Congr. Int. en Ing. Electrónica. Mem. ELECTRO, Vol. 40, pp. 56 - 61, Oct 2018, Chihuahua, Chih. México
http://electro.itchihuahua.edu.mx/memorias_electro/MemoriaElectro2018.zip
ISSN 1405-2172

APLICACIÓN WEB PARA ENCRIPTADO DE IMÁGENES BASADA EN CAOS.

Oscar Ricardo Acosta Del Campo[1], Hogilber Quintana Real[1], Cesar Ortega Corral[1], Florencio López Cruz[1].
[1]Universidad Tecnológica de Tijuana (UTT)
Tecnologías de la Información y Comunicación
Km. 10 Carretera Libre Tijuana-Tecate, El Refugio. Quintas Campestre. Tijuana, B.C., C.P. 22253
Tel +52 (664) 969 4700

e-mail de contacto oscar.acosta@uttijuana.edu.mx

RESUMEN.

En este trabajo se presentan los resultados parciales obtenidos con el desarrollo de una aplicación web prototipo para el encriptado de información, particularmente imágenes. Este proceso de encriptado se realiza a partir de la utilización de sistemas caóticos, debido a sus características ya bien documentadas son candidatos potenciales para la aplicación en comunicaciones seguras. En este trabajo se utilizó el sistema caótico de Henon, el cual es un sistema discreto idóneo para su implementación en sistemas digitales, además se utilizó un algoritmo de permutación-difusión de pixeles, con esta combinación se logra un sistema de encriptado-desencriptado completo. Esta aplicación fue desarrollada con el lenguaje de programación Python, los resultados de este trabajo son reportados a continuación.

Palabras Clave: Encriptado, Imagen, Caos, Henon, Sistemas Discretos.

ABSTRACT.

In this work, we report the partial results obtained with the development of a web application prototype for information encryption, particularly images. This encryption process is achieved from the use of chaotic systems, due their characteristics well documented they are potential candidates for the application in secure communications. In this work, we used the Henon chaotic system, which is a suitable discrete system for implementation in digital systems. Also, a pixel permutation-diffusion algorithm was used, with this combination a complete encryption-decryption system is achieved. This application was developed with the Python programming language, the results of this work are reported below.

Keywords: Encryption, Image, Chaos, Henon, Discrete systems.

1. INTRODUCCIÓN

El uso de sistemas de encriptado ha tenido un aumento considerable en los últimos años, esto debido a la falta de seguridad de los diferentes canales de comunicación, que hacen que la información que fluye por ellos pueda ser vulnerada por usuarios no autorizados, haciendo mal uso de la información, tal es el caso de, por ejemplo, espionaje y extorsión. Varios investigadores han realizado diversos estudios respecto al tópico de seguridad en los medios de comunicación, ver por ejemplo las referencias [1-2].

Con canales de comunicación muy demandados como el Internet, la cantidad de información que circula es muy grande, miles de Gigabytes de datos confidenciales son transmitidos a diario, estos datos pueden ser: información personal, familiar, fotografías, archivos importantes, etc. Al no tener un sistema eficiente que proteja estos datos, pueden ser fácilmente extraídos por piratas informáticos.

Los sistemas de encriptado actuales pueden ser simétricos (una clave secreta) o asimétricas (dos claves secretas); los métodos simétricos como 3DES (Triple Data Encryption Standard), AES (Advanced Encryption Standard), e IDEA (International Data Encryption Algorithm) tienen ventajas de velocidad mientras que los métodos asimétricos como RSA (Rivest, Shamir y Adleman) tienen ventajas de seguridad [3]. Sin embargo, todos ellos ya son muy conocidos y al ser expuestos durante tanto tiempo su efectividad ha disminuido, debido a que han sido analizados a profundidad por los piratas informáticos, logrando vulnerarlos y conseguir descifrar la información encriptada.

Motivados por la disminución de la seguridad de estos sistemas, diversos investigadores están realizando trabajos de investigación para generar nuevos métodos de encriptado que sean eficientes y seguros, actualmente las tendencias de investigación son criptografía caótica, criptografía basada en ADN, criptografía cuántica y criptografía de curvas elípticas, algunos ejemplos de estos trabajos se pueden ver en las referencias [4 - 7].

