Nexo Revista Científica
Vol. 36, No. 04, pp. 470-481/Septiembre 2023

ISSN-E 1995-9516
Universidad Nacional de Ingeniería

http://revistas.uni.edu.ni/index.php/Nexo
https://doi.org/10.5377/nexo.v36i04.16750

Uso del caos en cronogramas de llaves para un algorítmo de cifrado simétrico sin pérdida de información

Use of chaos in key schedules for a symmetrical encryption algorithm without data loss

Marlon David González-Ramírez [1,*], Eduardo Vega-Alvarado [1] , Arturo Morales-Rangel [2]

[1] Instituto Politécnico Nacional. Centro de Innovación y Desarrollo Tecnológico en Cómputo. Ciudad de México, México.
[2] BinaryCore. San Francisco Ca., USA.
[*]dgonzalezr@ipn.mx

(recibido/received: 5-mayo-2023; aceptado/accepted: 4-agosto-2023)

RESUMEN

Esta investigación presenta la construcción de un algoritmo de cifrado simétrico para imágenes de 512×512 pixeles, a color y escala de grises sin pérdida de información y sin compresión, principalmente en formato BMP. Este algoritmo consiste en 15 rondas con intercambio de cajas (S-box), permutaciones, operaciones X-OR y llaves del tamaño de la imagen. Se utilizó la operación de mapa logístico para la generación de S-boxes y para construir un cronograma de llaves del tamaño de la imagen se utilizaron las ecuaciones de E. Lorenz, números trascendentales y la curva elíptica. Para medir la fortaleza del resultado cifrado, se utilizó la entropía de la información y la prueba de bondad $\chi 2$ obteniendo resultados favorables y muy cercanos a algoritmos similares.

Palabras claves: Cifrado de imágenes; caos; curva elíptica, SPN.

ABSTRACT

This research presents the construction of a symmetric encryption algorithm for images of 512×512 pixels, in color and grayscale without loss of information and without compression, in BMP format. This algorithm consists of 15 rounds with S-box swapping, permutations, X-OR operations, and image size keys. The logistic map operation was used for the generation of S-boxes and to build a schedule of keys of the size of the image, the equations of E. Lorenz, transcendental numbers and the elliptic curve were used. To measure the strength of the encrypted result, the entropy of the information and the goodness test $\chi \wedge 2$ were used, obtaining favorable results and close to similar algorithms.

Keywords: Image encryption; chaos; elliptic curve, SPN.

1. INTRODUCCIÓN

Este trabajo presenta un algoritmo de protección de la información mediante el uso de cifrado de datos, especialmente imágenes de 512×512 pixeles sin perdida y sin compresión por las siguientes razones: la NOM-151 en México determina cómo se deben procesar y preservar documentos digitales (Lone & Qureshi, 2022; *Vista de Manual de Digitalización de Documentos*, n.d.) ya que existen imágenes con contenido sensible como lo son en el área de la medicina, militar, finanzas, etc. Aunque investigaciones destacan el uso y manejo del formato JPG por su uso común, por su característica del formato hay perdida de datos (Gupta et al., 2023; Vaseghi et al., n.d.), añadiendo que haya algoritmos que comprimen datos (Zhang et al., 2022). Bajo la primera razón este desarrollo se centra en cifrar imágenes en formato BMP y sin compresión de datos, esto permitirá proteger los mencionados objetos, así como lo mencionan los siguientes desarrollos: (Al-Kadei, 2020; Budiman et al., 2020; Krishnan Raghupathy et al., 2019). Este algoritmo es simétrico del tipo Substitution Permutation Network (SPN) el cual consta de 15 rondas, cajas de sustitución (S-box) y un cronograma dinámico de llaves y será llamado SecCaos-Image por las siguientes razones:

- Si bien, existe una base matemática para la propuesta de SPN, este puede adecuarse con diversas innovaciones como es el uso del de ciertas herramientas en el desarrollo y construcción de sus S-boxes, cronograma de llaves e incluso permutaciones.
- SecCaos-Image usa la ecuación de mapa logístico para la construcción de sus S-boxes y parte de SPN para su constitución. En el caso de (Khan et al., 2019) plantea a SPN y maneja la ecuación de mapa logístico para la construcción de las cajas y el cronograma de llaves. Para las S-box, existen algunos otros casos aplican el mapa caótico (Alanazi et al., n.d.).
- Para este desarrollo, las ecuaciones de E. Lorenz son utilizadas en este perfeccionamiento del cronograma de llaves del tamaño de la imagen y en conjunto con números trascendentes como e y la curva elíptica. Algunos otros trabajos importantes estas herramientas fueron aplicadas de manera cercana o para desarrollar algún otro módulo de su algoritmo (Ahuja et al., 2023; Ramzan et al., 2021; Zou et al., 2020).
- Para verificar la resistencia de los productos cifrados de aplicó análisis de entropía de la información en las llaves y en las imágenes, la correlación de las imágenes, la chi cuadrada χ^2, así como un análisis a las S-boxes.

