JIRBDAI

Journal of Institutional Research, Big Data Analytics and Innovation

AN IMAGE AUTHENTICATION SCHEME USING MSB-LSB STEGANOGRAPHY AND HASH BASED MESSAGE AUTHENTICATION CODE

EZECHI, N. E.*, ADENIJI, A. A., MOGBONJU M. M.

Department of Mathematics, University of Abuja, Nigeria

*Corresponding Author: Jide4nkiru@gmail.com

ABSTRACT

Image encryption prevents unauthorized access to images it does not provide verification that the encrypted image has not been altered or modified. This paper presents a mathematical authentication framework for encrypted images based on Most Significant Bit (MSB) extraction, HMAC based tag generation and the Least Significant Bit (LSB) embedding. The authentication process is performed after image encryption. For each encrypted pixel, the leftmost bit is extracted to construct an MSB-derived authentication representation. A shared secret is processed through the Hash based Message Authentication Code (HMAC) based Key Derivation Function (HKDF) to generate authentication-specific key material, which is subsequently used to generate an authentication tag using HMAC. The generated tag is embedded in designated LSB positions of the encrypted image. At the receiving end, the embedded authentication tag is extracted and the MSB-derived representation is reconstructed. A corresponding authentication tag is recomputed using the same derived authentication key, and the extracted and recomputed tags are compared to determine authentication validity. Finally, theoretical properties of the construction are established addressing MSB invariance under LSB embedding, authentication correctness, sensitivity to modifications affecting the authenticated MSB representation, authentication-tag integrity, domain separation, and random tag-forgery probability.

KEYWORDS: Image Authentication, HKDF, Hash based Message Authentication Code, Authentication Tag, Most Significant Bit, Least Significant Bit Embedding

I. INTRODUCTION

Image authentication is the process of validating the integrity of an image to ensure that it has not been altered. The authentication process in this work is to detect whether the encrypted image has been altered. The process of image authentication is important in this era of AI-generated content, digital manipulation and deepfakes, where visual misinformation can have adverse effects on many areas of life, such as journalism, legal proceedings, medicine and social media contexts. Authentication is a crucial security requirement in image communication systems because encryption alone does not guarantee that the received image has not been altered during transmission or storage. The encryption process is mainly concerned with protecting the confidentiality of image data, while the authentication process focuses on verifying the integrity and authenticity of the image. A good authentication mechanism should enable the receiver of the image to determine whether the received image has been modified or tampered with. Image encryption can include authentication by producing an authentication value from the original

Volume 2, Issue 3, 2026

https://doi.org/10.5281/zenodo.22165095

image or from the encrypted image and embedding this value within selected image pixels. The receiver then extracts the embedded authentication information and compares it to a new authentication value. If the two values are the same, the image is considered authentic. Otherwise, a mismatch indicates that the image may have been modified.

Steganography is the art of hiding information within other information. It offers a powerful approach to embedding authentication data directly into images. Among steganographic techniques, LSB (Least Significant Bit) embedding is widely used due to its imperceptibility. More recently, MSB (Most Significant Bit) embedding has gained attention for its robustness against compression and noise. This paper presents an image authentication technique that employs the left most bit extraction of an encrypted image and combines it with an authentication tag generated from HMAC and finally embeds this bit with the LSB of the image. The mathematical properties of this process are also discussed.

II. RELATED WORK

Yang *et al.* (2009) pioneered the adaptive LSB method of steganography, which uses the cover object's edges, brightness and texture to compute the number of k LSBs for hiding information. They found that the edge area can tolerate fewer changes compared to a highly textured portion of an image, but not more than a smooth area. They finally computed the LSB using the high-order bits of a pixel to achieve a more secure system. The LSB provides a suitable location for embedding authentication information because, when bits are changed in the LSB, it produces little change in the pixel intensity, while the MSB representation is derived from the more significant portion of the pixel value. The authentication process is therefore incorporated as a complementary layer of the encryption scheme.

Solomon & Adebanke, (2015) developed a hybrid algorithm that involves LSB and MSB to embed images and text into images. The algorithm suggested that secret messages are embedded into the Least Significant Bit and the Most Significant Bit of the cover image. Their result showed that the stego-images of LSB have the highest PSNR and the lowest MSE values, making it very efficient to hide the data inside the image. They also found out that, based on the encoding time, the combined LSB-MSB algorithm takes lesser time in embedding than LSB and MSB. Finally, they found out that the LSB algorithm gives a better performance than the combined LSB-MSB algorithm, but a larger file size of the cover images makes the combined algorithm produce better image quality than the MSB algorithm.

Gaurav (2015) proposed an image steganography technique in which data is embedded into an image. It involves bits of information being placed in the pixel values of an image so that an attacker cannot find out where the data is hidden. He was able to achieve this using the LSB and MSB pixels for hiding and retrieval of the message.