Una de las tendencias alternativas a los métodos de encriptado convencional es el encriptado caótico, el cual ha tenido bastante auge en tiempos recientes, las primeras contribuciones en esta área han sido en la sincronización de sistemas caóticos [8-14], debido a sus características y su potencial aplicación al encriptado de información, se han hecho numerosas investigaciones al respecto [15-16].

En este trabajo se aborda particularmente el encriptado de imágenes, debido a la gran cantidad de aplicaciones donde se requiere tal encriptado, como telemedicina, televisión por cable, imágenes militares, imagen personal, videoconferencia, sistemas biométricos, etc., en ellas se requiere un encriptado práctico, rápido y seguro. Las técnicas para encriptar imágenes más utilizadas pueden ser permutación de píxel, difusión de pixel, o permutación-difusión de pixel [17-21].

Los algoritmos de encriptado convencional como 3DES, AES, IDEA, etc., son excelentes algoritmos para el encriptado de texto y otros tipos de archivos, sin embargo, no son adecuados para el encriptado de imágenes a color para aplicaciones en tiempo real debido a la redundancia de información, alta correlación en píxeles adyacentes y al gran tamaño de los datos [22-23]. Las técnicas de encriptado de imágenes basadas en metodologías de compresión, usadas en imágenes en escala de grises son ineficientes y lentas [24-28], debido a que actualmente la gran mayoría de imágenes son a color.

En este trabajo se realiza el encriptado de imágenes a color, para ello se utiliza un algoritmo de encriptado presentado recientemente [29], el cual toma todas las características de la imagen plana, además puede ser utilizado en aplicaciones en tiempo real, donde se requiere un alto nivel de seguridad. Para garantizar un algoritmo de encriptado seguro, se hace uso del sistema caótico de Henon [30]. Este sistema caótico ha pasado varias pruebas realizadas con el fin de conocer su aleatoriedad y su potencial aplicación a comunicaciones seguras. Una de ellas es un conjunto de prueba basada en estadísticas conocido como NIST prueba desarrollada por el Instituto Nacional de Estándares y Tecnología [31]

Aquí se muestran los resultados parciales obtenidos en realización de una aplicación web prototipo para el encriptado de imágenes a color utilizando sistemas caóticos discretos, dichos resultados fueron generados utilizando el lenguaje de programación Python, se muestran los resultados de la interfaz obtenida, así como los resultados del proceso de encriptado-desencriptado.

El resto del documento está organizado de la siguiente manera: En la sección 2 se dan las preliminares, es decir, la teoría y proceso necesario para realizar el encriptado, los detalles del algoritmo utilizado, así como los del sistema caótico utilizado. En la sección 3 se muestran los resultados obtenidos, tanto de la interfaz generada como del proceso de encriptado y desencriptado. En la sección 4 se dan las conclusiones finales de este trabajo.

2. PRELIMINARES

2.1. Sistema caótico de Henon

El mapa de Hénon fue presentado por el astrónomo Francés Michel Hénon (1976). Dicho sistema esta descrito por las siguientes ecuaciones no lineales en diferencias [30]:

$$x(n+1) = \propto +\beta y(n) - x^2(n)$$
$$y(n+1) = x(n) \qquad (1)$$

Con el propósito de generar la dinámica caótica y un atractor caótico para el mapa de Hénon, se utilizan los siguientes valores para los parámetros en simulaciones numéricas generadas en Python, $\beta = 0.3$, $\alpha = 1.4$ y condiciones iniciales $x(0) = 0.11$ y $y(0) = 0$. El atractor caótico generado por el mapa de Hénon con los parámetros anteriores se ilustra en la Figura 1.

2.2. Proceso de encriptado de imagen.

Para comenzar, se elige una imagen a color $\boldsymbol{P}$ de $\boldsymbol{M\ x\ N\ x\ 3}$ píxeles, donde $\boldsymbol{M}$ son las filas, $\boldsymbol{N}$ las columnas y 3 son el componente RGB; cada componente RGB (rojo, verde, azul) tiene una dimensión $\boldsymbol{M\ x\ N}$ con $\boldsymbol{0\ -\ 255}$ valores. El algoritmo de encriptado utilizado se describe a detalle en [29].

