CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

# Cifrador de Bloque con Alternancia de Matrices de Permutación y Combinación de Algoritmos Cifradores Paralelos

Andrés Francisco Farías – Andrés Alejandro Farías

Departamento Académico de Ciencias Físicas, Matemáticas y Naturales
Universidad Nacional de La Rioja, La Rioja. Argentina
afarias665@yahoo.com.ar, andres_af86@hotmail.com

**Abstract.** Cifrador de bloque, desarrollado sobre la estructura de una red de Feistel, de tamaño de bloque de 256 bits, con clave de 128 bits y vector de inicialización de 256 bits, con alternancia de matrices de permutación y combinación de algoritmos cifradores paralelos, con modo de encadenamiento de bloques de cifrado de propagación (PCBC, Propagating Cipher Block Chaining).

El primer algoritmo contiene un generador binario pseudoaleatorio que está compuesto por cuatro Linear Feedback Shift Registers (LFSR), de distintas longitudes, cuyas secuencias se combinan mediante una operación de suma con acarreo. El segundo algoritmo contiene un generador binario pseudoaleatorio que está conformado por tres LFSR, de distintas longitudes, cuyas secuencias se combinan mediante una suma con acarreo.

Los algoritmos trabajan en paralelo y sus secuencias se combinan mediante una operación XOR.

Los LFSR de cada uno de los generadores contienen funciones booleanas de filtrado no lineal de cuatro variables. Las funciones fueron seleccionadas por sus propiedades criptográficas tales como: ser balanceadas, alto grado algebraico, cumplir con el Criterio de Avalancha Estricta (SAC, sigla en inglés) y tener alta no linealidad. Finalmente, el texto cifrado obtenido fue sometido a un conjunto de pruebas estadísticas de aleatoriedad.

**Keywords:** LFSR, cipher, key, boolean function, non-linearity

## 1. Introducción

El presente documento expone el desarrollo de un cifrador de bloque, basado en una Red de Feistel que permite el cifrado y descifrado utilizando la misma estructura, donde para el caso del descifrado se utilizan las subclaves cambiando el orden de las mismas [1], [2]. La clave adoptada es de 16 caracteres, es decir 128 bits.

## 2. Esquema del cifrador

El cifrado de bloque se denomina así por realizar el proceso de encriptación trabajando sobre cadenas de texto de igual longitud. En este caso se utilizaron bloques de 256 bits. Luego esos bloques son ensamblados siguiendo el modo de encadenamiento de bloques de cifrado de propagación (Propagating Cipher Block Chaining. (PCBC)). Básicamente la estructura del cifrador está conformada por; Red

de Feistel que consta de: Red de Feistel para cifrado, Red de Feistel para descifrado, Clave y subclaves, Vector de inicialización, Algoritmos de cifrado, Generadores binarios pseudoaleatorios, Modo de operación PCBC, Matrices de Permutación