Se propone que SecCaos-Image sea simétrico basado en rondas, con la capacidad de apostar por la incertidumbre, ya que, como se sabe el algoritmo asimétrico RSA tiene la vulnerabilidad que por medio del cálculo de un número primo por medio del algoritmo short aplicado en un análisis post-cuántico, es posible de computar manera eficiente (Rass & Schartner, 2020), de los contrario, hasta el momento los algoritmos de cifrado simétrico aún muestran resistencia a este tipo de cálculos.

2. CONSIDERACIONES GENERALES

2.1. Red SPN

Un algoritmo SPN es una red que consiste en dividir un mensaje en texto claro en bloques de bits de tamaño igual (2^n); en este caso constará de 15 rondas para llevar a cabo las sustituciones y las permutaciones en cada bloque. En cada ronda *r* se usa una llave *k* del conjunto de llaves *K* y una función de expansión *E*. La lista de operaciones de $E_k(K)$ será igual a $K^1, \ldots K^{n_r+1}$que dará como resultado un algoritmo público (Civino et al., 2019).

Para este caso, se proponen dos permutaciones: una caja de sustitución S-box diferente para cada proceso de cifrado denominada π_s; π_p, la cual permuta las posiciones de los bits de cada bloque, donde *l* y *m* son enteros positivos, así como se muestra en la Ec. (1).

$$\pi_s: \{0,1\}^l \rightarrow \{0,1\}^l$$
$$\pi_p: \{1, \dots, lm\} \rightarrow \{1, \dots, lm\} \quad (1)$$

De esta manera, el texto claro de la Ec. (2), tiene longitud *lm*, y *x* se interpreta como una concatenación de *m* cadenas de bits, y cada cadena contiene *l* bits.

$$x = (x_1, \dots, x_{lm}) \quad (2)$$

La Ec. (3), representa a $x(i)$:

$$x(i) = x(i-1)l + 1, \dots, xil) \quad (3)$$

2.2. Caos

Se aplica el caos en este algoritmo para desarrollas el caos en las S-box y en las llaves; en el primer caso se usa la ecuación de mapa logístico definido en la Ec. (4) (Murillo-Escobar et al., 2020):

$$x_{n+1} = r \times x_n(1 - x_n) \quad (4)$$

Donde *r* tiene el valor de 3.8817182818…, y se aplica con una longitud de más de 300 decimales. El rango de la variable *x* es $0 < x < 1$. Esto satisface que x_n sea determinista, a su vez, cualquier cambio para *r* o x_0. En pocas palabras, x_n no se puede predecir sin los cálculos previos.

Por otro lado, las ecuaciones diferenciales de E. Lorenz se aplican para construir el conjunto de llaves del tamaño de la llave en pixeles. Este sistema de ecuaciones se muestra en la Ec. (5) (Murillo-Escobar et al., 2020):

$$\begin{aligned} \frac{dy}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= rx - y - xz \\ \frac{dy}{dt} &= -bz + xy \end{aligned} \quad (5)$$

Los valores de los parámetros $\sigma, r\ y\ b$, determinan el comportamiento caótico. SecCaos-Image usa los siguientes valores mostrados en la Ec (6):

$$\begin{aligned} \frac{dy}{dt} &= -10(y - x) \\ \frac{dy}{dt} &= rx - y - xz \\ \frac{dy}{dt} &= xy - \frac{8z}{3} \end{aligned} \quad (6)$$

Este sistema de ecuaciones se solucionó tomando sus ángulos en radianes para aplicar el número e. Para dar resolver a la Ec (6), e se eleva a una potencia natural; finalmente, los puntos a generar de la curva elíptica se aplican para obtener un resultado a la ecuación final de este sistema.