La Figura 2 muestra el diagrama a bloques del procedimiento, donde se aprecia que la imagen pasa por una etapa de permutación, es decir, todas las posiciones de los pixeles son cambiadas de acuerdo a las dinámicas caóticas generadas por el sistema discreto, así como por una etapa de difusión, donde a cada pixel se le cambia su valor de acuerdo nuevamente a las dinámicas caóticas generadas por el sistema discreto, así mismo todos los valores de la imagen se suman con los datos caóticos generados por el sistema de Hénon, de esta forma se obtiene $\boldsymbol{Z}$, la cual será necesaria para recuperar la información.

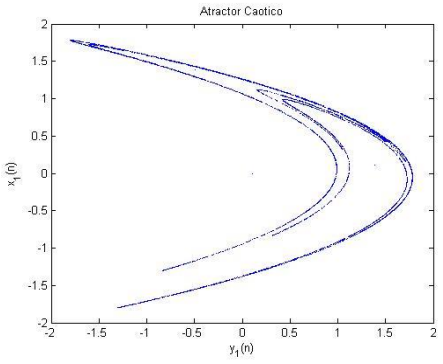

Figura 1. Atractor caótico del mapa de Hénon.

Se puede observar que es necesaria una clave secreta que consta de 32 caracteres o 128 bits, servirá como condición inicial para los sistemas caóticos Henon.

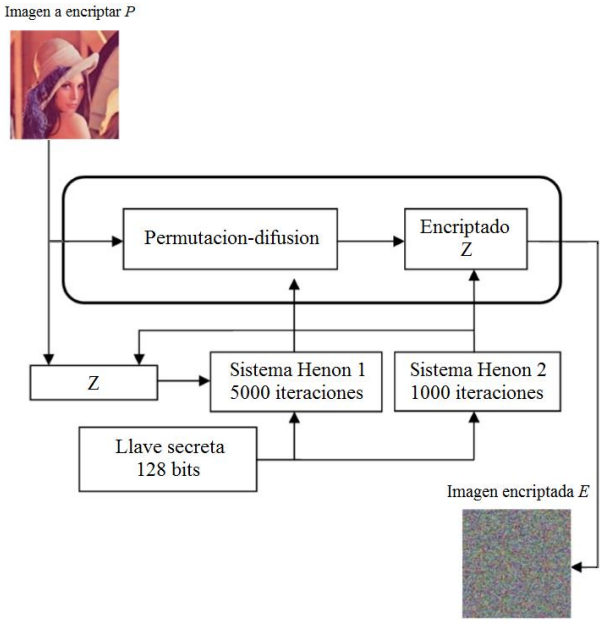

Figura 2. Diagrama a bloques del proceso de encriptado.

2.3 Proceso de desencriptado de imagen.

El proceso de desencriptado de imagen es básicamente el proceso inverso al del encriptado, en la Figura 3 se puede observar el diagrama a bloques de dicho proceso. Para este proceso es requerida la clave secreta elegida para el encriptado, si por alguna razón dicha clave secreta es distinta, no se podrá recuperar la información, puesto que las condiciones iniciales serán distintas y se generarán otras dinámicas caóticas distintas a las del proceso de encriptado. En este caso se realiza un proceso de permutación-difusión inverso (al utilizado para encriptar), de esta forma se obtendrá la misma posición de los pixeles, así como el mismo valor de pixel de la imagen original, teniendo de esta forma la imagen recuperada.