## 3. Red de Feistel para cifrado

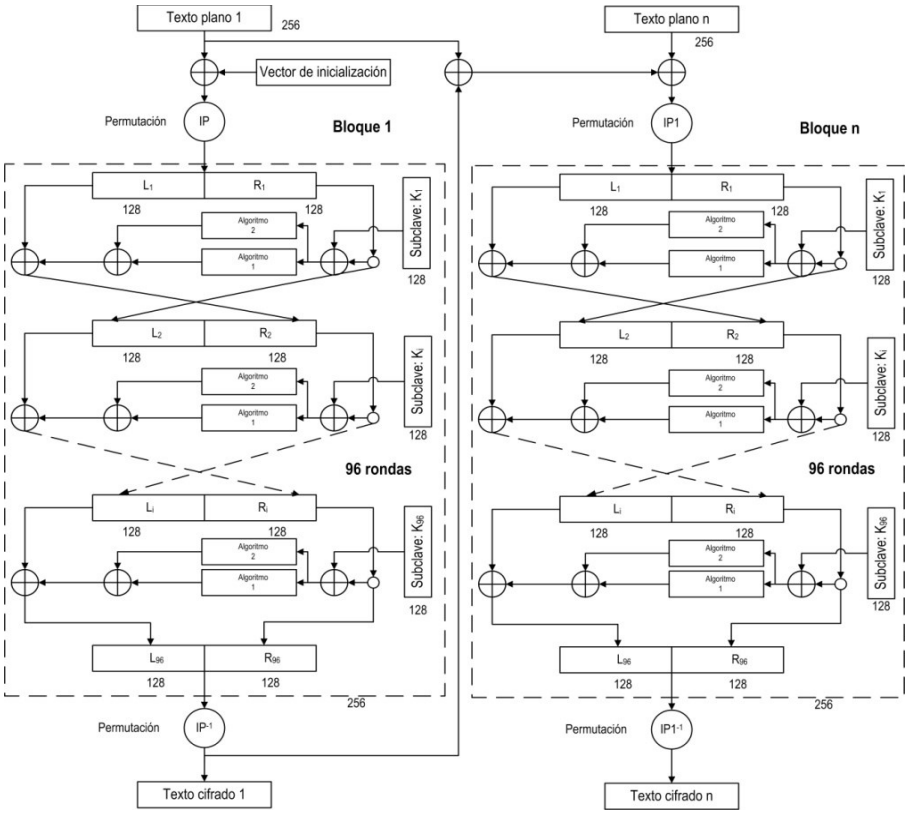


**Fig. 1**. Red de Feistel para cifrado

El proceso de cifrado consiste en dividir el texto plano en bloques de 256 bits, el primer bloque es sometido a una operación XOR con el vector de inicialización, luego al resultado se le realiza una permutación IP.

La salida de la permutación entra en la red de Feistel, que se detalla en la figura 1 y se producen 96 rondas, con sus respectivas subclaves, después se realiza una permutación IPI, para obtener el primer bloque de texto cifrado.

Para los siguientes bloques de texto plano, se realiza una operación XOR con los bloques de texto plano y cifrado del primer bloque y al resultado se le ejecuta una nueva operación XOR con el texto plano del bloque y la salida sufre una permutación IP1 (la otra matriz de permutación) antes de entrar a la red de Feistel y producir 96 rondas, con las subclaves correspondientes..

Después de esta operación se calcula la permutación IPI1 (la otra matriz de permutación inversa) y se consigue un nuevo bloque de texto cifrado. Las matrices de permutación se van alternando de un bloque a otro.