2.3. Entropía

La entropía de la información es una medida que se aplica para medir la calidad del cifrado y se calcula según la Ec. (7) (Davies et al., 2022; Murillo-Escobar et al., 2020; Ray & Chattopadhyay, 2021):

$$H(x) = -\sum_{x\varepsilon X} P(x) log_2 P(x) \tag{7}$$

La imagen es tratada como una matriz, donde se extraen los píxeles en formato de colores RGB (rojo, verde y azul); cada color será un Byte, ∴ , se tienen 2^n valores de color. Por su naturaleza de la imagen digital, n=8 con 256 valores en total. Si la probabilidad de ocurrencia de todos los eventos es 1, el valor máximo de entropía es 8. En la práctica, se busca una entropía cercana a 8 si la distribución de bits es uniforme, sin embargo, puede significar que no sean aleatorios por lo que se propone sustentar con más instrumentos de medición.

2.4. Correlación

Medición estadista que se realiza entre dos bits aleatorios. Se utiliza para medir el índice de relación entre estas variables. Este análisis se realiza en 3 direcciones: horizontal, vertical y diagonal. Se toma un pixel (RGB) de manera aleatoria entre los valores de 0 y 255. La representación de estos es de la siguiente manera: Y_r para rojo; Y_v para verde; Y_a para azul. Se obtiene el píxel adyacente al previamente seleccionado y se compara en las tres direcciones mencionadas. Estos valores se representan como: Z_r, Z_v y Z_a para rojo, verde y azul respectivamente. La Ec. (8) (Davies et al., 2022) representa la correlación:

$$r_{k;Y_r,Z_{r,}} = \frac{\frac{1}{N}(\sum_{i=1}^{N}(Y_{i,r}-\bar{Y}_r)(Z_{i,r}-\bar{Z}_r))}{\sqrt{\frac{1}{N}(\sum_{i=1}^{N}(Y_{i,r}-\bar{Y}_r)^2)(\frac{1}{N}\sum_{i=1}^{N}(Z_{i,r}-\bar{Z}_y)^2}} \tag{8}$$

2.5. Prueba de bondad

La prueba de bondad chi-cuadrada χ^2 es un estudio que determina si dos o más elementos son normales. Esta prueba muestra si los valores de las intensidades de los colores (RGB) tienen una distribución uniforme. En caso de ser así, la distribución es aleatoria. En este caso se debe cumplir que la $\chi^2 < 308$ para que el producto cifrado sea aleatorio. La Ec. (9) muestra la χ^2 donde O_i es el valor observado; exp_i es la cantidad esperada (Davies et al., 2022; Ray & Chattopadhyay, 2021).

$$\chi^2 = \sum_{i=1}^{i=k}(\frac{O_i - exp_i}{exp_i})^2 \tag{9}$$

3. ELEMENTOS DE CIFRADO

3.1. Algoritmo de permutaciones.

Se prone el uso del algoritmo de generación de permutaciones usado en (Garcia et al., 2019). Se parte de un número entero no negativo $m \geq 2$ con los siguientes conjuntos $N_m = \{n\ \varepsilon\ N\} |\ 0\ \leq n\ \leq m! - 1$ y $\Pi_m = \{\pi\}$. Se destaca que π es una permutación del arreglo 0,1,…,*m*-1. Como se explica en la referencia anterior, se aplica el algoritmo de la división de Euclides, dado que $n\ \varepsilon\ N$, para obtener la Ec. (10):

$$n = C_0(m-1)! + C_1(m-2)! + \cdots + C_{m-2}(1)! + C_{m-1}(0)! \tag{10}$$

Con base en *m*, $(m-1)!, (m-2)!, \ldots, 1!, 0!$ son fijos.

Se muestra un ejemplo de la permutación cuando m=8, $N_8 = \{n\ \varepsilon\ N\} |\ 0\ \leq n\ \leq 8! - 1$, y la permutación $\Pi_m = \{\pi | \pi\ es\ una\ permutación\ del\ arreglo\ 0,1, \ldots, 7\}$. n será 21699, entones se expresa la Ec. (11):

$$21699 = 4(7!) + 2(6!) + 0(5!) + 4(4!) + 0(3!) + 1(2!) + 1(1!) + 0(0!) \quad (11)$$

Se obtiene que $C_0 = 4$; $C_1 = 2$; $C_2 = 0$; $C_3 = 4$; $C_4 = 0$; $C_5 = 1$; $C_6 = 1$ y $C_7 = 0$, entonces n = 42075163. Esto se representa en la Tabla 1.

