in structure to CAST-128, which uses fixed S-boxes.

There are two parts here:

- A part that handles the expansion of the key.
- A part that handles the encryption of the data.

The expansion of the key:

Breaking the original key into a set of subkeys. Specifically, a key of no more than 448 bits is separated into 4168 bytes. There is a P-array and four 32-bit S-boxes. The P-array contains 18 32-bit subkeys, while each S-box contains 256 entries.

The encryption of data:

The encryption of the data: 64-bit input is denoted with an x, while the P-array is denoted with a Pi (where i is the iteration).

Figure1 shows the action of Blowfish. Each line represents 32 bits. The algorithm keeps two subkey arrays: the 18-entry P-array and four 256-entry S-boxes.

The S-boxes accept 8-bit input and produce 32-bit output. One entry of the P-array is used every round, and after the final round, each half of the data block is XORed with one of the two remaining unused P-entries.

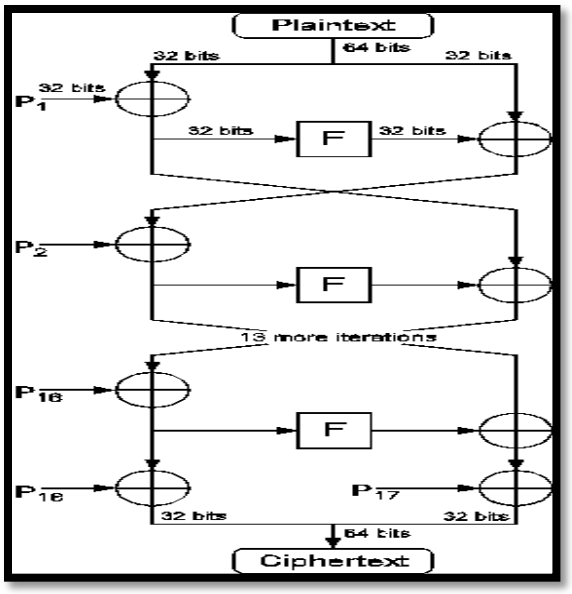



**Fig. 1 Blowfish algorithm**

## Figure explanation:

- Initialize the P-array and S-boxes.
- XOR P-array with the key bits. For example, P1 XOR (first 32 bits of key), P2 XOR (second 32 bits of key). Use the above method to encrypt the all-zero string.
- This new output is now P1 and P2.
- Encrypt the new P1 and P2 with the modified subkeys.
- This new output is now P3 and P4.
- Repeat 521 times in order to calculate new subkeys for the P-array and the four S boxes.

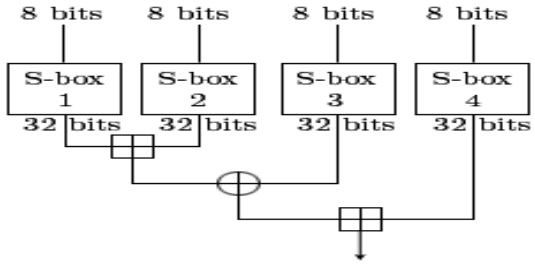



**Fig. 2 Blowfish F function**

## Advantage of Blowfish:

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 [1391].