## 4. Red de Feistel para descifrado

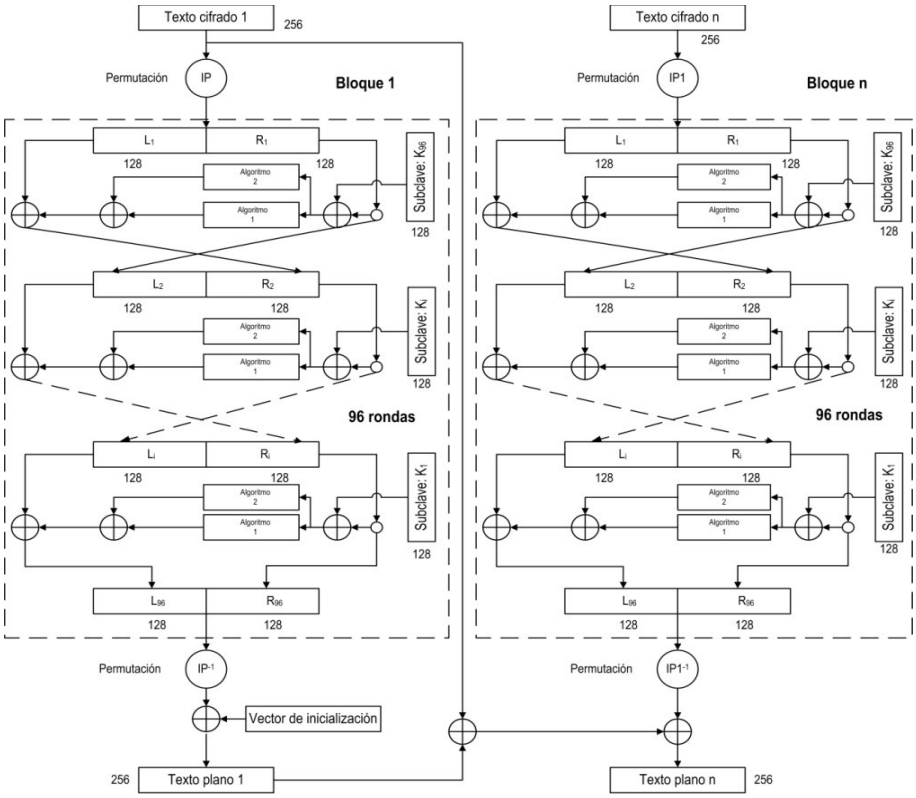


**Fig. 2**. Red de Feistel para descifrado

La Red de Feistel para descrifrado es similar a la anterior, pero en este caso se toma el texto cifrado y se lo divide en bloques de 256 bits.

Para el primer bloque de texto cifrado se realiza una permutación IP antes de entrar a la red de Feistel y realizar 96 rondas, con las claves introducidas en modo inverso, al resultado se le realiza una permutación IPI y luego se produce una operación XOR con el vector de inicialización para obtener el primer bloque de texto plano.

Para el resto de los bloques de texto cifrado, el proceso comienza con la permutación IP1, después se ingresa a la red de Feistel y se llevan a cabo 96 rondas, con las subclaves ingresadas en modo inverso.

Finalmente después de este proceso se hace una permutación IPI1 y a la salida se le aplica una operación XOR con la resultante de la operación XOR entre el texto cifrado

y texto plano del bloque anterior, para lograr un nuevo bloque de texto plano. Las matrices de permutación se van alternando de un bloque a otro.

## 5. Clave y subclaves

Como se dijo previamente, la clave está conformada con 16 caracteres (128 bits), de la que se generan 96 subclaves de 128 bits, siguiendo los pasos que se muestran en la figura 3.

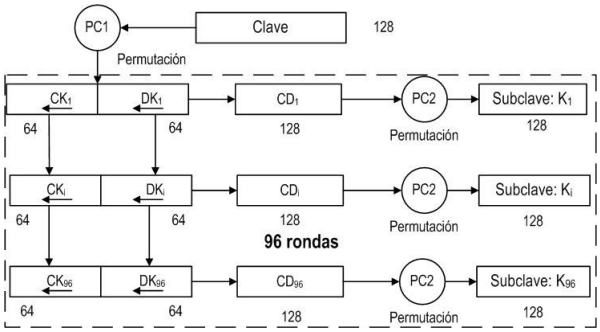


**Fig. 3**. Tratamiento de las subclaves

La clave es sometida a una permutación según la matriz de permutación PC1; luego se divide el bloque de 128 bits resultante en dos bloques de 64 bits, los que sufren desplazamiento de las posiciones de los bits de manera de tener 96 pares de bloques de 64 bits que corresponderán a las 96 subclaves.

Esos pares son ensamblados y luego sometidos a la permutación PC2, para obtener las 96 subclaves finales.

## 6. Vector de inicialización

Es para iniciar las tareas de encadenamiento de bloques, tanto de cifrado como de descifrado. Es única para todo el proceso, debe ser secreta como la clave y su longitud es igual a la de los bloques: 256 bits.

## 7. Algoritmos de cifrado

Los algoritmos de cifrado tienen la configuración que se indica en la figura 4, pero difieren en el generador binario pseudoaleatorio.que es distinto para cada uno.

Tienen una entrada de 128 bits, que conforman los estados iniciales para los LFSR del generador, que una vez cargados, realizan 128 ciclos, entregando 128 bits de salida.

