JCTA Journal of Computing Theories and Applications ISSN:3024-9104

*Research Article*

# Block-wise Authenticated DNA-based Image Encryption with Tamper Localization using HKDF-Derived Keys

**Bagus Satrio Waluyo Poetro [1,2,*], Kusworo Adi [3], and Aris Puji Widodo [4]**

[1] Doctoral Program in Information Systems, Postgraduate School, Universitas Diponegoro, Semarang 50241, Indonesia; e-mail : bagussatriowp@students.undip.ac.id
[2] Department of Informatics, Faculty of Industrial Technology, Universitas Islam Sultan Agung, Semarang 50112, Indonesia; e-mail : bagusswp@unissula.ac.id
[3] Department of Physics, Faculty of Science and Mathematics, Universitas Diponegoro, Semarang 50275, Indonesia; e-mail : kusworoadi@lecturer.undip.ac.id
[4] Department of Informatics, Faculty of Science and Mathematics, Universitas Diponegoro, Semarang 50275, Indonesia; e-mail : arispw@gmail.com
* Corresponding Author : Bagus Satrio Waluyo Poetro

**Abstract:** Confidentiality alone cannot establish whether a medical, forensic, cloud-stored, or remotely sensed image has been modified during transmission, while a single global authentication verdict cannot identify the affected region. This paper presents a block-wise authenticated DNA-based image encryption framework that integrates a DNA-chaos confidentiality core with pre-decryption integrity verification and spatial tamper localization. An authenticated ephemeral X25519 key exchange establishes a session secret, which is expanded by HKDF-SHA256 into four transcript-bound, domain-separated subkeys for permutation, DNA operations, diffusion, and authentication. Chaotic seeds are derived from a canonical plaintext hash and block coordinates, preserving strong plaintext differential sensitivity while confining post-encryption modifications to the affected blocks. Ciphertext integrity is enforced using a block-wise encrypt-then-MAC construction with 128-bit truncated HMAC-SHA256 tags that authenticate both the canonical header and each ciphertext block. A reduction-based security argument shows that the authentication layer provides ciphertext integrity and conditionally upgrades an IND-CPA encryption core to IND-CCA security under standard HKDF and HMAC assumptions, while the confidentiality claim remains explicitly conditional on the encryption core. Experiments on 30 tuberculosis chest radiographs across five independent sessions achieved a ciphertext entropy of 7.9972, near-zero adjacent-pixel correlation, 99.61% NPCR, and 33.47% UACI. Across five representative tampering attacks, the proposed framework achieved an observed 100% block-level true-positive rate, 0% false-positive rate, and required only 2.847 ± 0.258 ms for block-wise authentication with 6.25% tag overhead using the default 16×16 block configuration. These results demonstrate that the proposed framework effectively combines statistical confidentiality, modern cryptographic key management, and reliable block-level tamper localization within a unified authenticated image encryption architecture.

**Received: June, 13th 2026**
**Revised: July, 13th 2026**
**Accepted: July, 15th 2026**
**Published: July, 30th 2026**

**Keywords:** Authenticated image encryption; Chaotic image encryption; DNA cryptography; Encrypt-then-MAC; HKDF; HMAC; Tamper localization; X25519.



## 1. Introduction

The routine use of digital images in telemedicine, cloud diagnostics, forensic evidence, remote sensing, and secure image transmission makes three security properties jointly essential. Confidentiality prevents unauthorized disclosure, integrity reveals unauthorized modification, and authenticity associates protected data with an authenticated session or sender. In chest radiography, an undetected local edit may conceal or imitate pathology. In remote sensing, it may alter mapped objects, while in forensic workflows it may invalidate digital provenance. Encryption alone, however, only prevents unauthorized disclosure and provides no evidence that the ciphertext or its associated metadata has remained unchanged.

Chaos- and DNA-based image encryption remains attractive because chaotic permutation and diffusion effectively disrupt the high spatial correlation and redundancy of image data, whereas DNA encoding provides a reversible symbolic transformation layer [1]–[4]. Recent studies have further strengthened key management by introducing ephemeral elliptic-curve key agreement together with HKDF-based domain separation, replacing static or repeatedly reused chaotic parameters with fresh session-dependent keys [5]–[8].

Despite these advances, improved key management alone does not protect against active attacks. A receiver must be able to reject modified ciphertext before exposing any decrypted image, while all security-relevant metadata must be authenticated together with the ciphertext. Conventional authenticated encryption, including ChaCha20-Poly1305, provides confidentiality and integrity at the message level when correctly deployed [9], [10]. Its typical single-record design, however, produces only a global accept-or-reject decision. This is not a limitation of AEAD itself, but rather a mismatch in localization granularity. Although per-block AEAD can localize failures, it requires block-specific nonce management and associated-data construction, effectively replacing rather than extending an existing DNA-chaos encryption engine.

Tamper localization provides practical value beyond merely detecting modification. A global authentication failure only indicates that an image should not be trusted, whereas a spatial localization map identifies the affected anatomical, evidentiary, or geographic region, supports incident triage, and enables verified-block forensic inspection without treating corrupted content as authentic. Fragile watermarking and block-hash approaches can also provide localization [11], [12]; however, these techniques operate in the plaintext domain or embed additional information into the host image, representing a different deployment model from ciphertext authentication.

**Table 1.** Comparison with representative authenticated image-protection approaches.

| Approach | Integrity / Authentication Mechanism | Localization | Gap Relative to This Work |
|---|---|---|---|
| Global ChaCha20-Poly1305 [9], [10] | Standard AEAD with one message tag and associated data | No (default deployment) | Provides strong general-purpose authenticated encryption, but a single authentication record cannot identify the affected image region. Block-wise deployment additionally requires unique nonces and message framing. |
| Jasra and Moon [13] | Dynamic DNA/hyperchaos with ECC and signature-based authenticated exchange | Not reported | Authenticates key exchange and image provenance but does not provide context-bound pre-decryption block authentication. |
| Zhiqiang et al. [14] | ECC/HKDF, ECDSA, and a global HMAC over the plaintext hash | No | Performs global verification after decryption and does not generate a ciphertext-block tamper map. |
| Adi et al. [11]; MATADOR [12] | Fragile watermark or block-specific content hash | Yes | Designed for plaintext-domain authentication and recovery rather than jointly providing session-keyed ciphertext confidentiality. |
| Bhargav and Singh [15] | Structure-based block-wise image authentication | Yes | Employs a different authentication architecture and does not address the interaction between HKDF-managed key separation and differential sensitivity in DNA-chaos encryption. |
| Proposed framework | Transcript-bound 128-bit HMAC-SHA256 over the authenticated header, coordinates, dimensions, length, and ciphertext | Yes (before decryption) | Extends a DNA-chaos encryption engine with cryptographic block-wise localization while providing a formal security argument whose confidentiality guarantee is explicitly conditional on the underlying encryption core. |

The remaining research challenges are therefore coupled rather than independent. Position-preserved independent blocks are required to localize post-encryption modifications, yet block independence suppresses inter-block propagation. Furthermore, strict HKDF-based key separation may reduce differential sensitivity when the chaotic keystream is independent of the plaintext [5]. The proposed framework separates these two propagation mechanisms. A plaintext modification before encryption changes the global plaintext hash and consequently all block-specific chaotic seeds, whereas a ciphertext modification after encryption remains confined to the affected block and is identified through its context-bound authentication tag. Table 1 summarizes how this combination differs from representative authenticated image-protection approaches.

Based on the identified gaps, this paper makes four traceable contributions.

- Gap G1 (confidentiality without active-attack verification) → Contribution C1: A block-wise encrypt-then-MAC layer with a dedicated HKDF-derived authentication key that verifies the ciphertext together with all security-relevant header fields before decryption.
- Gap G2 (global detection without spatial evidence) → Contribution C2: Coordinate-bound block-wise HMAC-SHA256 authentication tags that generate a tamper-localization map while preventing block relocation, swapping, truncation, and cross-message splicing.
- Gap G3 (localization-induced independence versus differential sensitivity) → Contribution C3: Plaintext-coupled block-specific chaotic seeding that restores near-ideal NPCR and UACI during honest re-encryption while confining post-encryption channel modifications to the affected blocks.
- Gap G4 (limited protocol and practical evaluation) → Contribution C4: A complete protocol specification, a conditional reduction-based security argument, component-level ablation, block-size, security and latency analyses, an AEAD microbenchmark, and channel-impairment assessment.

Collectively, these contributions integrate modern cryptographic key management, authenticated ciphertext protection, and block-level tamper localization within a unified DNA-chaos image encryption framework while explicitly separating empirical statistical evidence from formal cryptographic security claims. The remainder of this paper is organized as follows. Section 2 reviews related work and further refines the identified research gap. Section 3 presents the proposed framework together with its security model. Section 4 reports the experimental evaluation, including confidentiality, localization, ablation, overhead, and channel analyses. Section 5 discusses the practical implications and limitations of the proposed framework, and Section 6 concludes the paper.