Kiswara *et al.* (2021) proposed an image authentication scheme by using a magic square of order 3 to authenticate an image. The magic square was used as an image protection tool by manipulating the image to prevent detection. Before the manipulation using the magic square, the image was first inserted with the LSB. In their algorithm, if the pixels in the image did not satisfy the magic square rule, the change would be detected and the color would turn white.

Zaid *et al.* (2023) proposed an image watermarking technique based on a Canny edge detection algorithm and an LSB approach to embed watermarks. They proposed that Canny edge detection is a simple yet efficient approach to detect edges in an image, in order to find places more suitable to embed the watermarks. LSB was also used to embed watermarks because it works at the pixel

Volume 2, Issue 3, 2026
https://doi.org/10.5281/zenodo.22165095

level in the time domain and has a higher capacity to add watermarks. These approaches were performed on different image processing and geometric attacks. They found that the parameters and MSE normalized correlation gave a value closer to one, indicating that the watermark is highly robust and did not break.

Rohit *et al.* (2024) introduced a new algorithm based on Sparse Approximation (SA), Quantum Encryption (QE) and a measurement matrix to achieve watermark encryption as well as its secure transmission by embedding it in a host image. They first divided the image into four sub-images before the sparse approximation was performed in the transform domain. They enhanced the security by transforming the sparse data in the Discrete Wavelet Transformation (DWT) domain and then applied a sparse coefficient exchange. Finally, they generated an encrypted watermark through a series of inverse sparsification, DWT, and sampling. The watermark information was what they used for authentication.

Muhammad *et al.* (2024) proposed a steganography technique that utilizes a combination of Most Significant Bit (MSB) matching and Least Significant Bit (LSB) substitution. They first divided confidential messages into pairs of bits and then connected them with the MSB of individual pixels using pair matching. This enabled the storage of 6 bits in one pixel by modifying a maximum of three bits. They compared their work with the Zakariya scheme, and this achieved an improvement in the hiding capacity from 11% to 22%, while it maintained a minimum Peak Signal to Noise Ratio (PSNR) of 37dB.

Muhammad *et al.* (2024) introduced a novel RGB-to-RGB image steganography algorithm that embeds the Josephus permutation into the LSB 3-3-2 technique. Their technique employed the chaotic logistic map function within the range $3.5688 < r < 4$ to generate the keystream. To ensure that the parameters induced chaotic behavior, the keystream generation was assessed using the Lyapunov exponent and bifurcation diagram. The keystream they generated was then scrambled using the Josephus permutation, hence providing randomness and security. They were also able to show that their technique is safe against brute-force attacks.

The existing image authentication approaches uses a variety of techniques in authenticating images like watermarking, LSB based embedding, and combined MSB-LSB hiding methods. This paper proposes an image authentication scheme that employs MSB extraction, HMAC based tag generation and LSB embedding. We also study some mathematical properties of this scheme.

III. METHODOLOGY AND DISCUSSION

In this work, an authentication mechanism incorporates the use of Least Significant Bit (LSB), Most Significant Bit (MSB) and Hash based Message Authentication Code (HMAC) to authenticate the encrypted image.

A. Least Significant Bit (LSB) and Most Significant Bit (MSB)

The LSB is the binary digit (bit) at the far-right side of a binary number. In digital computing, data is stored as binary 0's and 1's. The LSB is the binary digit (bit) at the far-right side of a binary number. Changing this bit causes the smallest change to the number's magnitude. It generally determines whether a number is odd (1) or even (0). The MSB is the binary digit (bit) at the far-left side of a binary number. It is the most significant bit because changing it causes the largest change to the number's magnitude.

Volume 2, Issue 3, 2026
https://doi.org/10.5281/zenodo.22165095

Authentication Tag

An authentication tag is the data generated from the encrypted image and a secret authentication key. Its purpose is to allow the receiver check if the image has been altered. In this scheme, a new authentication key is used to prevent the authentication mechanism from being confused with the encryption key. The authentication key is obtained from the shared secret key ECDH using HKDF. In this paper, ECDH is used to get shared secret key that is used as input to HKDF, but it will have different information telling it that this key is for authentication.
Hence, the authentication key is:

$K_A = HKDF(S, info_A, L)$

where S is the shared ECDH secret key, $info_A$ identifies the purpose of the derived key as authentication and L is the required key length. Hence, the authentication tag T_E is:

$$T_E = HMAC(K_A, C)$$

where C represents the encrypted image data.

The Authentication Algorithm

After the encryption process is carried out, the MSB of each pixel block is extracted, and this is used as the raw message C in the key. Next, we produce the authentication tag:

$T_E = HMAC(K_A, C)$

Lastly, we overwrite the LSBs of the image with T_E.