4. Variably Secure. Blowfish's key length is variable and can be as long as 448 bits.

## 3. 128 - Bits AES Algorithm

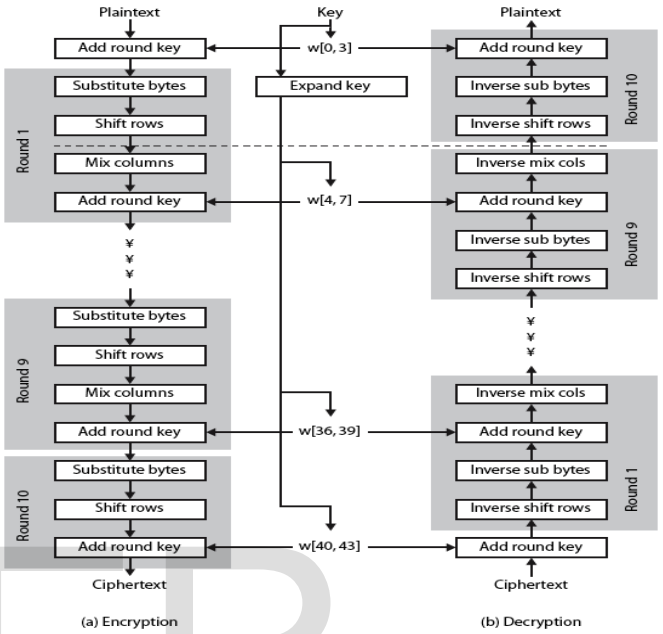



**Fig. 3 AES flowchart**

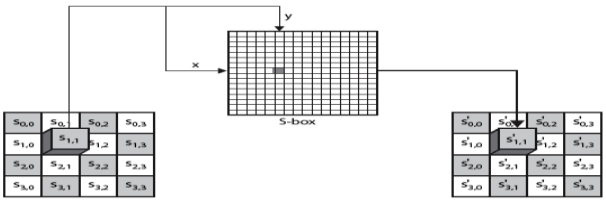
Figure 3 shows the structure of AES in more detail. The cipher consists of N rounds, where the number of rounds depends on the key length: 10 rounds for a 16-byte key; 12 rounds for a 24-byte key; and 14 rounds for a 32-byte key. The first N – 1 rounds consist of four distinct transformation functions: SubBytes, ShiftRows, MixColumns, and AddRoundKey, which are described subsequently. The final round contains only 3 transformations, and there is a initial single transformation (AddRoundKey) before the first round, which can be considered Round 0. Each transformation takes one or more 4 x 4 matrices as input and produces a 4 x 4 matrix as output. Figure 4 shows that the output of each round is a 4 x 4 matrix, with the output of the final round being the ciphertext. Also, the key expansion function generates N + 1 round keys, each of which is a distinct 4 x 4 matrix.

Each round key serves as one of the inputs to the AddRoundKey transformation in each round.

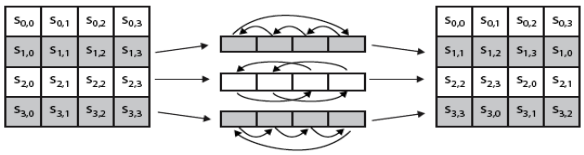
## The main 4 functions in AES:

**1. SubByte**

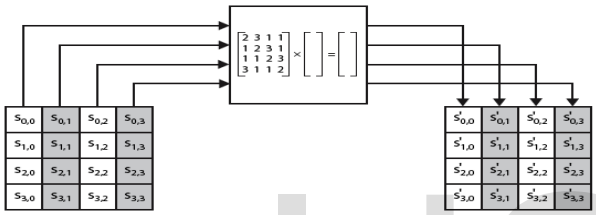
IJSER




**2. Shift Row**



**3. Mix Columns**



**4. Add Round Key**

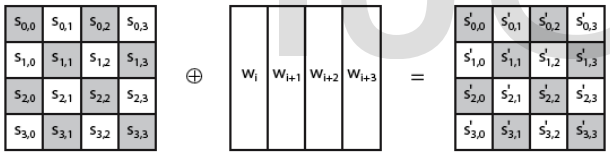


**Fig. 4 The output of each round**

## Advantage of AES:

1. Advanced Encryption Standard (AES) algorithm works on the principle of Substitution Permutation network.
2. AES has more elegant mathematical formulas behind it, and only requires one pass to encrypt data. AES was designed from the ground up to be fast, unbreakable and able to support the tiniest computing devices imaginable. The big differentiators between AES and Triple-DES are not strength of security, but superior performance and better use of resources.
3. Advanced Encryption Standard not only assures security but also improves the performance in a variety of settings such as smartcards, hardware implementations etc.
4. AES is federal information processing standard and there are currently no known non-brute-force direct attacks against AES.
5. AES is strong enough to be certified for use by the US government for top secret information.