## 2. Related Work

### 2.1. Chaos- and DNA-based Image Encryption

Hybrid DNA-chaos image encryption combines chaotic permutation with DNA-domain diffusion to disrupt spatial correlation and produce noise-like ciphertexts [1]–[4], [16], [17]. Recent studies have further improved these frameworks by introducing hyperchaotic confusion-diffusion mechanisms [18]–[21] together with cryptographic hashing to increase plaintext sensitivity and key dependence. These approaches consistently report favorable statistical characteristics, including high entropy, low adjacent-pixel correlation, and near-ideal NPCR and UACI values.

Although these metrics provide useful empirical evidence regarding image transformation quality, they do not constitute formal cryptographic guarantees. In particular, entropy, correlation, NPCR, and UACI cannot replace security notions such as IND-CPA, IND-CCA, or ciphertext integrity. Accordingly, the present work treats these statistical measures as empirical diagnostics while explicitly considering the confidentiality claim to be conditional on the underlying encryption core. This distinction separates statistical image security from formal cryptographic security and forms the basis of the subsequent authentication design.

### 2.2. Key Management for Image Encryption

Modern key-management techniques reduce static-key reuse and cross-stage key dependence. X25519 provides an approximately 128-bit classical security target using compact

32-byte public keys [8], while HKDF extracts pseudorandom session keys from the ECDH shared secret and expands them into labeled, context-specific subkeys [7], [22]. Previous DNA-chaos image encryption schemes have adopted ECDH/HKDF-based session keys [5], [6], whereas other image-encryption studies combined ECC/HKDF with global authentication mechanisms [14]. Machine-learning-assisted key generation has also been investigated [23], although it remains orthogonal to the standards-based authenticated ephemeral key agreement considered in this work.

Key separation alone, however, does not establish peer authenticity. X25519 provides key agreement but does not authenticate the communicating parties. Consequently, the exchanged ephemeral public keys must be verified using certificates, digital signatures, preshared authentication keys, or an already authenticated secure channel. This assumption is explicitly incorporated into the proposed protocol and is discussed in Section 3.2. While modern key management effectively strengthens session-key generation, it does not by itself address ciphertext authentication or spatial tamper localization.

### 2.3. Integrity and Authenticated Encryption

Authenticated encryption is formally defined through both confidentiality and integrity notions. Bellare and Namprempre showed that encrypt-then-MAC using independent keys provides ciphertext integrity and upgrades an IND-CPA encryption scheme to IND-CCA security when the MAC is unforgeable [24]. HMAC provides the required keyed authentication primitive [25], while ChaCha20-Poly1305 represents a standardized AEAD construction [9], [10]. A single global authentication tag is appropriate when an image is treated as one indivisible record, but it cannot identify which image region has been modified. Per-block AEAD offers a practical alternative; however, it requires careful management of nonce uniqueness together with block ordering, dimensions, and message context through associated data.

Image-specific localization has commonly been addressed using fragile watermarking, reversible authentication, or block-hash techniques. Recent methods can detect and localize modifications and, in some cases, recover the altered content [11], [12], [15]. Nevertheless, these approaches generally operate in the plaintext domain, prioritize imperceptibility or recovery, or omit an authenticated ephemeral ciphertext layer. DNA-based encryption schemes with authentication have also been reported [13], [14], yet the reviewed methods do not jointly provide transcript-bound pre-decryption block authentication, spatial tamper localization, and a mechanism that mitigates the differential-sensitivity degradation introduced by HKDF-managed key separation. Consequently, authenticated encryption and image-specific localization remain largely separate research directions within DNA-chaos image encryption.

### 2.4. Research Gap

The refined research gap is not the absence of image authentication itself, but rather the absence of a unified architecture that combines multiple security properties within an HKDF-managed DNA-chaos framework. Specifically, existing approaches do not simultaneously provide: (i) authenticated ciphertext and security-relevant metadata before decryption, (ii) cryptographic block-level tamper localization without modifying the plaintext representation, (iii) protection against block relocation, truncation, and cross-message splicing, and (iv) near-ideal plaintext differential sensitivity despite preserving independent block processing.

These requirements are inherently interdependent. Improving localization through block independence tends to reduce inter-block diffusion, whereas strengthening key separation may reduce differential sensitivity when chaotic initialization is independent of the plaintext. The framework proposed in this paper is therefore designed specifically to address these coupled challenges by integrating authenticated key management, plaintext-coupled chaotic initialization, and block-wise ciphertext authentication within a single architecture.

## 3. Proposed Method

### 3.1. System Overview

Figure 1 illustrates the overall workflow of the proposed framework, which is organized around three complementary objectives: authenticated session establishment, confidential image encryption, and block-level integrity verification with tamper localization. The sender first

establishes an authenticated session and derives four domain-separated subkeys using X25519 and HKDF-SHA256. These subkeys independently drive permutation, DNA operations, diffusion, and authentication. The plaintext image is then canonicalized, partitioned into position-preserved blocks, encrypted using block-specific chaotic seeds, and authenticated together with the canonical header. On the receiver side, every authentication tag is verified before decryption to generate a block-level tamper map and enforce the selected acceptance policy.

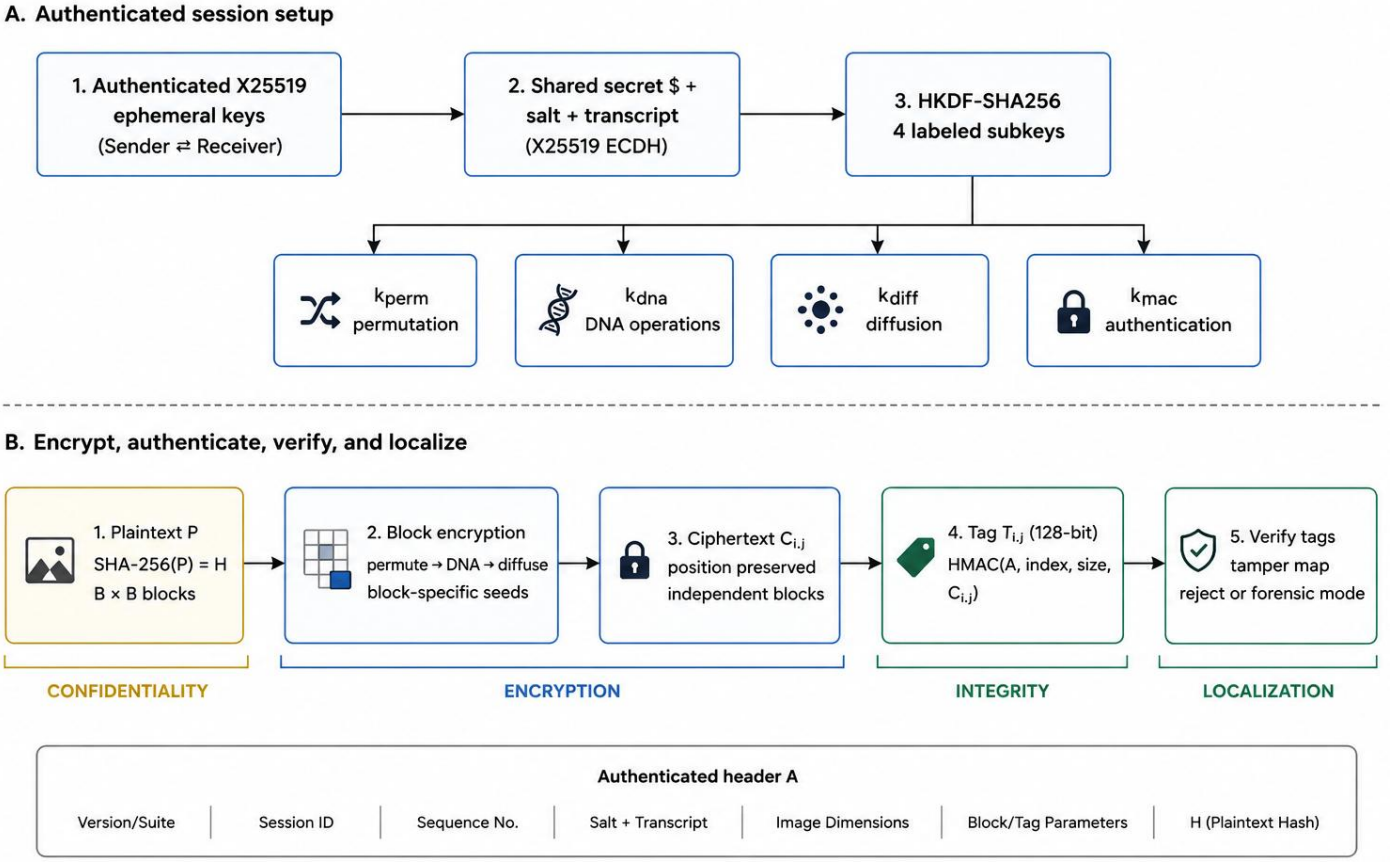



**Figure 1.** Workflow of the proposed block-wise authenticated DNA-chaos image encryption framework.

This processing order is intentional. Authenticated key establishment remains independent of the plaintext, plaintext coupling influences only chaotic seed generation, and authentication protects the transmitted representation rather than the decrypted image. Consequently, confidentiality, integrity, and tamper localization are implemented through distinct cryptographic components with independently derived HKDF subkeys instead of relying on statistically random-looking ciphertext as evidence of authenticity.