Tabla 1. Tabla de permutaciones

$C_0 = 4$	$C_1 = 2$	$C_2 = 0$	$C_3 = 4$	$C_4 = 0$	$C_5 = 1$	$C_6 = 1$	$C_7 = 0$
0	0	0	5	5	3	3	3
1	1	1	1	1	1	6	
2	2	6	6	6	6		
3	3	3	3	3			
4	7	7	7				
5	5	5					
6	6						
7							

Fuente: (Garcia et al., 2019)

3.2. Construcción de una S-box

Se parte de la Ec. (4) con valores para $r = 3.88171$ ..., con una longitud de 313 dígitos. El procedimiento es el siguiente:

- x_0 es el valor aleatorio de inicio con valores entre 0< x_0<1.
- Se itera hasta n=10000, donde n <1. Este número otorga un conjunto de decimales que no siguen un patrón. Para cada iteración, se toman mil números en formato hexadecimal.
- Se calculan las constantes de la Ec. (10), donde $C_i = b_i$, donde b_i es el valor asociado a un bloque (byte) después del punto decimal.
- Se aplica el siguiente algoritmo:

Algoritmo 1. Generación de Permutaciones

```
input : Image size h
apply X[0] = 0. . .X[h − 1] = h − 1
for i ← 0 to h − 1 do
    if i = h − 1 then
        Y [i] = X[Ci]
    end
    else if Ci = h − 1 − i then
        Y [i] = X[Ci]
    end
   else  Y[i] = X[Ci]
       X[Ci] = X[h − 1 − i]
   end
end
end
output Y [i]
```

3.3. Procedimiento de Cifrado

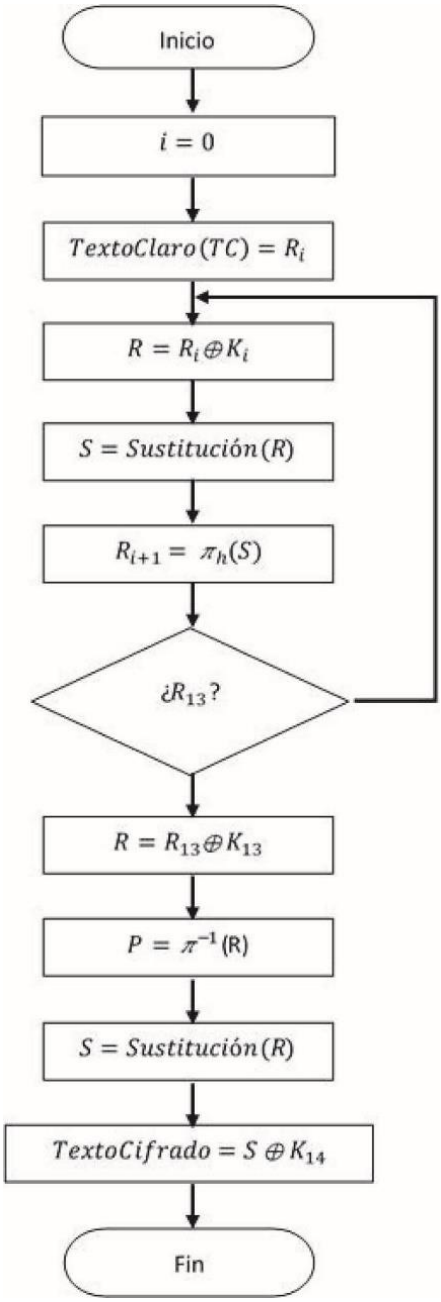

Figura 1. Proceso de Cifrado.

La Figura 1, muestra el proceso de cifrado que trata en descomponer una imagen en una matriz, donde se toman los pixeles en el formato RGB para constituir el texto claro (PT); con el tamaño de la imagen se construye la permutación calculando cada constante C_i de manera aleatoria empleando el número π del punto decimal a la derecha; Se aplica una operación $\oplus$ con la llave k_{i+1}, cumpliendo con las siguientes premisas:

- K es una cadena de 512 bits asociada al entero positivo l. Se realiza el producto $l \times \pi$, del cual se selecciona un bloque de 24 bits, con el argumento de que las imágenes utilizadas en este experimento no exceden una resolución de 2^{24}.