## AES Round:

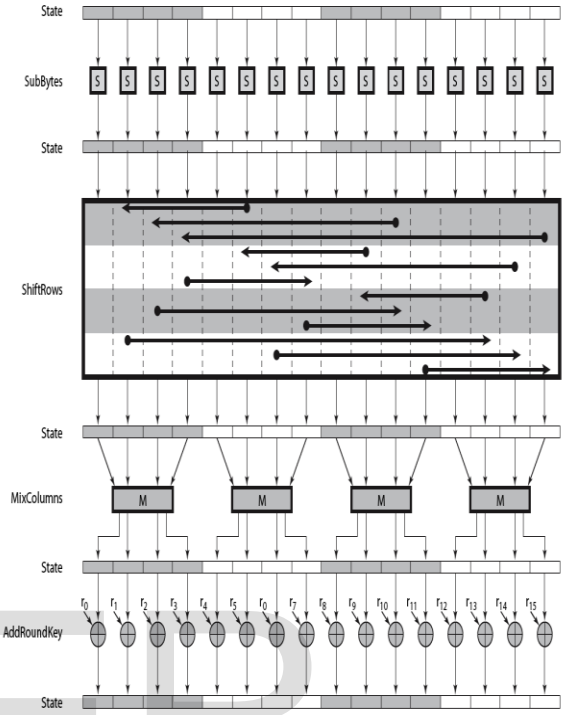



**Fig. 5 AES round**

## 4. Comparison between AES, DES, RSA, and Blowfish

| | Key type | Key size | Block size |
|---|---|---|---|
| **AES** | Symmetric | 128 bits | 128 bits, 192 bits, and 256 bits |
| **DES** | Symmetric | 64 bits (56 bits are actually used) | 64 bits |
| **RSA** | Asymmetric | Not specified | Not specified |
| **Blowfish** | Symmetric | 64 bits | From 32 bits to 448 bits |

**Table 1: Comparison between AES, DES, RSA, Blowfish**

IJSER

## 5. Statistical Tests

Let $s = s_0; s_1; s_2; \ldots\ldots ; s_{n-1}$ be a binary sequence of length n. This subsection presents four statistical tests that are commonly used for determining whether the binary sequence s possesses some specific characteristics that a truly random sequence would be likely to exhibit. It is emphasized that the outcome of each test is not definite, but rather probabilistic. If a sequence passes all four tests, there is no guarantee that it was indeed produced by a random bit generator [19]. These tests are:

- Frequency test (Monobit test).
- Serial test.
- Poker test.
- Run test.

For a significance level of $\alpha = 0.05$, the threshold values for freq., serial, poker, and run tests are 3.8415, 5.9915, 14.0671, and 9.4877 respectively [10]. Our tests results are given in figure 3.

## 6. Hybrid Encryption Algorithm

NOTE: The following is first idea but we still developing our algorithm.

**Statistical tests of Text Data:**

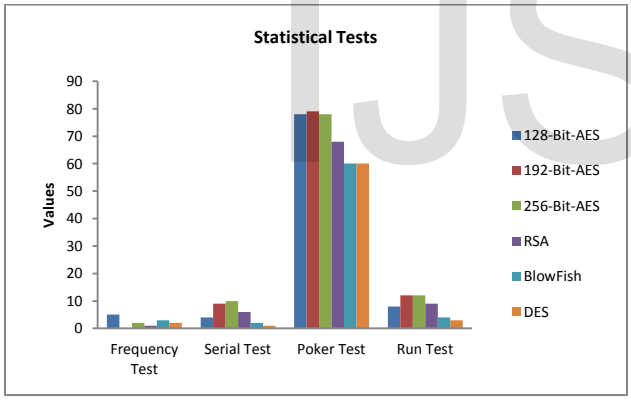



**Fig. 6 Tests values**

## Encryption:

There are four parts to this algorithm:

1. Part that handles the expansion of the key used in blowfish.

2. Part that handles the expansion of the key used in AES.

3. Part that handles the encryption of the data using blowfish.

4. Part that handles the encryption of the encrypted data from blowfish using AES128.

**Part 1:**

Blowfish key: break the original key into a set of subkeys. Specifically, a key of no more than 448 bits is separated into 4168 bytes. There is a P-array and four 32-bit S-boxes. The P-array contains 18 32-bit subkeys, while each S-box contains 256 entries.

**Part 2:**

AES key: expansion of 128 bit only from the key whish will give 10 partial keys used in the initial round, 9 main rounds and one final round.

**Part 3:**

Make the encryption of 128 bit from plain text using blowfish by making encryption to the first 64 bit then to the second 64 bit.

**Part 4:**

Take the output of the encrypted 128 bit that comes from making blowfish twice and make this output the input plain text to AES algorithm.