This separation also simplifies the subsequent security analysis. Each component can be evaluated according to its own security objective while remaining interoperable within a unified authenticated image encryption framework. The following subsections describe these components in detail, beginning with authenticated session establishment and key derivation.

### 3.2. Key Establishment and Derivation

The proposed framework adopts ephemeral Elliptic Curve Diffie-Hellman (ECDH) over Curve25519 (X25519) to establish a unique shared secret for each communication session [8]. Two communicating parties independently generate ephemeral private scalars $a$ and $b$, and exchange the corresponding public keys.

$$A = aG, \quad B = bG \tag{1}$$

where $G$ denotes the Curve25519 base point. Each party verifies the peer's ephemeral public key through a certificate/signature, a pre-shared authentication key, or an already authenticated channel, then computes the shared secret.

$$S = abG, \tag{2}$$

The implementation rejects an all-zero shared secret and securely erases the ephemeral private scalars immediately after use. Ephemeral X25519 is adopted because it targets approximately 128-bit classical security, uses compact 32-byte public keys, is standardized for interoperable protocols, and is designed for efficient implementation with regular arithmetic [8]. Compared with a conventional uncompressed P-256 public-key encoding, the 32-byte X25519 representation reduces handshake overhead while providing a comparable classical security target. Certified P-256 nevertheless remains a valid deployment alternative [8], [26]. Accordingly, X25519 is not claimed to be intrinsically stronger; it is selected primarily for its compact representation, implementation maturity, and broad protocol adoption.

A fresh 32-byte salt is generated for every session, and HKDF-SHA256 extracts a pseudorandom key from the shared secret $S$ [7]:

$$PRK = HKDF - Extract(salt, S) \tag{3}$$

Four independent 32-byte subkeys are subsequently derived using distinct transcript-bound labels:

$$ki = HKDF - Expand(PRK, infoi, L), \quad i \in I, \tag{4}$$

where $I = \{perm, dna, diff, mac\}$, $L = 32$ bytes, and $info_i = \text{"JCTA} - \text{DNA} - \text{AE} - \text{v1"} \,||\, session\_id \,||\, transcript\_hash \,||\, label\_i$

The transcript hash includes both ephemeral public keys together with their role ordering. HKDF extraction mitigates any non-uniformity in the ECDH output, whereas labeled expansion prevents accidental cross-stage key reuse [7]. Consequently, permutation, DNA operations, diffusion, and authentication operate with cryptographically separated key material while remaining reproducible within the authenticated session.

The canonical authenticated header $A$ contains the protocol version, algorithm suite, 128-bit session identifier, 64-bit message sequence number, salt, ephemeral-key transcript, image dimensions, channel count, block size, tag length, and plaintext hash $H$. Receivers maintain the largest accepted sequence number, or an equivalent replay window, for each active session because a MAC alone cannot distinguish a legitimate replay from the original transmission.

Peer authentication during key exchange is essential. X25519 without authenticated public keys remains vulnerable to an active man-in-the-middle adversary that establishes separate session secrets with the sender and receiver. The image HMAC authenticates data relative to the resulting session key, but it cannot retroactively authenticate an unauthenticated key-establishment process. This assumption is therefore treated as part of the system model rather than as a property provided by the authentication layer itself.

### 3.3. Plaintext-Coupled Chaotic Seeding

Plaintext coupling is introduced to recover global differential sensitivity without introducing inter-block ciphertext chaining. The image is first canonicalized as unsigned 8-bit raster bytes in row-major order, preceded by fixed-width big-endian fields representing the image height, width, and channel count. Let $P_{can}$ denote this canonical representation, and compute

$$H = \text{SHA256}(P_{can}) \tag{5}$$

The resulting hash becomes the common plaintext-dependent context for all block-specific seed generation. Consequently, every block remains independently encrypted while preserving global sensitivity to plaintext modifications.

For encryption stage $s \in \{perm, dna, diff\}$ and block coordinates $(i, j)$, the digest and corresponding chaotic parameters are computed as

$$d_{s,i,j} = \text{HMAC} - \text{SHA256}(ks, \text{seed} \parallel H \parallel uint16(i) \parallel uint16(j)),$$
$$(x_{0,s,i,j}, r_{s,i,j}) = \Phi(d_{s,i,j}) \quad (6)$$

This construction assigns an independent chaotic trajectory to every encryption stage and image block while maintaining deterministic reconstruction during decryption. The mapping $\Phi$ is fully specified as follows. Let $u_0 = OS2IP(d[0:8])$, $u_i = OS2IP(d[8:16])$ where both values are interpreted as unsigned big-endian integers. The initial state and control parameter are then defined as $x_0 = \frac{u_0+0.5}{2^{64}}$, $r = 3.9 + 0.1\left(\frac{u_1}{2^{64}}\right)$. These definitions place $x_0$ strictly within the interval $(0,1)$ and $r$ within $[3.9,4.0)$. Incorporating the block coordinates $(i,j)$ prevents equally sized blocks from sharing identical stage streams. The first 1000 chaotic states are discarded as burn-in before keystream generation begins.

Since the plaintext hash $H$ depends on every byte of the canonical image representation, a one-bit modification to the plaintext changes the seeds of every block simultaneously. In contrast, a ciphertext modification occurring after encryption does not alter $H$ and therefore remains confined to the affected block. This separation between plaintext sensitivity and ciphertext locality forms the basis for combining strong differential behavior with reliable block-level tamper localization.

Each encryption stage employs the Logistic-Sine Map (LSM),

$$xn + 1 = \left(r\, xn(1 - xn) + \frac{(4 - r)\sin(\pi xn)}{4}\right) mod\ 1, \quad (7)$$

After the burn-in period, each chaotic state is converted into a byte according to $b_n = [256x_n]$, while permutation indices are obtained by stably sorting the next $B^2$ real-valued states, using the original position to resolve exact ties. These implementation details eliminate ambiguity and ensure deterministic execution across independent implementations. They do not, however, transform the floating-point LSM into a standardized cryptographic pseudorandom generator. Accordingly, the reduction presented in Section 3.6 treats the confidentiality of the DNA-chaos encryption core as an explicit assumption rather than deriving formal cryptographic security directly from chaos statistics.

### 3.4. Block-wise DNA-Chaos Encryption

The image is partitioned into non-overlapping $B \times B$ blocks. Block positions are preserved throughout encryption, each block uses independent stage streams, and no inter-block chaining is introduced. When the image dimensions are not divisible by $B$, each edge block retains its actual height and width. These dimensions, together with the ciphertext length, are authenticated as part of the corresponding block tag, thereby avoiding ambiguous zero padding. This organization is deliberately adopted so that post-encryption modifications remain spatially localized while preserving independent block processing.

The block-wise encryption procedure consists of three sequential stages: permutation, DNA-domain transformation, and diffusion. Each stage provides a complementary confidentiality function while preserving compatibility with the subsequent authentication layer.

1. Permutation: For block $(i,j)$, the $k_{perm}$-derived chaotic sequence generates a stable within-block permutation $\pi$ by sorting $B^2$ chaotic states. Only pixel positions inside the current block are reordered. This stage primarily provides spatial confusion by disrupting local pixel correlation, while intentionally remaining independent of the integrity mechanism.
2. Each 8-bit pixel value is divided into four 2-bit symbols. A rule index $1 + (b_n \bmod 8)$ selects one of the eight Watson-Crick-consistent encoding rules listed in Table 2. Each symbol is encoded, XORed with a 2-bit chaotic DNA keystream, and subsequently decoded using the same rule. Employing all eight reversible mappings avoids a fixed DNA representation and increases symbolic diversity during encryption. Nevertheless, this DNA stage is inherited from the confidentiality component; the novelty of the proposed framework lies in its authenticated block-wise architecture rather than in introducing new DNA primitives.

**Table 2.** Watson-Crick-consistent DNA encoding rules used in the proposed framework.

| Rule | 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| 1 | A | C | G | T |
| 2 | A | G | C | T |
| 3 | C | A | T | G |
| 4 | C | T | A | G |
| 5 | G | A | T | C |
| 6 | G | T | A | C |
| 7 | T | C | G | A |
| 8 | T | G | C | A |

3. Diffusion: The DNA-stage output is XORed bytewise with the $k_{diff}$ -derived block keystream,

$$C = D2 \oplus KS_{diff} \tag{8}$$

to produce the ciphertext block $C_{ij}$. During decryption, inverse diffusion, DNA decoding, and inverse permutation are applied using the authenticated plaintext hash $H$ recovered from the verified header. Accordingly, the sender performs the sequence $P_{can} \rightarrow \text{hash} \rightarrow \text{partition} \rightarrow \text{permute} \rightarrow \text{DNA} \rightarrow \text{diffuse}$, whereas the receiver first authenticates the canonical header $A$ together with every ciphertext block before executing the inverse operations. This processing order ensures that ciphertext integrity is verified prior to plaintext recovery, preventing modified ciphertext from entering the decryption stage.