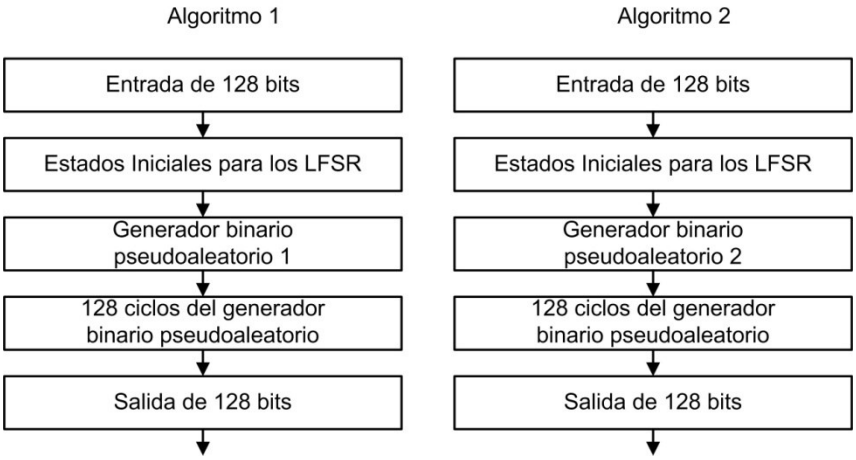


**Fig. 4**. Algoritmos de cifrado

## 8. Generadores binarios pseudoaleatorios

Está conformado por cuatro y tres LFSR con funciones booleanas de filtrado no lineal de cuatro variables, balanceadas y de alta no linealidad. Luego se combinan, mediante suma con acarreo, para obtener la secuencia final.

### 8.1 Características de los LFSR

Los LFSR que se utilizan tienen la siguiente estructura, un LFSR de base con un polinomio de conexión primitivo que produce el cálculo de la retroalimentación. En paralelo tenemos una función booleana de filtrado no lineal de cuatro variables, que se alimenta con cuatro valores de los registros del LFSR. La salida de la función y la retroalimentación del polinomio se someten a una operación XOR para obtener la retroalimentación definitiva del LFSR.

### 8.2 Definición de los generadores

Para los dos generadores se adoptó la misma estructura, cambiando entre ellos las longitudes de los LFSR y las funciones booleanas de filtrado no lineal.

Básicamente se utilizan para el primer generador cuatro LFSR con funciones booleanas de filtrado no lineal, cuyas secuencias se combinan mediante suma con acarreo, figura 5

Para el segundo generador se dispone de tres LFSR, también con funciones booleanas de filtrado no lineal, de los que se obtiene un secuencia mediante la aplicación de la suma con acarreo, figura 5

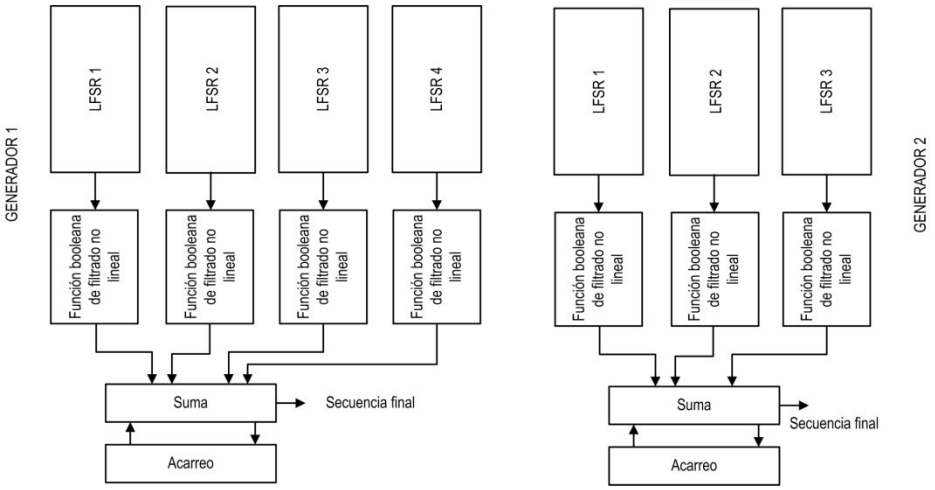


**Fig. 5.** Esquema generador binario pseudoaleatorio 1 y 2

### 8.3 Elección de los distintos LFSR

En tabla se indican las longitudes y polinomios primitivos de cada uno de ellos [3], [4], [5].