- Al calcular las constantes de la permutación C; se establece que $C_j = a_j \ mod \ h - j$, donde h es el tamaño de la imagen en pixeles.
- Con base en el algoritmo anterior, se asocia π_h
- Del cronograma de llaves K de longitud h, se seleccionan 14 llaves k_i de forma aleatoria $l \times \pi$, partiendo del punto decimal a la derecha; Se toma un bloque de bits del tamaño de la imagen D_0 y se ejecuta una operación xor a la llave k_i.
- Se aplica la S-box calculada hasta la ronda R_{13}.
- El resultado anterior, se emplea una permutación inversa π_h^{-1} a la cadena R.
- Finalmente, a la sustitución se ejecuta la una xor a la k_{14}.

Para ejemplificar este punto, se generó aleatoriamente una S-box de 8×8 en formato hexadecimal, que se muestra en la Tabla 2. Cabe mencionar que en cada proceso de cifrado-descifrado, una S-box diferente se crea y se integra, por ello, las características de estas van a variar y van a presentar diferentes virtudes.

Tabla 2. S-box generada en formato hexadecimal.

HX															
106	15	117	43	99	203	176	65	94	63	217	18	174	82	170	90
24	164	139	128	101	166	75	13	76	152	119	22	77	215	11	111
155	93	135	31	184	84	132	208	123	29	116	1	23	192	206	202
175	46	87	222	252	12	186	52	157	156	19	127	190	7	160	150
27	137	219	173	209	42	246	197	98	180	134	83	48	231	61	141
232	198	207	236	142	162	47	122	64	149	144	9	79	131	96	58
194	54	50	55	253	239	124	244	130	171	102	220	33	229	200	181
8	177	126	109	189	248	70	60	227	107	179	32	237	168	214	249
59	66	213	36	72	210	118	243	71	26	147	3	161	0	228	120
196	153	95	185	103	145	143	53	230	183	20	4	129	73	167	86
138	245	205	38	195	187	28	216	238	25	233	148	226	225	218	49
39	113	35	234	133	211	158	178	193	115	5	78	92	51	44	110
165	91	56	191	69	114	45	125	224	21	254	199	212	104	85	242
108	240	154	37	97	255	159	34	57	68	2	17	146	201	241	88
112	140	151	89	74	204	221	182	41	223	40	121	172	16	247	81
169	62	14	163	67	10	188	250	30	105	136	6	251	100	80	235

Tabla 3. Características de una S-box para un proceso de cifrado-descifrado.

Balance	0
Nonlinearity	94
Absolute indicator	96
Sum of square indicator	259840
Corelation immunity	0
Algebraic degree	7
Transparency order	7.812
Propagation characteristic	0
Number of opposite fixed points	1
Composite algebraic immunity	4
Robustness to differential cryptanalysis	0.961
Delta uniformity	10
SNR (DPA) (F)	9.867
Confusion coefficient variance	0.102003

La Tabla 3, muestra una S-box generada en un proceso de cifrado-descifrado. Si bien, hay trabajaos donde se destaca un alta no linealidad en las S-box como por ejemplo (Zahid et al., 2021), donde se muestra que la fortaleza de algunos algoritmos de cifrado se centra en construcción de una S-box de alta no linealidad

que en promedio otorga un valor 115.75, SecCaos-Image apuesta a que, en cada ejecución una S-box diferente se construya y se integre en el proceso, de esta manera, será añadido un elemento de incertidumbre ante un posible ataque al proceso de cifrado. En ese mismo sentido, generar y construir una S-box de alta no linealidad pude requerir de un tiempo y recurso considerable, tal y como se muestra en la Tabla 10 de la anterior referencia.

3.4. Imágenes utilizadas.

Las imágenes propuestas para esta investigación son de dominio púbico y se visualizan en la Figura 2. Son objetos bien conocidos para el procesamiento de imágenes en formato BMP de 512 × 512 pixeles basado en la literatura adjunta a este trabajo.

Figura 2. Imágenes para examinar.

4. Análisis de resultados

4.1. Imágenes cifradas.

A continuación, se muestran las imágenes cifradas con SecCaos-Image. Como se muestra en la Figura 3, Figura 4, Figura 5 y Figura 6 se produjo el proceso de cifrado, donde a simple vista no se distingue patrón alguno relacionado al contenido de la imagen original.