### 3.5. Block-wise Authentication and Tamper Localization

Following encryption, every ciphertext block is authenticated using the encrypt-then-MAC paradigm [24]. This design aligns the authentication granularity with the position-preserved block structure, allowing integrity verification and spatial tamper localization to be performed simultaneously before decryption.

For block $(i,j)$, let $h_{ij}$ and $w_{ij}$ denote the actual block dimensions, and let $\text{len}(\cdot)$, represent a fixed-width length prefix. The authentication tag is computed as

$$T_{ij} = Trunc_{128}\Big(HMACSHA256\big(k_{mac}, \text{len}(A) \parallel A \parallel \text{"block"} \parallel i \parallel j(9) \parallel h_{ij} \parallel w_{ij} \parallel \text{len}(C_{ij}) \parallel C_{ij}\big)\Big) \tag{9}$$

All integer fields use fixed-width big-endian encoding. Authenticating the canonical header A protects the plaintext hash, image dimensions, algorithm identifiers, protocol parameters, sequence number, and ephemeral-key transcript against modification. Binding block coordinates and dimensions prevents relocation attacks, while authenticating the ciphertext length together with the declared block grid prevents truncation. The inclusion of the session identifier and message sequence number further prevents cross-message splicing.

The authentication tag is truncated to $\tau$=128 bits and verified using constant-time comparison. This choice matches the approximately 128-bit classical security target of X25519, reduces storage by half compared with a full 256-bit HMAC output, and retains a maximum random-guess acceptance probability of $2^{-128}$ for each modified block [25], [27]. For an image containing N blocks, a conservative union bound gives an upper bound of $N \times 2^{-\tau}$ for at least one accidental acceptance when every block is modified.

The receiver independently recomputes each authentication tag and constructs the block tamper map.

$$M_{ij} = 1\big[T'_{ij} \neq T_{ij}\big] \tag{10}$$

where $1[\cdot]$ denotes the indicator function. The localization result is obtained from the union of all blocks for which $M_{ij}$=1. A modified ciphertext block is accepted only if an adversary successfully forges its authentication tag or a truncated-tag collision occurs. Consequently,

detection is not mathematically deterministic, but the residual acceptance probability remains negligible under the stated security assumptions. The default block size of $B = 16$ is selected from the measured trade-off knee, providing 16×16 spatial localization granularity with 6.25% tag storage overhead, a region IoU of 0.664, and a mean authentication time of 2.847 ms in the reference Python implementation. A smaller block size ($B = 8$) offers finer localization at the cost of approximately 25% tag overhead, whereas larger blocks reduce storage and computational overhead while decreasing localization resolution.

### 3.6. Threat Model

The proposed framework is analyzed under a network adversary that controls the communication channel. The adversary may observe, drop, replay, reorder, inject, and adaptively modify transmitted messages [28]. It may also request encryptions and submit candidate ciphertexts for verification. However, the adversary is not assumed to compromise endpoint memory, authenticated long-term peer credentials, or freshly generated X25519 private scalars. Side-channel attacks, endpoint malware, denial-of-service attacks, and post-quantum adversaries are outside the scope of this work. Random transmission errors are evaluated separately in Section 4.9 because the authentication layer is intentionally designed to treat accidental corruption and malicious modification identically.

The security argument is based on the following assumptions.

- (A1) Peer ephemeral public keys are authenticated, and the X25519/HKDF output is computationally indistinguishable from independent subkeys for an uncompromised session.
- (A2) The DNA-chaos encryption core $E$, when instantiated with fresh session material and the authenticated header context, satisfies IND-CPA security.
- (A3) HMAC-SHA256 behaves as a pseudorandom function and is strongly unforgeable under chosen-message attack. Truncation limits random-guess security to $\tau$ bits.
- (A4) Every ($session\_id, sequence\ number$) pair is unique, and replay protection is correctly enforced.

These assumptions separate the properties established by standard cryptographic primitives from those explicitly assumed for the underlying DNA-chaos confidentiality mechanism. Consequently, the following propositions validate the authenticated composition without claiming a standalone proof for the encryption core itself.

Proposition 1 (Ciphertext Integrity): Under assumptions A1 and A3–A4, the proposed context-bound block authentication construction satisfies INT-CTXT security. For $q_v$ verification attempts over at most N blocks per image, the forgery advantage is bounded by the HKDF/HMAC distinguishing and forgery advantages together with $\frac{q_v N}{2^\tau}$. Any newly accepted modified image necessarily contains at least one previously unseen tuple $(A, i, j, h, w, \mathrm{len}(C), C)$ accompanied by a valid authentication tag. Replacing the derived MAC key with an independent random key reduces successful acceptance to either an HMAC forgery or a random $\tau$-bit guess. Furthermore, authenticating the header, block coordinates, dimensions, and ciphertext length prevents valid authentication data from being reused through relocation, truncation, protocol downgrade, or cross-message splicing.

Proposition 2 (Conditional Adaptive Chosen-Ciphertext Privacy): Under assumptions A1–A4, the encrypt-then-MAC construction provides IND-CCA privacy provided that the underlying encryption core $E$ is IND-CPA secure. The verification-and-decryption oracle rejects every candidate containing a newly invalid authentication tag before releasing any plaintext. Except when Proposition 1 fails, accepted verification queries correspond only to previously generated ciphertexts or legitimate replays that are subsequently rejected by replay-state enforcement. The remaining security challenge therefore reduces directly to the IND-CPA security of $E$, consistent with the generic composition theorem of Bellare and Namprempre [24]. Together, Propositions 1 and 2 establish that the proposed authentication layer strengthens the security guarantees of the overall construction without requiring modifications to the underlying DNA-chaos encryption algorithm. The formal contribution therefore lies in the authenticated composition rather than in proving new security properties of the confidentiality primitive itself.

Scope of the Proof: The preceding propositions formally validate the authentication composition but do not establish assumption A2 for the floating-point DNA-chaos encryption core. Statistical image-security measures, including entropy, adjacent-pixel correlation, NPCR, and UACI, provide useful empirical evidence but cannot establish IND-CPA security. Accordingly, this work claims formal ciphertext integrity together with a conditional IND-CCA upgrade under the stated assumptions, while treating a formal proof of the confidentiality core as future work.

This distinction intentionally limits the security claims of the proposed framework. Rather than interpreting favorable image statistics as cryptographic proof, the manuscript explicitly separates empirical evaluation from formal security analysis. Such separation provides a clearer interpretation of the achieved guarantees and avoids overstating the security properties of the underlying chaos-based encryption mechanism.

## 4. Results and Discussion

### 4.1. Experimental Setup

The proposed framework was evaluated using the Tuberculosis (TB) Chest X-ray Database [29], [30]. Thirty chest radiographs, comprising 15 normal and 15 tuberculosis images, were converted to 8-bit grayscale and resized to 256×256 pixels. Unless otherwise stated, the block size was fixed at B=16 and the authentication tag length at $\tau$=128 bits. Every experiment was repeated over five independently negotiated sessions with fresh salts, resulting in a total of 150 independent encryption instances. Differential-security measurements were obtained from ten single-pixel, single-bit perturbation trials for each image. Unless otherwise specified, all reported results are presented as the mean ± standard deviation.

The selected dataset serves as a pilot security-evaluation benchmark rather than a clinical population sample. Accordingly, the reported statistical observations are limited to the evaluated grayscale chest radiographs and image resolution and are not intended to establish modality-independent generalization.

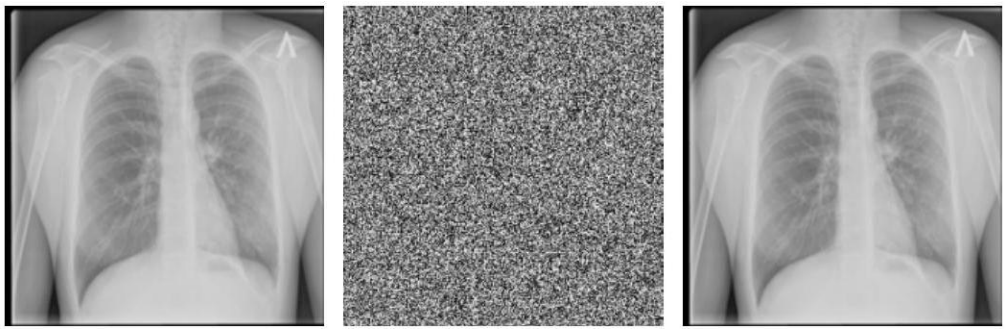
The authentication and AEAD microbenchmark were designed to isolate the overhead introduced by the proposed protection layer because the underlying DNA-chaos encryption core remains unchanged from the confidentiality baseline. The benchmark uses 30 independent payloads of 65,536 bytes with 40 repeated measurements per payload on an Intel Xeon E5-2673 v4 processor (2.30 GHz), Linux x86-64, Python 3.12.13, and cryptography 46.0.0 under single-threaded execution. All reported execution times correspond to one encrypted image. The combined X25519 key establishment and 128-byte HKDF expansion require a median setup time of 0.212 ms. These reference Python measurements are intended to characterize the relative computational overhead of the proposed authentication layer rather than optimized native or hardware-assisted performance.