**Tabla 1.** LFSR, longitudes y polinomios primitivos del generador

| LFSR | Longitud | Polinomios primitivos |
|---|---|---|
| 1 | 71 | $P(x)_1 = x^{71} + x^{59} + x^{53} + x^{48} + 1$ |
| 2 | 67 | $P(x)_2 = x^{67} + x^{61} + x^{33} + x^3 + 1$ |
| 2 | 61 | $P(x)_3 = x^{61} + x^{44} + x^{19} + x^{15} + 1$ |
| 3 | 59 | $P(x)_4 = x^{59} + x^{54} + x^{46} + x^{26} + 1$ |

### 8.4 Propiedades criptográficas deseables

A continuación se indican algunas de las propiedades criptográficamente más significativas, adoptadas para este trabajo [6], [7], [8].

- Función Balanceada:
- No Linealidad:
- Grado Algebraico:
- SAC: Tabla de funciones para los generaadores

Siguiendo los criterios arriba indicados las funciones booleanas aceptadas, para los generadores, por tener buenas propiedades criptográficas, se enumeran a continuación:

**Tabla 2.** Funciones aceptadas para el generador 1

| $f_{NAF}$ | |
|---|---|
| $f_{5775} = a \oplus b \oplus a \cdot b \oplus a \cdot c \oplus a \cdot d$ | $f_{1550} = b \oplus a \cdot b \oplus a \cdot c \oplus b \cdot d \oplus c \cdot d$ |
| $f_{4529} = a \oplus c \oplus a \cdot c \oplus b \cdot c \oplus c \cdot d$ | $f_{1579} = b \oplus a \cdot b \oplus a \cdot c \oplus a \cdot d \oplus b \cdot c \cdot d$ |

$f_{2402} = b \oplus a \cdot c \oplus b \cdot c \oplus d \oplus c \cdot d$ $\quad$ $f_{1585} = b \oplus a \cdot b \oplus a \cdot c \oplus d \oplus a \cdot d$

$f_{3338} = a \oplus a \cdot b \oplus c \oplus b \cdot c \oplus b \cdot d$

En figura 6, se indica la conformación final de las generadores 1 y 2.

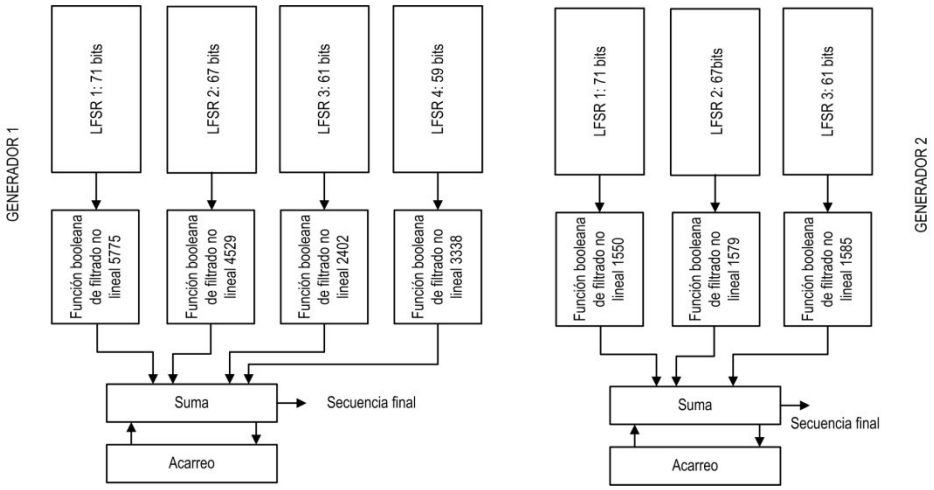


**Fig. 6**. Esquema generador pseudoaleatorio 1 y 2

## 9. Modo de operación:

Modo de encadenamiento de bloques de cifrado de propagación (Propagating Cipher Block Chaining.(PCBC))