Figura 3. Imagen de Lena sin cifrar y cifrada

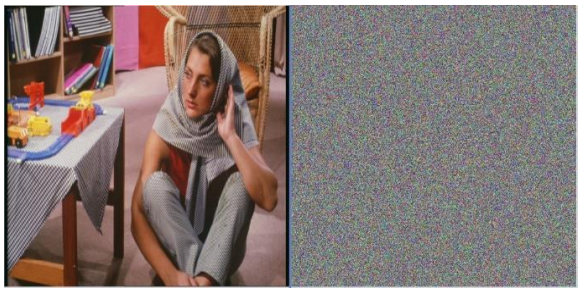

Figura 4. Imagen de Barbara sin cifrar y cifrada

Figura 5. Imagen del Jet sin cifrar y cifrada

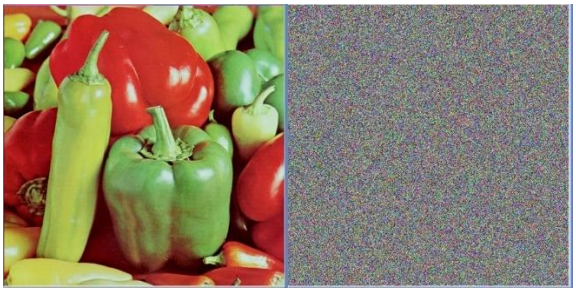

Figura 6. Imagen de Peppers sin cifrar y cifrada

4.2. Análisis de entropía

Como un objeto de estudio de esta investigación, se realizó un análisis al conjunto de llaves K_h aplicando la Entropía de la información. La Tabla 4, muestra el promedio de la entropía a las 14 llaves en cada una de las imágenes. Si bien, la mayoría de las investigaciones focalizan sus estudios en la entropía de los productos cifrados, se considera importante para este desarrollo analizar la entropía de las llaves, dado a que son la fortaleza de un sistema de cifrado simétrico.

Tabla 4. Entropía de las llaves

Imagen	Entropía usando k
Lena	7.99977
Barbara	7.99985
Jet	7.99977
Peppers	7.99976

A continuación, la Tabla 5 muestra la entropía de los objetos cifrados con la finalidad de mostrar una comparación con relación a la literatura presentada. Si bien, los resultados presentados pueden ser muy cercanos, SecCaos-Image contempla una entropía en la mayoría de sus resultados de 7.999, en comparación con el resto que presentan en su mayoría una entropía de 7.99.

Tabla 5. Entropía de las imágenes cifradas con SecCaos-Image

Imagen	SecCaos-Image	(Zahid et al., 2021)	(Ramzan et al., 2021)	(Ahuja et al., 2023)
Lena	7.99933	7.9971	7.9974	7.9997
Barbara	7.99954	7.9967	-	-
Jet	7.99926	7.9973	-	7.9980
Peppers	7.99921	7.9975	7.9969	7.9971

4.3. Análisis de la Correlación

La Tabla 6, se despliegan los resultados de la correlación de las imágenes cifradas de la Figura 3, Figura 4, Figura 5 y Figura 6. En esos casos, se demuestra que la correlación entre los puntos es cercana a cero.

Tabla 6. Correlación de las imágenes cifradas con SecCaos-Image

Imagen	SecCaos-Image	(Zahid et al., 2021)	(Ramzan et al., 2021)	(Ahuja et al., 2023)
Lena	-0.0004	0.0203	-0.0012	-0.0034
Barbara	0.0091	-0.0138	-	-
Jet	0.0082	-0.0058	-	0.0084
Peppers	-0.0062	-0.0045	-0.0013	-0.0039

4.4. Análisis de la prueba de bondad (χ^2)

La Tabla 7 presenta los valores de la chi-cuadrada para las imágenes cifradas, en todos los valores son aceptables y confirman la aleatoriedad. En el caso de (Ahuja et al., 2023) la evaluación presentada fue realizada de diferente forma por lo que sus valores no representan un valor a comparar en este caso.

Tabla 7. Análisis de la chi-cuadrada de las imágenes cifradas con SecCaos

Imagen	SecCaos-Image	(Zahid et al., 2021)	(Ramzan et al., 2021)	(Ahuja et al., 2023)
Lena	246.46	266.16	234.15	-
Barbara	256.76	296.71	-	-
Jet	266.4	246.66	-	-
Peppers	285.4	230.5	236.10	