Integrity evaluation considers five representative ciphertext manipulations: region replacement, scattered bit flips, copy-move, salt-and-pepper corruption, and a single-bit ciphertext modification. A block is counted as a true positive when its ciphertext differs from the original and the corresponding authentication tag fails verification. Conversely, an unchanged block whose verification fails is counted as a false positive. Localization accuracy is additionally evaluated using pixel-level IoU between the union of flagged blocks and the pixel attack mask. Consequently, attacks affecting only a few pixels are expected to produce relatively low pixel IoU even when the correct block is successfully identified.

### 4.2. Visual and Statistical Confidentiality

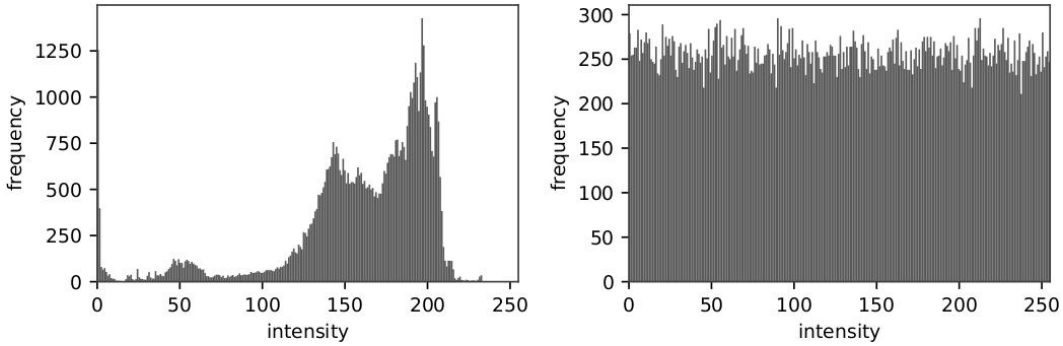
The visual confidentiality of the proposed framework is illustrated in Figure 2. The ciphertext appears visually indistinguishable from random noise, whereas the decrypted image is identical to the original plaintext, demonstrating lossless recovery. Then, the corresponding intensity histograms are presented in Figure 3. The clustered histogram of the plaintext is transformed into an approximately uniform ciphertext distribution spanning the full 8-bit intensity range. Next, the adjacent-pixel correlation distributions are shown in Figure 4. The strong linear correlation observed in the plaintext disappears after encryption, indicating effective decorrelation of neighboring pixels. These visual observations provide empirical evidence that the proposed transformation substantially reduces the statistical structure of the

tested images. As discussed in Section 3.6, however, histogram uniformity and pixel decorrelation are evaluated only as statistical diagnostics and are not interpreted as formal cryptographic security guarantees.



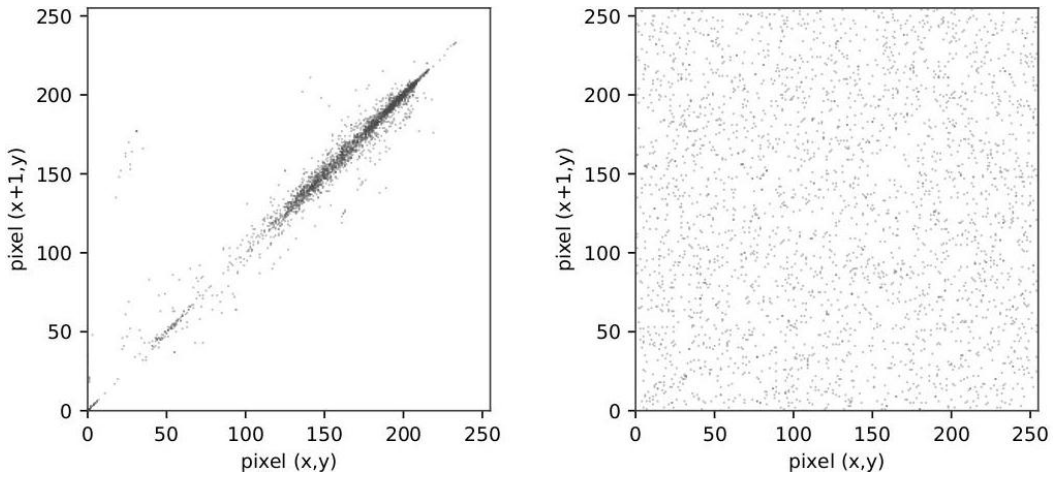
(a) (b) (c)

**Figure 2.** End-to-end encryption results: (**a**) plaintext; (**b**) ciphertext; and (**c**) lossless decrypted image.




(a) (b)

**Figure 3.** Intensity histograms of (**a**) the plaintext and (**b**) the ciphertext.




(a) (b)

**Figure 4.** Adjacent-pixel correlation of (**a**) the plaintext and (**b**) the ciphertext.

Table 3 summarizes the quantitative statistical confidentiality results obtained over all images and independent sessions. The average ciphertext entropy reaches 7.9972 ± 0.0002, approaching the theoretical 8-bit maximum. Horizontal, vertical, and diagonal adjacent-pixel correlations remain close to zero across all evaluated instances. The consistently small standard deviations indicate stable behavior over the evaluated dataset and repeated session keys, although the limited dataset does not support claims of modality-independent generalization.

**Table 3.** Statistical confidentiality metrics over 30 chest radiographs and five independent sessions.

| Metric | Proposed (Mean ± Std.) | Ideal |
|---|---|---|
| Plaintext entropy | 7.3631 ± 0.4699 | — |
| Ciphertext entropy | 7.9972 ± 0.0002 | 8.0000 |
| Horizontal correlation | 0.00008 ± 0.00388 | 0 |
| Vertical correlation | 0.00002 ± 0.00400 | 0 |
| Diagonal correlation | −0.00015 ± 0.00426 | 0 |

Taken together, the visual observations and statistical measurements consistently indicate that the proposed framework removes the spatial redundancy of the evaluated images while maintaining stable confidentiality characteristics across independent encryption sessions. These results support the effectiveness of the confidentiality component under the evaluated experimental conditions while remaining consistent with the previously discussed conditional security claims.

### 4.3. Differential Sensitivity

Plaintext differential sensitivity was evaluated using the standard NPCR and UACI metrics [31]. Table 4 compares the proposed framework with the theoretical reference values and the confidentiality-only baseline reported in [5]. Plaintext-coupled chaotic seeding substantially improves differential behavior. The average NPCR increases from 16.7251% in the baseline to 99.6098%, while the average UACI increases from 6.3289% to 33.4694%. Both values closely match the theoretical expectations for statistically independent 8-bit ciphertexts [16]. These improvements are specifically attributable to the proposed plaintext-coupled seed generation, which restores global differential sensitivity while preserving independent block processing.

**Table 4.** Differential sensitivity metrics compared with the theoretical references and the confidentiality-only baseline.

| Metric | Proposed (Mean ± Std.) | Ideal | Baseline [5] |
|---|---|---|---|
| NPCR (%) | 99.6098 ± 0.0078 | 99.6094 | 16.7251 |
| UACI (%) | 33.4694 ± 0.0401 | 33.4635 | 6.3289 |

Figure 5 provides a visual comparison between the proposed framework, the theoretical reference values, and the confidentiality-only baseline. The proposed results closely follow the theoretical expectations, whereas the baseline remains substantially below the desired differential behavior. These observations support the effectiveness of plaintext-coupled chaotic seeding in recovering global plaintext sensitivity.