3. RESULTADOS.

Los resultados del proceso de encriptado-desencriptado fueron obtenidos usando el lenguaje de programación Python, así como la interfaz web desarrollada. Esta aplicación consta de 4 secciones, la sección de inicio de sesión, donde los usuarios no registrados se registran para poder acceder al portal principal o bien para iniciar sesión con su usuario y contraseña. La 2da sección es la de encriptado, donde el usuario accederá al proceso de encriptado, abriendo la locación de la imagen y seleccionándola para encriptar, además en esta sección el usuario necesita introducir una clave secreta que le servirá para desencriptar la imagen cuando lo requiera. La sección 3 es donde se realiza el proceso de desencriptado, aquí el usuario abre la locación de la imagen encriptada y es necesario ingresar la llave secreta correcta para que se realice el desencriptado, si no es la llave correcta la imagen no se podrá desencriptar. La cuarta sección corresponde a la biblioteca, que es donde se pueden observar todas las imágenes encriptadas, separadas por días, meses y años

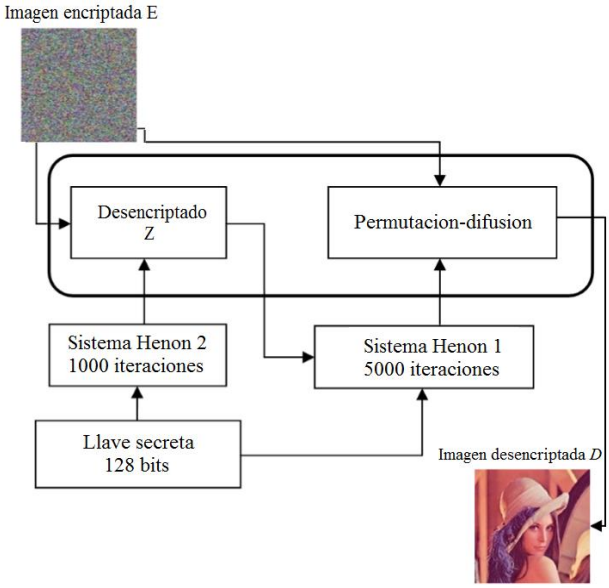

Figura 3. Diagrama a bloques del proceso de desencriptado.

3.1 Inicio de sesión.

Esta es la pantalla principal de la aplicación, en esta sección los usuarios iniciarán sesión, ingresando su usuario y contraseña, si no se encuentran registrados, este proceso será necesario para continuar. La pantalla principal se aprecia en la Figura 4.

Figura 4. Ventana principal para inicio de sesión.

3.2 Encriptado de imagen.

Una vez iniciada la sesión, se abrirá la siguiente sección, que es la de encriptado, aquí es donde se selecciona la imagen a encriptar, para poder realizar el encriptado es necesario agregar una lleva secreta, que debe constar de 32 caracteres, pudiendo ser números del 0 al 9 o letras de la A a la F. Esta ventana se aprecia en la Figura 5.

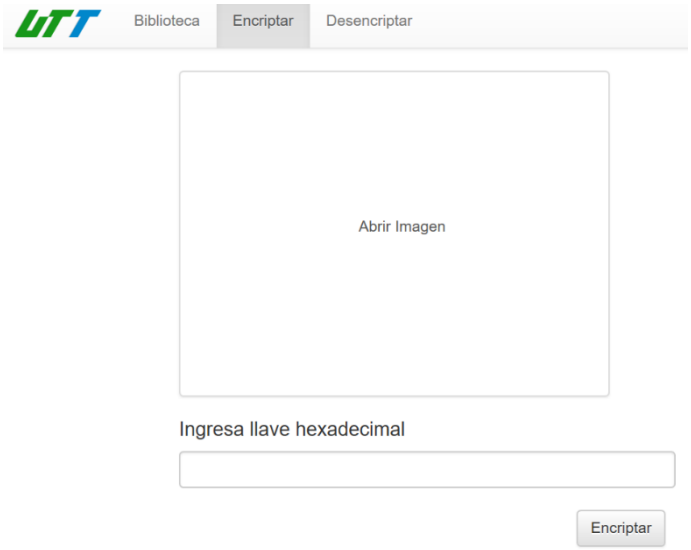

Figura 5. Sección para el encriptado de imagen.