5. CONCLUSIONES

SecCaos-Image es un algoritmo de cifrado simétrico de imágenes, que basa su fortaleza en su cronograma de llaves las cuales fueron planteadas bajo el conjunto de ecuaciones diferenciales de E. Lorenz y el número trascendente e y puntos aleatorios de la curva elíptica, que incluso, se puede optar por algún otro método para dar respuesta; se contaron con 2^{512} llaves de longitud del objeto. Por otra parte, aplicó un método descrito y utilizado en otras investigaciones para desarrollar una S-box diferente para cada ejecución, así como, una permutación variable. Se logró confirmar una aplicación y un algoritmo robusto que puede resistir ataques de fuerza bruta y diferenciales, ya que su costo computacional es muy alto derivado del conjunto de llaves, la S-box, la permutación variable y el número de rondas. Los resultados de las pruebas planteadas fueron aceptables y en la mayoría de los casos, algunos atributos de este desarrollo son superiores a los comparados. El desarrollo funcionó para imágenes en formato BMP, y por el momento no es compatible para algún tipo de compresión o perdida de datos. Por último, el desarrollo de la aplicación fue realizado en Java 8 con interfaz FX, con un tiempo de procesamiento en las imágenes menor a 0.4s para cada lectura, despliegue y cifra-descifrado de cada figura.

REFERENCIAS

Ahuja, B., Doriya, R., Salunke, S., Hashmi, M. F., Gupta, A., & Bokde, N. D. (2023). HDIEA: high dimensional color image encryption architecture using five-dimensional Gauss-logistic and Lorenz system. *Connection Science*. https://doi.org/10.1080/09540091.2023.2175792

Alanazi, A. S., Munir, N., Khan, M., Hussain, I., & Tools, M. (n.d.). *A novel design of audio signals encryption with substitution permutation network based on the Genesio-Tesi chaotic system*. https://doi.org/10.1007/s11042-023-14964-3

Al-Kadei, F. H. M. (2020). Two-level hiding an encrypted image. *Indonesian Journal of Electrical Engineering and Computer Science*, *18*(2), 961–969. https://doi.org/10.11591/ijeecs.v18.i2.pp961-969

Budiman, M. A., Saputra, M. Y., & Handrizal. (2020). A Hybrid Cryptosystem Using Vigenère Cipher and Rabin-p Algorithm in Securing BMP Files. *Data Science: Journal of Computing and Applied Informatics*, *4*(2), 89–99. https://doi.org/10.32734/JOCAI.V4.I2-4173

Civino, R., Blondeau, · Céline, & Sala, · Massimiliano. (2019). Differential attacks: using alternative operations. *Designs, Codes and Cryptography*, *87*, 225–247. https://doi.org/10.1007/s10623-018-0516-z

Davies, S. R., Macfarlane, R., & Buchanan, W. J. (2022). Comparison of Entropy Calculation Methods for Ransomware Encrypted File Identification. *Entropy 2022, Vol. 24, Page 1503*, *24*(10), 1503. https://doi.org/10.3390/E24101503

Garcia, V. M. S., Ramirez, M. D. G., Carapia, R. F., Vega-Alvarado, E., & Escobar, E. R. (2019). A Novel Method for Image Encryption Based on Chaos and Transcendental Numbers. *IEEE Access*, *7*, 163729–163739. https://doi.org/10.1109/ACCESS.2019.2952030

Gupta, M., Singh, V. P., Kamlesh, &, Gupta, K., Piyush, &, Shukla, K., Kumar, K., & In Piyush, G. K. C. (2023). *An efficient image encryption technique based on two-level security for internet of things*. *82*, 5091–5111. https://doi.org/10.1007/s11042-022-12169-8

Khan, M. F., Ahmed, A., Saleem, K., & Shah, T. (2019). A Novel Design of Cryptographic SP-Network Based on Gold Sequences and Chaotic Logistic Tent System. *IEEE Access*, *7*, 84980–84991. https://doi.org/10.1109/ACCESS.2019.2925081

Krishnan Raghupathy, B., Sekar, K. R., Manikandan, R., Manikandan, G., Bala Krishnan, R., Preethivi, E., Sekar, K., & Prassanna, J. (2019). An Approach with Steganography and Scrambling Mechanism for Hiding Image over Images. *International Journal on Emerging Technologies*, *10*(1), 64–67. https://www.researchgate.net/publication/341642823

Lone, M. A., & Qureshi, S. (2022). Optik-International Journal for Light and Electron Optics RGB image encryption based on symmetric keys using Arnold transform, 3D chaotic map and affine hill cipher. *Optik-International Journal for Light and Electron Optics*, *260*, 168880. https://doi.org/10.1016/j.ijleo.2022.168880