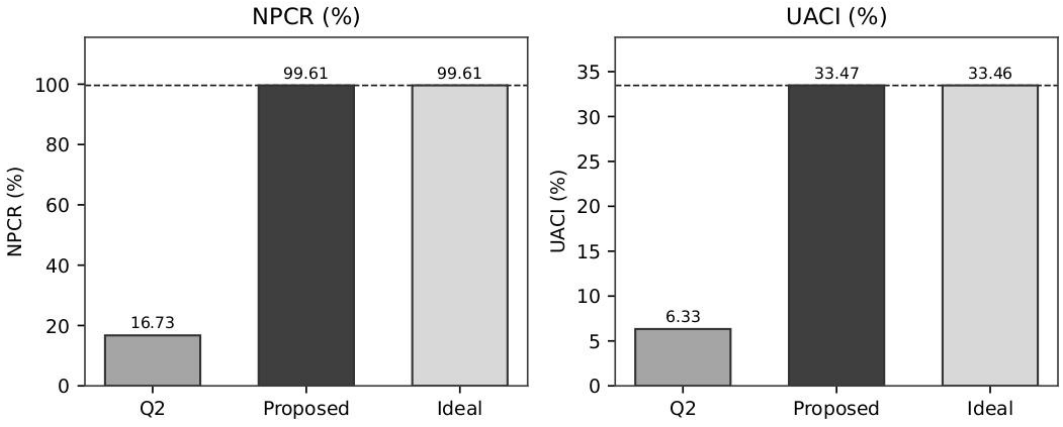



**Figure 5.** Comparison of NPCR and UACI with the theoretical references and the confidentiality-only baseline.

### 4.4. Key Sensitivity

The sensitivity of the proposed framework to cryptographic key changes was evaluated by modifying a single bit of the negotiated session key. Since the modification propagates through HKDF, all four derived subkeys change simultaneously, resulting in independent permutation, DNA, diffusion, and authentication streams. The resulting ciphertext pair achieves an NPCR of 99.61% and a UACI of 33.50%. Decryption using the one-bit-different key produces a noise-like image with a ciphertext entropy of 7.997 and adjacent-pixel correlations remaining close to zero.

Figure 6 illustrates the corresponding visual behavior. A one-bit key modification produces substantially different ciphertexts, while decryption with the incorrect key fails to recover meaningful image content. These observations demonstrate strong empirical key sensitivity throughout the complete encryption pipeline. As with the statistical confidentiality metrics, this experiment illustrates output divergence under key perturbation but is not interpreted as formal evidence of resistance to cryptographic key-recovery attacks.

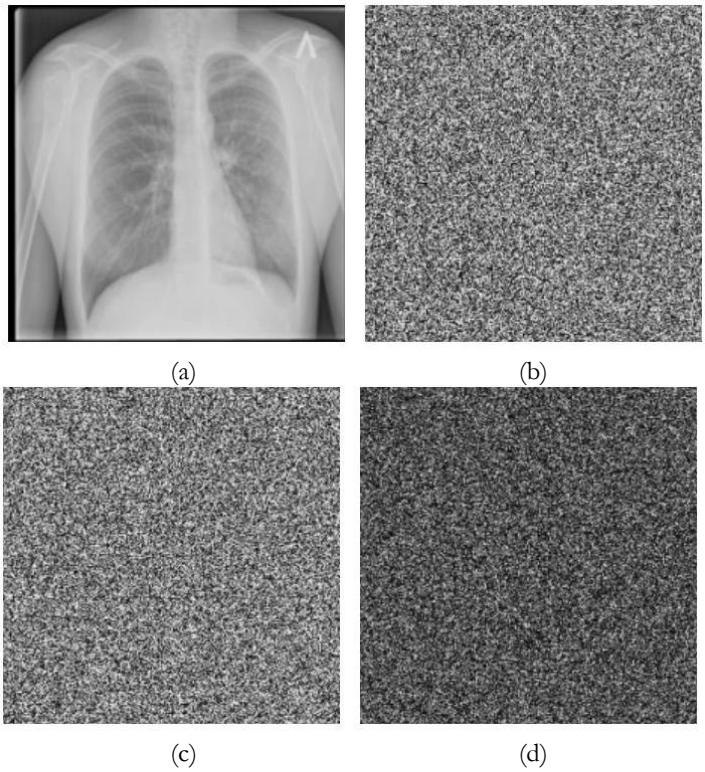
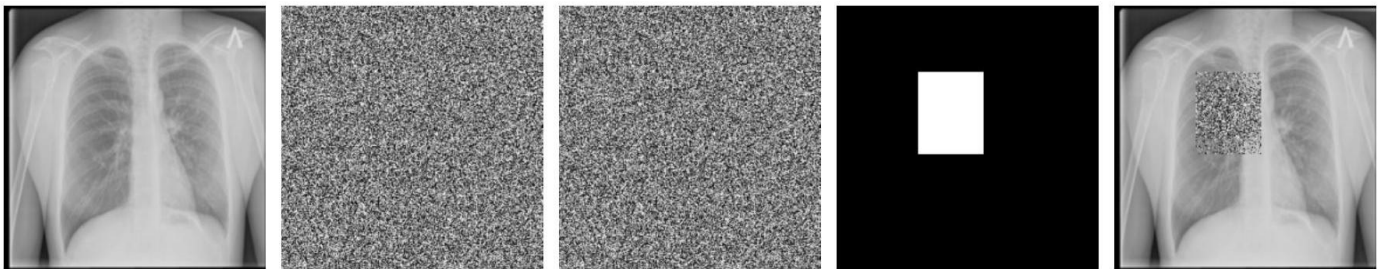



**Figure 6.** Key sensitivity under a one-bit modification of the negotiated session key: (**a**) plaintext; (**b**) ciphertext; (**c**) decrypt with wrong key; (**d**) $|C(K) - C(K')|$.

### 4.5. Integrity and Tamper Localization

The proposed authentication layer was evaluated under five representative ciphertext-tampering scenarios. These include region replacement, scattered bit flips, copy-move attacks, salt-and-pepper corruption, and single-bit ciphertext modification. Figure 7 presents a representative region-replacement attack. The corresponding authentication tags correctly identify the affected blocks, while the resulting corruption remains confined to the modified region because the encryption architecture does not employ inter-block ciphertext chaining. Under the normal receiver policy, authentication is always performed before decryption, and any failed verification causes the entire image to be rejected. The corrupted decrypted image shown in Figure 7 is therefore included solely for forensic visualization of error confinement and does not represent content that would be accepted or released by the system. The quantitative localization results are summarized in Table 5. The evaluation includes 30 images with ten independent trials for each attack type.

Every modified block was detected, while no unchanged block was incorrectly flagged during the finite evaluation, resulting in an observed 100% true-positive rate, 0% false-positive rate, perfect block-level IoU, and 100% image-level detection. These experimental observations are consistent with the expected behavior of a cryptographic MAC. Nevertheless, they should not be interpreted as mathematically deterministic because a modified block may still be accepted with a probability bounded by the HMAC forgery advantage together with the $2^{-128}$ random-guess probability established in Section 3.6. The relatively low pixel-level IoU observed for sparse attacks reflects the block-based localization granularity rather than incorrect detection.



(a) (b) (c) (d) (e)

**Figure 7.** Representative block-level tamper localization under a region-replacement attack: (**a**) plaintext; (**b**) ciphertext; (**c**) tempered ciphertext; (**d**) localization; (**e**) decrypted (tampered).

**Table 5.** Observed integrity verification and block-level tamper localization results.

| Attack | TPR | FPR | Block IoU | Pixel IoU | Image Detection |
|---|---|---|---|---|---|
| Region replacement | 100% | 0% | 1.000 | 0.650 | 100% |
| Scattered bit flip | 100% | 0% | 1.000 | 0.004 | 100% |
| Copy-move | 100% | 0% | 1.000 | 0.586 | 100% |
| Salt-and-pepper | 100% | 0% | 1.000 | 0.011 | 100% |
| Single-bit modification | 100% | 0% | 1.000 | 0.004 | 100% |

### 4.6. Block-Size Ablation

The block size directly influences the trade-off among localization granularity, authentication storage overhead, computational latency, and the conservative whole-image forgery bound. Smaller blocks provide finer spatial localization at the expense of increased authentication metadata and per-block processing, whereas larger blocks reduce storage and computational cost while coarsening localization resolution.

Table 6 summarizes the effect of varying the block size on localization performance and tag storage. Reducing the block size from B=64 to B=8 increases the region-level pixel IoU from 0.253 to 0.826, demonstrating substantially improved localization accuracy. This improvement is accompanied by an increase in authentication-tag storage from 0.25 KiB to 16 KiB. For sparse attacks, the pixel-level IoU remains low because localization is intentionally performed at block granularity, although the correct tampered block continues to be identified.

**Table 6.** Effect of block size on localization accuracy and authentication-tag storage.

| Block Size (B) | Number of Blocks (N) | Tag Storage (KiB) | Storage Overhead | Pixel IoU (Region Replacement) | Pixel IoU (Bit Flip) |
|---|---|---|---|---|---|
| 8 | 1024 | 16.00 | 25.00% | 0.826 | 0.016 |
| 16 | 256 | 4.00 | 6.25% | 0.664 | 0.004 |
| 32 | 64 | 1.00 | 1.56% | 0.416 | 0.001 |
| 64 | 16 | 0.25 | 0.39% | 0.253 | 0.001 |

The corresponding authentication performance is presented in Table 7. Smaller blocks require a larger number of independent HMAC evaluations together with additional Python framing operations, leading to higher authentication latency. Nevertheless, the authenticated

byte volume remains linear with the image size because only the number of authenticated blocks changes.