## 10. Matrices de Permutación

Se recurre a matrices distribución aleatoria de las posiciones, para obtenerlas se utiliza un generador de números aleatorios, en esta ocasión se adopta un generador congruencial multiplicativo[7]. El generador tiene la siguiente expression

$$x_{i+1} = (a_x \cdot x_i)mod\ m_x$$

Donde: $a_x = multiplicador\ \ m_x = módulo\ \ x_0 = semilla$

**Tabla 3.** Tabla.6. Matriz IP

| Matriz | módulo | multiplicador | semilla |
|---|---|---|---|
| IP | 1048576 | 2011 | 2047 |
| IP1 | 1048576 | 3421 | 3571 |
| PC1 | 1048576 | 1297 | 1229 |
| PC2 | 1048576 | 1789 | 1663 |

## 11. Pruebas de aleatoriedad

### 11.1 Elección de las pruebas estadísticas

Fueron seleccionadas algunas pruebas de la Norma NIST Special Publication 800-22, del trabajo de Rukhin (et al.) [9]: Prueba de frecuencia, Prueba de frecuencia en un bloque, Prueba de rachas, Prueba de rachas de unos en un bloque, Prueba de sumas acumuladas, Prueba de entropía aproximada

### 11.2 Pruebas sobre el generador

Se analizaron cien secuencias binarias de 100.000 bits, obtenidas del generador a partir de cien claves distintas. El nivel de significancia adoptado para las pruebas estadísticas es de $\alpha = 0{,}01$. La hipótesis nula es: $H_0 \rightarrow p\_valor > 0{,}01$

### 11.3 Proporción de muestras que pasan las pruebas

Para el análisis de los resultados, se determina la proporción de muestras que superan las pruebas, y con esos datos se construye un gráfico de puntos, luego se verifica si los mismos caen dentro de los límites superior e inferior, donde k es el número de muestras. $LS, LI = (1-\alpha) \pm 3 \cdot \sqrt{\alpha \cdot (1-\alpha)/k}$

En nuestro caso $k = 100$ y el nivel de significancia elegido es: $\alpha = 0.01$, los límites quedan: $LS = 1{,}02$ y $LI = 0{,}96$

Se consideran todas pruebas, los resultados se indican en la tabla.

**Tabla 4.** Pruebas

| Pruebas | Proporción | Superior | Inferior |
|---|---|---|---|
| Frecuencias | 1,00 | 1,02 | 0,96 |
| Frecuencias en un Bloque | 0,99 | 1,02 | 0,96 |
| Rachas | 1,00 | 1,02 | 0,96 |
| Rachas de Unos en un Bloque | 0,99 | 1,02 | 0,96 |
| Sumas Acumuladas Adelante | 1,00 | 1,02 | 0,96 |
| Sumas Acumuladas Atrás | 1,00 | 1,02 | 0,96 |
| Entropía Aproximada | 1,00 | 1,02 | 0,96 |

En figura 7, se aprecia el resultado, en definitiva, la secuencia que entrega el generador supera las pruebas de aleatoriedad.