Debido a la gran cantidad de formatos de imágenes digitales, el algoritmo de encriptado debe ser universal, es decir, debe de encriptar cualquier imagen con resultados en seguridad y funcionamiento similares. En este trabajo se utilizaron de forma didáctica 2 imágenes para comprobar la efectividad de la aplicación. Para el proceso de encriptado se utilizó como primera imagen plana *P*, la icónica imagen de "Lena", la cual se encontraba en formato PNG con una dimensión de 512×512. La imagen de Lena se aprecia en la Figura 6. Como segunda imagen plana se tomó la de un ave en formato JPG con una dimensión de 1024×768, la cual se aprecia en la Figura 7. El encriptado fue realizado utilizando como llave secreta "*1A2B3C4D5F6A7B8C9D0F1F2A3B4C5D6E*", la cual al ser hexadecimal consta de 128 bits, esta clave secreta se utilizó para ambas imágenes.

La ventana con las imágenes seleccionadas y encriptadas se muestra en las Figuras 8 y 9. La Figura 8 para la imagen de Lena y la 9 para la del halcón. Como resultado del proceso, se obtuvieron las imágenes encriptadas lo que confirma la efectividad de la aplicación desarrollada. Se puede observar que el resultado al encriptar una imagen es otra imagen indistinguible, es decir, la información original no se puede reconocer de la imagen encriptada.

Figura 6. Imagen de "Lena" utilizada para el encriptado.

Figura 7. Imagen de un halcón utilizada para el encriptado.

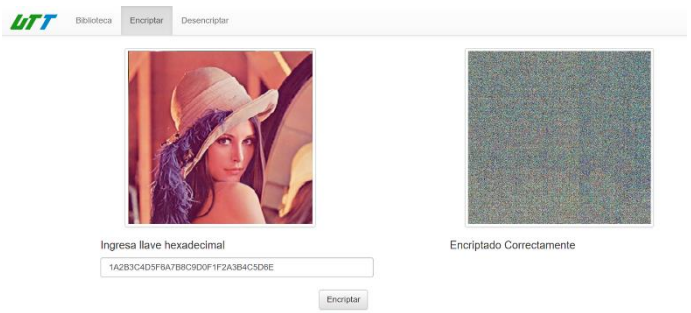

Figura 8. Encriptado de imagen de Lena.

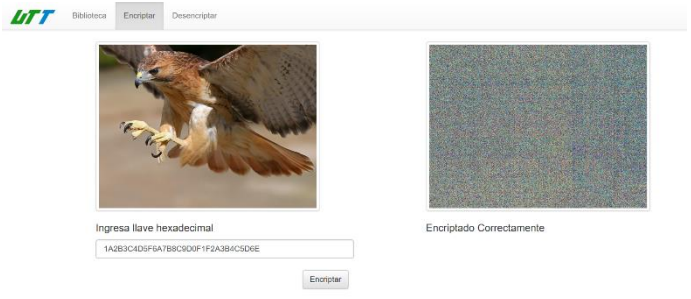

Figura 9. Encriptado de imagen de halcón.

3.3 Desencriptado de imagen.

Al ingresar a la sección de desencriptar se podrá seleccionar la imagen encriptada que se desea desencriptar, esta pantalla se aprecia en la Figura 10.

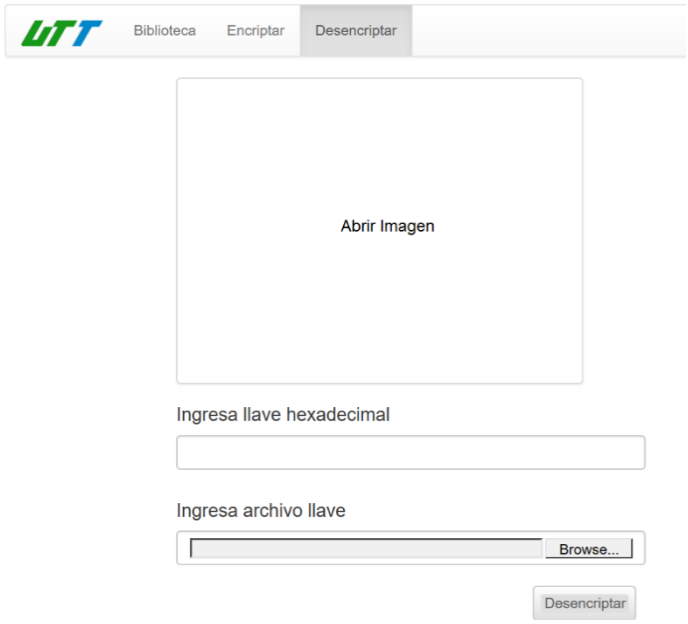

Figura 10. Sección para desencriptar imagen.