**Table 7.** Effect of block size on authentication performance and conservative forgery bound.

| Block Size (B) | HMAC Time (ms) | Throughput (MB/s) | Streaming Block + Tag | Conservative Forgery Bound |
|---|---|---|---|---|
| 8 | 7.957 ± 0.718 | 8.24 | 80 B | $2^{-118}$ |
| 16 | 2.847 ± 0.258 | 23.02 | 272 B | $2^{-120}$ |
| 32 | 1.261 ± 0.093 | 51.95 | 1040 B | $2^{-122}$ |
| 64 | 0.686 ± 0.079 | 95.49 | 4112 B | $2^{-124}$ |

Figure 8 visualizes the relationship between localization accuracy and authentication overhead. As the block size decreases, localization becomes increasingly precise, while storage overhead and per-block authentication cost increase accordingly. Based on this trade-off, the default configuration of B=16 provides a practical balance among localization accuracy, storage efficiency, authentication latency, and the conservative forgery bound.

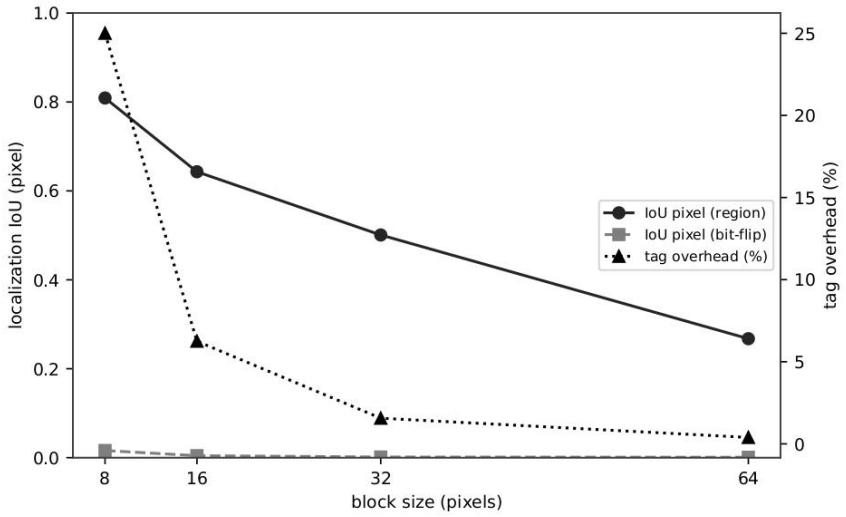



**Figure 8.** Trade-off between localization accuracy and authentication overhead for different block sizes.

### 4.7. Component-Level Ablation

To better understand the contribution of each design component, an ablation study was performed by selectively removing or modifying individual elements of the proposed framework. The evaluated variants include measured baselines together with analytical and structural modifications that cannot always be represented through image-security statistics alone. The results are summarized in Table 8. Removing plaintext-coupled seeding substantially reduces NPCR and UACI because this component is responsible for restoring global plaintext sensitivity. Replacing block-level authentication with a single global HMAC preserves the confidentiality metrics but eliminates block-level tamper localization. Eliminating HMAC removes the ability to detect active ciphertext modification entirely. Likewise, collapsing the HKDF labels does not necessarily affect statistical image-security metrics, yet invalidates the cryptographic key-separation assumption required by the authenticated composition. Finally, the DNA stage belongs to the inherited confidentiality mechanism rather than the proposed authentication contribution; consequently, removing it requires a separate confidentiality evaluation and is not considered evidence for or against the authentication design.

The ablation study demonstrates that the observed improvements originate from distinct design choices rather than from a single mechanism. Plaintext-coupled seeding primarily contributes to differential sensitivity, whereas block-level HMAC enables integrity verification and tamper localization. HKDF-based key separation supports the reduction-based security argument without necessarily producing a measurable statistical signature, illustrating that

some security properties are established analytically rather than empirically. Together, these observations reinforce that the proposed framework combines complementary mechanisms whose contributions are independently justified by measurement, structural analysis, or cryptographic reasoning.

**Table 8.** Component-level ablation and interpretation of the proposed framework.

| Variant / Component | Differential Behavior | Integrity / Localization | Interpretation and Evidence |
|---|---|---|---|
| Full framework | NPCR = 99.6098%; UACI = 33.4694% | Yes / Yes | Measured reference implementation. |
| No plaintext coupling [5] | NPCR = 16.7251%; UACI = 6.3289% | Can remain Yes / Yes | Measured baseline demonstrating the contribution of global plaintext coupling. |
| Single global HMAC tag | Unchanged in principle | Yes / No | Analytical result. One authentication decision cannot identify the modified block, while storage overhead decreases to 0.0244%. |
| No HMAC | Unchanged in principle | No / No | Structural analysis. Statistical encryption metrics cannot detect active ciphertext modification. |
| No position-preserved blocks | Potentially stronger inter-block propagation | Yes / No block localization | Structural analysis. Removing block positions eliminates meaningful localization coordinates. |
| Collapsed HKDF labels | No expected statistical signature | Security assumption weakened | Structural analysis. Independent encryption and authentication keys are no longer guaranteed. |
| No DNA operation | Not re-evaluated | Authentication unchanged | The DNA stage belongs to the inherited confidentiality component rather than the proposed authentication contribution. |

### 4.8. Runtime, Memory, and AEAD Comparison

The runtime evaluation focuses on the incremental overhead introduced by the proposed authentication layer. Under the default configuration (B=16), authentication requires an average processing time of 2.847 ms together with 4 KiB of tag storage for a 256×256 grayscale image. Table 9 compares the proposed protection layer with global and block-wise implementations of HMAC-SHA256-128 and ChaCha20-Poly1305 using the same payload.

**Table 9.** Reference Python microbenchmark of protection-layer implementations for 256 × 256 grayscale images.

| Protection Layer | Number of Tags | Time (ms) | Wire Overhead | Block-Level Localization |
|---|---|---|---|---|
| Global HMAC-SHA256-128 | 1 | 0.266 ± 0.033 | 0.0244% | No |
| Block HMAC-SHA256-128 ((B=16)) | 256 | 2.847 ± 0.258 | 6.25% | Yes |
| Global ChaCha20-Poly1305 | 1 | 0.075 ± 0.011 | 0.0244% | No |
| Block ChaCha20-Poly1305 ((B=16)) | 256 | 2.232 ± 0.281 | 6.25% | Yes |

The global protection mechanisms achieve the lowest execution time and communication overhead because authentication is performed only once for the entire image. However, they cannot identify the location of a modification. In contrast, the block-wise approaches provide explicit spatial localization at the cost of additional authentication metadata and repeated per-block cryptographic operations.

The reported block-wise timings include Python loop overhead, block extraction, message framing, and repeated cryptographic library invocations, whereas the global implementations invoke optimized native primitives only once. Consequently, the comparison quantifies the computational cost associated with block-level localization rather than implying that HMAC is inherently slower than ChaCha20-Poly1305.

If the confidentiality mechanism is replaced with a standard AEAD algorithm, block-wise ChaCha20-Poly1305 provides a practical authenticated-encryption alternative with equivalent localization capability. In contrast, the proposed HMAC-based authentication layer is specifically intended for scenarios in which the existing DNA-chaos encryption engine is preserved while adding independently keyed authentication. Under the default configuration, the total execution time of the proposed framework is $T_{\mathrm{DNA-chaos}} + 2.847\mathrm{ms}$, plus approximately 0.212 ms for X25519 key establishment and HKDF expansion whenever a new session is negotiated.

### 4.9. Transmission Errors and Packet Loss

The proposed authentication layer follows a fail-closed design rather than an error-tolerant one. Under the default configuration (B=16), each authenticated block record consists of 2048 ciphertext bits together with a 128-bit authentication tag, yielding a total protected record length of 2176 bits. Assuming an independent channel bit-error rate of p, the probability that at least one protected bit is corrupted is $1 - (1 - p)^{2176}$, and the corresponding block is rejected with overwhelming probability. Likewise, if either the ciphertext block or its authentication tag is missing, the header-declared block grid immediately identifies the missing block coordinate. These results are derived analytically and therefore complement, rather than replace, the image-encryption experiments presented in the previous subsections. Table 10 summarizes the analytical behavior of the proposed framework under representative channel impairments.