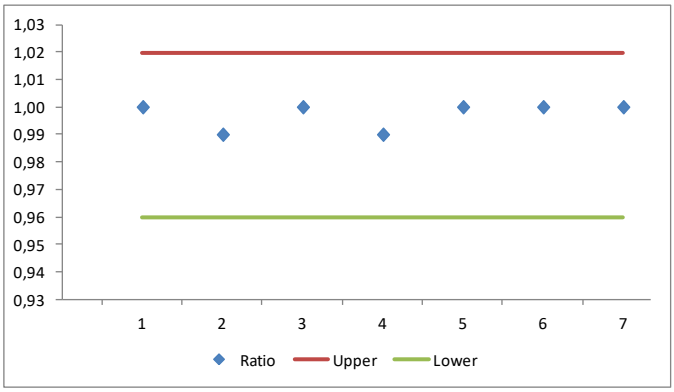


**Fig. 7.** Gráfico de puntos

## 12. Comparación de frecuencias

Superposición de gráficos de frecuencias para observar las diferencias entre texto plano y texto cifrado, figura 8.

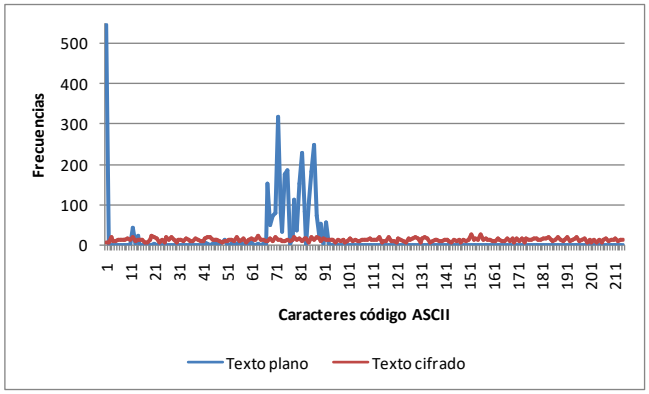


**Fig. 8.** Frecuencias de caracteres del texto plano y cifrado

## 13. Conclusiones

Un cifrador de bloque de 256 bits, con algunas propiedades importantes tales como clave de 128 bits y vector de inicialización de 256 bits y la incorporación de algoritmos de cifrado que contienen generadores binarios pseudoaleatorios.

Para futuras versiones se pueden incorporar entre otras cosas: claves más largas y mayor cantidad de algoritmos y otros métodos de concatenación de bloques.

El resultado obtenido del texto cifrado tiene una frecuencia de caracteres aleatorios, lo que hace difícil un criptoanálisis basado en la estadística de aparición de caracteres.

## Referencias

1. Karakoç, F., Demirci, H., Harmanc, A.: AKF: A Key Alternating Feistel Scheme for Lightweight Cipher Designs, Information Processing Letters. 115, 359--367 (2015)
2. Bogdanov, A.: Analysis and Design of Block Cipher Constructions. Fakultat fur Elektrotechnik und Informationstechnik an der Ruhr-Universitat Bochum (2009)
3. Clark, J., Jacob, J., Maitra, S., Stanica, P.: Almost Boolean Functions: The Design of Boolean Functions by Spectral Inversion. Computational intelligence. 20. (3), 450—462 (2004)
4. Braeken, A.: Cryptographic Properties of Boolean Functions and S-Boxes. Faculteit Ingenieurswetenschappen. Katholieke Universiteit Leuven (2003)
5. Elhosary, A., Hamdy, N., Farag, I., Rohiem, I.: State of the Art in Boolean Functions Cryptographic Assessment. International Journal of Computer Networks and Communications Security.1. (3), 88--94 (2013)
6. García Méndez, P.: Descripción Polinomial de los Sistemas de Cifrado DES y AES. Universidad Autónoma Mexicana, México (2011)
7. Fishman, G.: Multiplicative Congruential Random Number Generators with Modulus 2ß: An Exhaustive Analysis for ß = 32 and a Partial Analysis for ß = 48. Mathematics of Computation. 54. (189), 33--344 (1990)
8. Duta, C., Mocanu, B., Vladescu, F., Gheorghe, L.: Randomness Evaluation Framework Of Cryptographic Algorithms. International Journal on Cryptography and Information Security (IJCIS), Vol. 4, No. 1, (2014)
9. Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., and Vo, S., “A Statistical Prueba Suite for Random and Pseudorandom Number Generators for Cryptographic Applications”, National Institute of Standards and Technology, (2000).