Verification: The Reverse Pipeline

To verify if the image is authentic, the receiver first extracts T_E (extracted) from the LSBs. Next, the receiver zeroes out the LSBs, extracts $MSB'C'$, and recomputes T_R (recomputed) (redoing step 2 over the MSB). Lastly, we compare T_R (recomputed) against T_E (extracted). If they match, the MSB data is authentic.

Table 1: Shows the authentication algorithm

Step	Sender (Embed)	Receiver (Verify)
1	Set all LSBs to 0	Extract LSBs → this is T_E (extracted)
2	Extract MSB data C	Set all LSBs to 0
3	Derive $K_A = HKDF(C, salt, ImageID, info)$	Derive K_A (the same way)
4	Compute $T_E = HMAC(K_A, C)$	Extract MSB data C'
5	Overwrite LSBs with T_E	Compute $T_R = HMAC(K_A, C')$ Where $T_R = .T_E$
6	Save as lossless format (PNG/BMP)	Pass if and only if $T_E = T_R$ Verify $(C^{\mid}) = \begin{cases} 1, & T_E = T_R \\ 0, & T_E \neq T_R \end{cases}$

Theoretical Properties

In this section, we investigate some of the theoretical properties of the entire authentication process.
Let the encrypted image be $C = (C_{ij}) \in \{0, 1, \dots, 255\}^{m\,x\,n}$
Each pixel can be represented as:

$$C_{ij} = (b_{7ij}\, b_{6ij}\, b_{5ij}\, b_{4ij}\, b_{3ij}\, b_{2ij}\, b_{1ij}\, b_{0ij})_2$$

The MSB extraction map $\mu : \{0, \dots, 255\} \rightarrow \{0, 1\}$ is defined by $\mu(C_{ij}) = b_{7ij} = \frac{C_{ij}}{128}$

Volume 2, Issue 3, 2026
https://doi.org/10.5281/zenodo.22165095

For the entire image, we define the MSB extraction as $M = \mu(C) = (\mu(C_{ij}))ij$
The authentication key is $KA = HKDF(Z, S, \gamma)$
where Z is the shared secret, S is the salt, and γ is the authentication information. So that the authentication tag $T = HMAC_{KA}(M)$.
Let $Emb(C,T)$ denote embedding T into selected LSB positions of C, and let $Ext(C^*)$ denote extraction of the embedded tag.
Let the authenticated encrypted image is $C^* = Emb(C,T)$
With extraction key $T = HMAC_{KA}(\mu(C))$:

Lemma 1

Let $\mu(Emb(c,t)) = \mu(C)$ for every embedded bit $t \in \{0,1\}$. Consequently, $\mu(Emb(C,T)) = \mu(C)$
Proof.
Let an 8-bit pixel be $C = (b_7 b_6\, b_5\, b_4\, b_3\, b_2\, b_1\, b_0)_2$. The LSB embedding replaces only b_0 with a bit t. Hence,

$$Emb(C,t) = (b_7 b_6\, b_5\, b_4\, b_3\, b_2\, b_1\, t)_2$$

But the MSB remains b_7. Therefore, $\mu(Emb(c,t)) = b7 = \mu(C)$
Applying this to every pixel of C gives
$\mu(Emb(C,T)) = \mu(C)$. ■
This shows that MSB authentication data are invariant under LSB embedding.

Proposition 1

Let $T = HMAC_{K_A}(\mu(C))\ and\ C^* = Emb(C,T)$.
If $Ext(C^*) = T$ and the receiver derives the same K_A, then
Verify $(C^*, K_A) = 1$ (authentication is successful)
Proof:
At the sender, $T = HMAC_{K_A}(\mu(C))$
After embedding, $C^* = Emb(C,T)$
From Lemma 1 we have

$$\mu(C^*) = \mu(C)$$

So, the receiver extracts $T_E = Ext(C^*) = T$
and recomputes $T_R = HMAC_{K_A}(\mu(C^*))$
Since $\mu(C^* = \mu(C)$, we have

$$T_R = HMAC_{K_A}(\mu(C)) = T$$

Thus $T_E = T_R$. Hence, authentication is successful:
So, verify $(C^*, K_A) = 1$■

Proposition 2

Let $T = HMAC_{K_A}(\mu(C))$. Suppose the embedded tag is replaced by Q ≠ T. Then,
$Verify\ (Emb(C,\tilde{T}), KA) = 0$ (That is, authentication is unsuccessful)
Proof.
Let $\tilde{C} = Emb(C,\ \tilde{T})$
The receiver extracts $T_E = \tilde{T}$
Since only the LSB has been modified, by Lemma 1 we have

Volume 2, Issue 3, 2026
https://doi.org/10.5281/zenodo.22165095

$$\mu(\tilde{C}) = \mu(C)$$

The receiver computes

$$T_R = HMAC_{K_A}(\mu(\tilde{C})) = HMACKA(\mu(C)) = T$$

Since $\tilde{T} \neq T$, we have $T_E \neq T_R$.