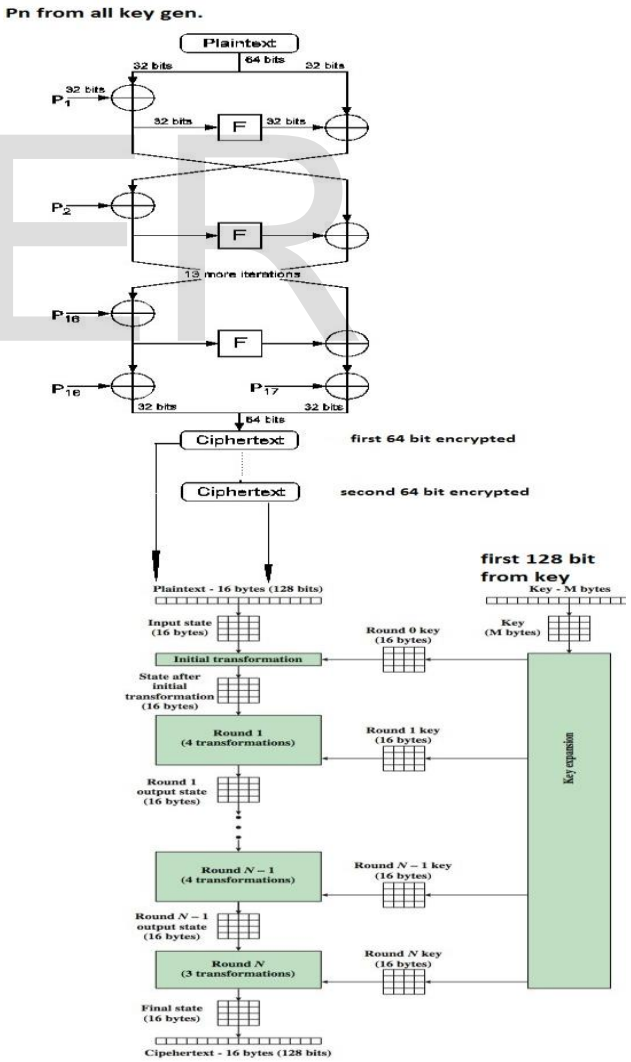



**Fig. 7 Hybrid Encryption Algorithm**

IJSER

## Decryption:

1. Part that handles the expansion of the key used in blowfish.

2. Part that handles the expansion of the key used in AES.

3. Part that handles the decryption using AES128 to the encrypted data using blowfish.

4. Part that handles the decryption of the data using blowfish.

In decryption part one and part two are the same as there is no change in key generation for both blowfish and AES. For part 3 and 4 we will start with part 4 then part 3.

Figure 7 describes the main steps of Hybrid Encryption Algorithm encryption and how it start normally with the input key from user and make blowfish encryption two times to get 128 bit encrypted then we make AES encryption one time to the 128 bit output from the two times blowfish encryption. using only the first 128 bit from key as the key may get too long as we can use up to 448 bit or 576 bit key in blowfish , then finally we get the 128 bit encrypted .

## 7. Contribution of This Paper and Future Work

In this paper, Hybrid Encryption Algorithm has been introduced. The proposed technique of Hybrid Encryption Algorithm combines between difficulty of estimation the original text and verity of using the different key on blowfish and AES encryption that we introduce cipher text more difficult for estimation so our algorithm is at high level of security and we need that to use it in specific applications like military applications, hardware and software companies that need security in their products, banks, networks companies, big websites that have big databases and mobile networks. Also a comparison between AES, DES, RSA, and Blowfish encryption algorithms are discussed. Statistical tests of AES, DES, RSA, and Blowfish algorithms have been examined.

## REFERENCES

**[1]** William Stallings, Network Security Essentials: Applications and Standards, Prentice Hall, 4th edition, 2011.

**[4]** B. Schneier, Speed Comparisons of Block Ciphers On a Pentium, Retrieved 12:04:58, July 27, 2008 from http://www.schneier.com/blowfish-speed.html.

**[5]** B. Schneier, in: Applied Cryptography, second ed., John Wiley & Sons, Inc., New York, 1996.

**[6]** B. Schneier, The Blowfish Encryption Algorithm. Retrieved July 27, 2008 from http://www.schneier.com/blowfish.html.

**[8]** Pieprzyk, J.; Hardjono, T.; and Seberry, J., Fundamentals of Computer Security. New York: Springer-Verlag, 2003.

**[13]** J. Daemen and V. Rijmen, AES Proposal: Rijndael, AES Algorithm Submission,September 3, 1999.

**[14]** A. Lee, NIST Special Publication 800-21, Guideline for Implementing Cryptographyin the Federal Government, National Institute of Standards and Technology,November 1999.

**[15]** J. Nechvatal, et. al., Report on the Development of the Advanced Encryption Standard (AES), National Institute of Standards and Technology, October 2, 2000.

**[17]** Wenbo Mao, Modern Cryptography: Theory and Practice: By Hewlett-Packard Company, Publisher: Prentice Hall PTR, Pub Date: July 25, 2003, ISBN: 0-13-066943-1.

**[7]** Avi Kak, "AES: The Advanced Encryption Standard, Lecture Notes on "Computer and Network Security"", February, 2013.

## Biography

***Ali E. Taki El_Deen*** *(IEEE Senior Member) received the PhD degree in Electronics and Communications Engineering in "Encryption and Data Security in Digital Communication Systems". He has a lot of publications in various international journalsand conferences. His current research interests are in multimedia processing, wireless communication systems, Microcontroller and Field Programmable Gate Array (FPGA) applications.*

IJSER