Para realizar el correcto desencriptado de la imagen es necesario utilizar la misma clave secreta con la que se encriptó la imagen, dicha clave es "*1A2B3C4D5F6A7B8C9D0F1F2A3B4C5D6E*", las ventanas con las imágenes seleccionadas y desencriptadas se aprecian en las Figuras 11 y 12 respectivamente. Se puede observar en ambas figuras que las imágenes se recuperaron en su totalidad, es decir, se desencriptaron correctamente.

Con esto se comprueba la efectividad de la aplicación desarrollada siendo completada con el uso de sistemas caóticos discretos, en este caso el sistema caótico de Hénon, en investigaciones posteriores se realizará un análisis contra los diferentes ataques que pudiera sufrir tal información.

3.4 Biblioteca.

En esta sección se encuentran todas las imágenes que han sido encriptadas. Podemos buscar o seleccionar una imagen por año, mes y día. Esta sección se muestra en la Figura 13.

3.5 Análisis de efectividad

Una forma de corroborar la efectividad del encriptado, es por medio del histograma de las imágenes, ya que nos muestra los niveles de intensidad de los colores rojo, verde y azul de las imágenes. En la figura 14a, por ejemplo, se muestra el histograma de la imagen de Lena utilizada para realizar el encriptado. En cambio, en la figura 14b se aprecia el histograma de esta misma imagen, pero encriptada, podemos observar, que para la imagen encriptada se muestra una grafica plana de los niveles de estos colores y al comparar estos 2 histogramas, podemos observar que son totalmente distintos.

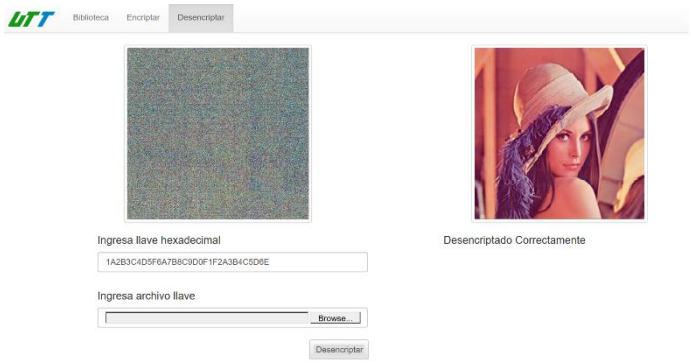

Figura 11. Imagen de "Lena" recuperada.

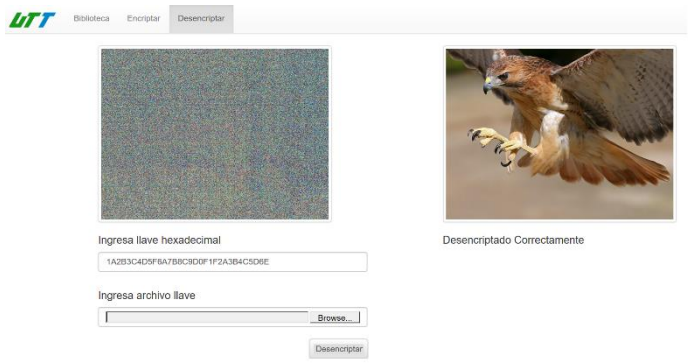

Figura 12. Imagen de "halcón" recuperada.