Therefore, verify $(\tilde{C}, K_A) = 0$ ■

This implies that the authentication was unsuccessful.

Proposition 3:

If $C' \neq C$ but $\mu(C') = \mu(C)$, then $HMACKA(\mu(C')) = HMACKA(\mu(C))$

Proof.

The assumption gives $\mu(C') = \mu(C)$

Using the same secret key and the deterministic nature of HMAC, we have

$$HMACKA(\mu(C')) = HMACKA(\mu(C))$$

Therefore, $T' = T$. ■ This is a limitation of the scheme.

Proposition 4

Let C' be a modified image satisfying $\mu(C') \neq \mu(C)$. Then

$$Pr[HMAC_{K_A}(\mu(C')) = HMAC_{K_A}(\mu(C))] \leq \varepsilon$$

Proof.

The sender generates $T = HMAC_{K_A}(\mu(C))$

Suppose the attacker changes the image so that $\mu(C') \neq \mu(C)$

The receiver recomputes $T' = HMAC_{K_A}(\mu(C'))$

For $T = T'$,

This implies that $HMAC_{K_A}(\mu(C')) = HMAC_{K_A}(\mu(C))$

For a secure HMAC, finding a valid collision or forgery is assumed computationally infeasible.

$$Pr[T' = T] \leq \varepsilon$$

IV. CONCLUSION

This paper presented a mathematical authentication framework for encrypted images based on the integration of MSB extraction, HKDF-based authentication-key derivation, HMAC authentication-tag generation, and LSB embedding. The authentication algorithm extracts the leftmost bit of each encrypted pixel to form an MSB-derived authentication representation. An authentication-specific key is derived from a shared secret using HKDF and a dedicated authentication context, after which HMAC is applied to generate an authentication tag. The tag is embedded in designated least significant bit positions of the encrypted image. During verification, the embedded tag is extracted and compared with a tag recomputed from the reconstructed MSB representation using the corresponding authentication key. The mathematical analysis established several fundamental properties of the proposed construction. In particular, LSB embedding preserves the selected leftmost MSB representation, thereby enabling consistent reconstruction of the authenticated data during verification. The correctness result demonstrates that a properly generated and unmodified authenticated encrypted image is accepted by the verification procedure. The analysis also shows that modifications affecting the authenticated MSB representation are expected to invalidate authentication. In addition, direct modification of the embedded authentication tag results in verification failure unless a valid replacement tag is produced. Future

Volume 2, Issue 3, 2026
https://doi.org/10.5281/zenodo.22165095

work can extend the scheme to generate separate authentication tags for each image block, enabling precise tamper localization.

REFERENCES

[1] *Gaurav. (2015) A New Method for Image Steganography Using LSB and MSB.* International Journal of Recent Research Aspects, ISSN: 2349-7688, 2, 4.

[2] Kiswara, A. Santoso, M., Lailatun, N. & Moh. H. (2021): *Image Authentication based on magic square*: proceedings of the international conference on mathematics, geometry, statistics and computation (IC-MaGeStiC)

[3] Muhammad, Z. A., Omer, R., Hafiz, M. H., Waqas, S., Tenvir, A.& Gyu S. C. (2024). *Elevating Image Steganography: A Fusion of MSB Matching and LSB Substitution for Enhanced Concealment Capabilities. Tech Science Press, Computers, Materials & Continua, Vol. 79, Issue 2, pages 2923–2943.*

[4] Muhammad, R.Y., Suryadi, M.T, Catur, A. & Muhammad F. S. (2024). *Image-to-Image Steganography with Josephus Permutation and Least Significant Bit (LSB) 3-3-2 Embedding*. Applied Sciences, 14(16), 7119. https://doi.org/10.3390/app14167119

[5] Rohit, A., Kuldeep, N. T., Ranjeet, K. S., Nitin, A. S.& Umesh, G. (2024). *A secure image authentication technique based on sparse approximation and quantum mechanism.* Science Direct. Digital Signal Processing.

[6] Solomon, O. A. and Adebanke, O. (2015). *On the Image Quality and Encoding Times of LSB, MSB and Combined LSB-MSB Steganography Algorithms Using Digital Images*. International Journal of Computer Science and Information Technology 6.

[7] Yang, H., Sun, X. & Sun, G. (2009). *A High-Capacity Image Data Hiding Scheme Using Adaptive LSB Substitution*. Radio engineering, 18, 4, 509–516.

[8] Zaid, B. F., Abid, I, Furqan, R., Isabel de la Torre Diez, Daniel, G., Manuel, M. V. & Imran, A. (2023). *Image Watermarking Using Least Significant Bit and Canny Edge Detection.* Sensors (Basel), National Library of Medicine.

https://doi.org/10.5281/zenodo.22165095