Murillo-Escobar, M. A., Meranza-Castillón, M. O., López-Gutiérrez, R. M., & Cruz-Hernández, C. (2020). A chaotic encryption algorithm for image privacy based on two pseudorandomly enhanced logistic maps. *Studies in Computational Intelligence*, *884*, 111–136. https://doi.org/10.1007/978-3-030-38700-6_5/COVER

Ramzan, M., Shah, T., Hazzazi, M. M., Aljaedi, A., & Alharbi, A. R. (2021). Construction of S-Boxes Using Different Maps over Elliptic Curves for Image Encryption. *IEEE Access*, *9*, 157106–157123. https://doi.org/10.1109/ACCESS.2021.3128177

Rass, S., & Schartner, P. (2020). *Authentic Quantum Nonces*. 35–44. https://doi.org/10.1007/978-3-319-72596-3_3

Ray, S. N., & Chattopadhyay, S. (2021). Analyzing surface air temperature and rainfall in univariate framework, quantifying uncertainty through Shannon entropy and prediction through artificial neural network. *Earth Science Informatics*, *14*(1). https://doi.org/10.1007/s12145-020-00555-5

Vaseghi, B., Mobayen, S., Hashemi, S., & Fekih, A. (n.d.). *Fast Reaching Finite Time synchronization Approach for Chaotic Systems With Application in Medical Image Encryption*. https://doi.org/10.1109/ACCESS.2021.3056037

Vista de Manual de digitalización de documentos. (n.d.). Retrieved March 28, 2023, from https://bagn.archivos.gob.mx/index.php/legajos/article/view/2001/1979

Zahid, A. H., Iliyasu, A. M., Ahmad, M., Shaban, M. M. U., Arshad, M. J., Alhadawi, H. S., & El-Latif, A. A. A. (2021). A Novel Construction of Dynamic S-Box with High Nonlinearity Using Heuristic Evolution. *IEEE Access*, *9*. https://doi.org/10.1109/ACCESS.2021.3077194

Zhang, B., Xiao, D., Huang, H., & Liang, J. (2022). *Compressing Cipher Images by Using Semi-tensor Product Compressed Sensing and Pre-mapping; Compressing Cipher Images by Using Semi-tensor Product Compressed Sensing and Pre-mapping*. https://doi.org/10.1109/DCC52660.2022.00020

Zou, C., Zhang, Q., Wei, X., & Liu, C. (2020). Image Encryption Based on Improved Lorenz System. *IEEE Access*, *8*, 75728–75740. https://doi.org/10.1109/ACCESS.2020.2988880

SEMBLANZA DE LOS AUTORES

Marlon David González-Ramírez: Doctor en Ingeniería de Sistemas por parte del Instituto Politécnico Nacional (IPN). Es profesor titular en el Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional y miembro del Sistema Nacional de Investigadores. Sus líneas de investigación son la seguridad informática, criptografía, comunicación de datos y desarrollo de aplicaciones.

Eduardo Vega-Alvarado: Nació en la Ciudad de México, México, en 1965. Se recibió como Licenciado en Ingeniería en Comunicaciones y Electrónica de la Escuela Superior de Ingeniería Mecánica y Eléctrica, México, en 1992, como M.Sc. en Sistemas Digitales del Centro de Investigación Tecnológica en Computación, México, en 1996, y como Doctor en Sistemas Computacionales y Electrónicos de la Universidad Autónoma de Tlaxcala, México, en 2017. Actualmente es Investigador de tiempo completo y Profesor del Centro de Innovación y Desarrollo Tecnológico en Cómputo, Instituto Politécnico Nacional, Ciudad de México. Sus áreas de interés incluyen la optimización con metaheurísticas, el desarrollo de algoritmos híbridos y su aplicación en ingeniería.

Arturo Morales-Rangel: Nació en la Ciudad de México y es M. Sc. en Tecnología de Cómputo del Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional. Motivado por la industria del cine y los efectos visuales, persiguió su sueño de convertirse en parte del gran juego en Hollywood y contribuir a la magia de la cinematografía con nuevas soluciones. Eso fue lo que lo para desarrollar un software para la automatización de VFX llamado 3D Generic. Su cartera incluye producciones de fama mundial donde sus habilidades y software se utilizaron como parte crucial de la ejecución de proyectos. Hoy es uno de los miembros activos más jóvenes y actualmente reside en el Área de la Bahía de San Francisco.