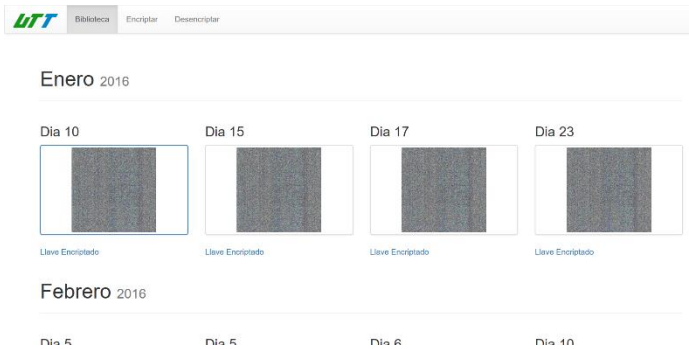

Figura 13. Sección de biblioteca de la aplicación.

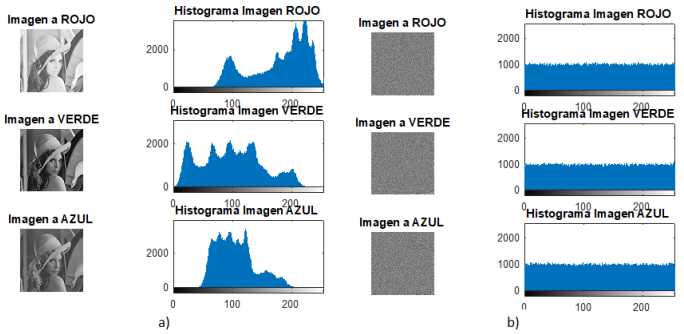

Figura 14. Histograma de la imagen de Lena, sin encriptar (a) y encriptada (b).

4. CONCLUSIONES

En esta aplicación web prototipo se realizó el encriptado de información, específicamente imágenes, para tal efecto se utilizó un algoritmo de encriptado propuesto recientemente donde se realiza un proceso de permutación-difusión a nivel de pixel, para este trabajo se propuso utilizar un sistema caótico discreto que complementa al algoritmo seleccionado, se utilizó el sistema discreto de Hénon, que nos ofrece muy buenas características para las comunicaciones seguras. Los resultados del desarrollo muestran una interfaz donde se puede iniciar sesión, para continuar con el proceso una sección de encriptado, que es donde se seleccionará la imagen a encriptar y se establecerá una clave secreta, también se muestra la sección de desencriptado, que es donde se recuperara la imagen original, esto ingresando la clave secreta correcta, además de una sección de biblioteca, que contiene a todas las imágenes encriptadas. Con los resultados visuales se puede constatar la efectividad de la aplicación ya que en las imágenes encriptadas no se aprecia la imagen original.

REFERENCIAS

[1] Rongyu He, Chaowen Chang, Guolei Zhao, Zheng Qin y Xi Qin, (2008), Police Security Communication over Public Cellular Network Infrastructure, NCA '08 Seventh IEEE International Symposium on Network Computing and Applications, pp 232-235.

[2] C.K. Huang, Y.H. Hsu, W.Y. Chen, S.K. Changchien, C.M. Hung, C.H. Liu, y Y.R. Tian, (2009), High Security Image Encryption by Two-stage Process, ICICS 2009 7th International Conference on Information, Communications and Signal Processing, pp 1-5.

[3] A. Uhl, A. Pommer, Image and video encryption: from digital rights management to secured personal communication, Advances in Information Security 15 (2004) 161.

[4] Qiang Zhang, Ling Guo y Xiaopeng Wei. (2010) Image Encryption Using DNA Addition Combining with Chaotic Maps. Mathematical and Computer Modelling, 52(11-12), pp. 2028–2035.

[5] Cruz-Hernández C., López-Gutiérrez R.M., Aguilar-Bustos A.Y. y Posadas-Castillo C. (2010). Communicating Encrypted Information Based on Synchronized Hyperchaotic Maps. International Journal of Nonlinear Sciences and Numerical Simulation, 11(5), pp. 337–349.

[6] Sharbaf, M.S. (2011), Quantum cryptography: An emerging technology in network security, 2011 IEEE International Conference on Technologies for Homeland Security (HST). pp 13-19.

[7] Kodali, R.K, (2014), Implementation of ECC with hidden Generator point in Wireless Sensor Networks, 2014 Sixth International Conference on Communication Systems and Networks (COMSNETS), pp 1-4.

[8] L.M. Pecora and T.L. Carroll, Phys. Rev. Lett. 64 (1990) 821.

[9] C. Cruz-Hernandez and A.A. Martynyuk, Advances in chaotic dynamics with applications, Vol. 4 (Cambridge Scientific Publishers Ltd., 2010).

[10] C. Cruz-Hernandez and H. Nijmeijer, Int. J. Bifurc. Chaos 10 (2000) 763.

[11] H. Sira-Ramırez and C. Cruz-Hernandez, Int. J. Bifurc. Chaos 11 (2001) 1381.

[12] D. Lopez-Mancilla and C. Cruz-Hernandez, ´ Nonlinear Dyn. Syst. Theory 5 (2005) 141.

[13] U. Feldmann, M. Hasler and W. Schwarz, Int. J. Circ. Theory Appl. 24 (1996) 551.

[14] H. Nijmeijer and I.M.Y. Mareels, IEEE Trans. Circ. Syst. I 44 (1997) 882.

[15] A.N. Pisarchik and M. Zanin, Physica D 237 (2008) 2638.

[16] A. Kanso, M. Ghebleh, Commun Nonlinear Sci Numer Simulat 17 (2012) 2943.

[17] J. C. Yen, J. I. Guo, An efficient hierarchical chaotic image encryption algorithm and its vlsi realization, IEE Proceedings Vision, Image and Signal Processing 47 (2) (2000) 167–175.

[18] H.-C. Chen, J.-C. Yen, A new cryptography system and its vlsi realization,
Journal of Systems Architecture 49 (7-9) (2003) 355–367.

[19] G. Chen, Y. Mao, C. K. Chui, A symmetric image encryption based on 3d chaotic maps, Chaos Solitons & Fractals 21 (3) (2004) 749–761.

[20] P. P. Dang, P. M. Chau, Image encryption for secure internet multimedia applications, IEEE Transactions on Consumer Electronics 46 (3) (2000) 396–403.

[21] M. Zeghid, M. Machhout, L. Khriji, A. Baganne, R. Tourki. A modified AES based algorithm for image encryption, International Journal of Computer Science & Engineering 1 (1) (2007) 70.

[22] F. B. Muhaya, M. Usama, M. K. Khan, Modified aes using chaotic key generator for satellite imagery encryption, Emerging Intelligent Computing Technology and Applications 5754 (2009) 1014–1024.

[23] A. J. Menezes, P. C. van Oorschot, S. A. Vanstone, Handbook of applied cryptography, CRC Press.

[24] H. K. L. Chang, J. L. Liu, A linear quadtree compression scheme for image encryption, Signal Processing: Image Communication 10 (4) (1997) 279– 290.

[25] G. Nagaraju, T. V. H. Lakshimi, Image encryption using secret-key images and scan patterns, Int. J. of Advances in Computer, Electrical & Electronics Engineerig 2.

[26] N. G. Bourbakis, Image data compression-encryption using g-scan patterns, IEEE International Conference on Computational Cybernetics and Simulation 2 (1997) 1117–1120.

[27] C. C. Chang, M. S. Hwang, T. S. Chen, A new encryption algorithm for image cryptosystems, The Journal of Systems and Softwares 58 (2) (2001). 1690–1701.

[28] T.-H. Chen, S.-C. Wu, Compression-unimpaired batch-image encryption combining vector quantization and index compression, Information Sciences 180 (9) (2010) 1690–1701.

[29] M.A. Murillo-Escobar, C. Cruz-Hernández, F. Abundiz-Pérez, R.M. López-Gutiérrez, O.R. Acosta Del Campo, A RGB image encryption algorithm based on total plain image characteristics and chaos, Signal Processing, Volume 109, April 2015, Pages 119-131.

[30] Dmitriev A. S., G. A. Kassian, Khilinsky A. D. “Chaotic synchronization of Henon mappings: the information approach”, Technical Physics Letters 28 (2002) 5.

[31] A. Rukhin, et al, “A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications”, NIST (2001), http://csrc.nist.gov/rng/.