**Table 10.** Analytical fail-closed behavior under independent transmission impairments (B=16, N=256).

| Channel Impairment | Error / Loss Rate | Analytical Outcome | Operational Implication |
|---|---|---|---|
| Random bit error | $10^{-6}$ per bit | 0.217% of block records rejected | Retransmit the affected block record. |
| Random bit error | $10^{-5}$ per bit | 2.153% rejected | Authentication detects corruption but cannot determine whether it is accidental or malicious. |
| Random bit error | $10^{-4}$ per bit | 19.556% rejected | Reliable transport or forward-error correction becomes necessary. |
| Random bit error | $10^{-3}$ per bit | 88.663% rejected | The communication channel becomes impractical without retransmission or error correction. |
| Block-packet loss | 0.01% | 97.472% probability that all 256 records arrive | Expected missing blocks: 0.0256. |
| Block-packet loss | 0.1% | 77.404% probability that all records arrive | Expected missing blocks: 0.256. |
| Block-packet loss | 1% | 7.631% probability that all records arrive | Expected missing blocks: 2.56; retransmission or FEC is required. |

In reliable transport protocols such as TCP, TLS, and QUIC, ordinary retransmission mechanisms generally recover packet loss before the proposed image-authentication layer is executed. In contrast, UDP-based or disruption-tolerant communication requires each block to be transmitted together with its index and authentication tag, while retransmission or forward-error-correction mechanisms should be employed to improve delivery reliability.

The proposed framework is intentionally designed to detect and localize corrupted or missing block records rather than recover them. Treating authentication failures as harmless transmission noise would also permit malicious ciphertext modification to pass undetected. Therefore, the fail-closed behavior represents a deliberate security design choice rather than a limitation of the authentication mechanism.

## 5. Discussion

The principal contribution of the proposed framework is architectural rather than the introduction of new cryptographic primitives. X25519, HKDF, DNA encoding, chaotic

maps, and HMAC are all established techniques; the novelty lies in their composition to satisfy two security objectives that are typically considered conflicting. Specifically, a pre-encryption plaintext modification propagates through the plaintext hash $H$ and changes every block seed, producing strong global differential behavior. In contrast, a post-encryption channel modification leaves $H$ unchanged, affects only the corresponding position-preserved ciphertext block, and is detected before decryption through its block authentication tag. This separation provides a systematic way to reconcile avalanche behavior with spatial tamper localization within a single framework.

To clarify the positioning of the proposed framework, Table 11 qualitatively compares it with conventional DNA-based image encryption, standardized authenticated encryption, and fragile watermarking. The comparison emphasizes the protection model, localization capability, principal advantage, and primary trade-off of each approach without introducing additional experimental claims.

**Table 11.** Qualitative comparison of the proposed framework with representative image-protection approaches.

| Approach | Protection and Verification Model | Tamper Localization | Principal Advantage | Primary Trade-off |
|---|---|---|---|---|
| Conventional DNA-based image encryption [1]–[4], [16], [17]. | DNA encoding combined with chaotic permutation and diffusion primarily provides confidentiality; cryptographic integrity is not inherent. | No | Preserves image-oriented confusion, diffusion, and plaintext sensitivity. | Modified ciphertext and protocol metadata cannot be authenticated or spatially localized reliably. |
| Authenticated encryption (e.g., ChaCha20-Poly1305) [9], [10]. | Standardized authenticated encryption over ciphertext and associated data. | Global by default; block-level localization requires separate records. | Mature, efficient, and formally analyzed confidentiality and integrity. | Block-wise deployment requires additional framing, nonce management, and associated-data binding, while replacing the DNA-chaos confidentiality core. |
| Fragile watermarking [11], [12] | Authentication evidence is embedded in or derived from the host image. | Yes | Supports spatial tamper detection and, in some schemes, content recovery. | Operates primarily in the plaintext domain, may modify the host image, and does not inherently provide session-keyed ciphertext confidentiality. |

As summarized in Table 11, the proposed framework is not intended to replace standardized AEAD constructions or to suggest that they are generally inferior. Standard AEAD remains the preferred choice for general-purpose authenticated encryption. Instead, the proposed design addresses a different problem: preserving an existing DNA-chaos image-encryption pipeline while introducing session-bound, pre-decryption integrity verification with block-level localization. In this context, the framework complements rather than competes with conventional authenticated-encryption approaches.

The experimental results further illustrate that the design parameters provide configurable operational trade-offs. Smaller authentication blocks improve localization granularity at the cost of higher storage and computational overhead, whereas larger blocks favor efficiency while reducing spatial resolution. Likewise, the selected 128-bit authentication tag aligns with the classical security target of X25519 while avoiding unnecessary storage overhead associated with longer tags. These parameters should therefore be regarded as deployment choices rather than universally optimal settings.

The communication analysis also clarifies the intended robustness of the proposed authentication layer. Corrupted or missing records are detected and localized but are intentionally not corrected. Consequently, accidental transmission errors and malicious modifications produce the same authentication outcome, reflecting a deliberate fail-closed security policy. Practical deployment therefore depends on complementary protocol services, including authenticated peer credentials, replay protection, reliable transport or forward-error correction, constant-time tag verification, strict parsing of protocol fields, and rejection of any image that fails authentication.

Several limitations should be acknowledged. First, the confidentiality guarantee remains conditional because no formal IND-CPA proof is provided for the floating-point DNA-chaos encryption core; statistical image-security metrics cannot substitute for such a proof.

Second, including the plaintext hash $H$ in the authenticated header permits an offline confirmation test for guessed low-entropy images. Deployments concerned with this information leakage should replace $H$ with a key-protected commitment or protect the seed material under a separate derived key before reevaluating the differential behavior. Third, peer authentication for X25519 together with replay-state management remains an external protocol requirement. Fourth, the image evaluation is limited to 30 resized grayscale tuberculosis radiographs, and broader validation using color images, native resolutions, multiple imaging modalities, and larger datasets is needed before generalizing the statistical observations. Fifth, the runtime evaluation isolates the incremental authentication layer on a single Python/CPU platform and does not represent an optimized production implementation. Finally, the framework localizes corrupted or missing blocks but does not recover them. Future work will focus on strengthening the confidentiality foundation by formally analyzing or replacing the floating-point DNA-chaos core, introducing a protected plaintext commitment, publishing interoperable implementation test vectors, evaluating larger and more diverse image datasets, and integrating retransmission or erasure-coding mechanisms for communication over unreliable channels.

## 6. Conclusions

This paper presented a context-bound, block-wise authenticated DNA-based image-encryption framework that integrates confidentiality, pre-decryption integrity, and spatial tamper evidence. Authenticated ephemeral X25519 and HKDF-SHA256 establish fresh domain-separated keys, plaintext-coupled block-specific chaotic seeds recover global differential sensitivity, and 128-bit HMAC-SHA256 tags bind each ciphertext block to its header, position, dimensions, length, session, and sequence number. The experimental results demonstrate that the proposed framework achieves its intended objectives. The tested radiographs exhibit high statistical confidentiality and near-ideal differential sensitivity, while the authentication layer provides reliable pre-decryption integrity verification and block-level tamper localization. The reduction-based argument further establishes ciphertext integrity and a conditional IND-CCA upgrade under explicit assumptions, while acknowledging that the formal confidentiality of the DNA-chaos core remains an open problem. Overall, the proposed framework shows that global plaintext sensitivity and localized post-encryption tamper evidence can coexist within a unified authenticated image-encryption architecture. The resulting design provides a practical foundation for applications that require both encrypted image transmission and actionable integrity verification.

**Author Contributions:** Conceptualization: B.S.W.P. and K.A.; Methodology: B.S.W.P.; Software: B.S.W.P.; Validation: B.S.W.P., K.A., and A.P.W.; Formal analysis: B.S.W.P.; Investigation: B.S.W.P.; Resources: B.S.W.P.; Data curation: B.S.W.P.; Writing—original draft preparation: B.S.W.P.; Writing—review and editing: K.A. and A.P.W.; Visualization: B.S.W.P.; Supervision: K.A. and A.P.W.; Project administration: K.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The Tuberculosis (TB) Chest X-ray Database is publicly available through IEEE DataPort, doi: 10.21227/mps8-kb56 [29]. The exact 30-image subset identifiers and implementation are available from the corresponding author on reasonable request.

**Acknowledgments:** The authors acknowledge the public availability of the Tuberculosis (TB) Chest X-ray Database used in the experimental evaluation. The authors also acknowledge the use of AI to assist with English grammar refinement and language editing. The authors reviewed and edited the generated text as necessary and take full responsibility for the content of this manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest

## References


[1] H. Zhang, X. Feng, J. Sun, and P. Yan, "Chaotic Image Security Techniques and Developments: A Review," *Mathematics*, vol. 13, no. 12, p. 1976, Jun. 2025, doi: 10.3390/math13121976.

[2] L. Huang, C. Ding, Z. Bao, H. Chen, and C. Wan, "A DNA Encoding Image Encryption Algorithm Based on Chaos," *Mathematics*, vol. 13, no. 8, p. 1330, Apr. 2025, doi: 10.3390/math13081330.

[3] C. Bhaya, M. Zain, and A. K. Singh, "A DNA-based color image cryptosystem using chaotic maps, spiral mixing and non-linear binary operator," *Sci. Rep.*, vol. 15, no. 1, p. 33813, Sep. 2025, doi: 10.1038/s41598-025-04021-4.

[4] M. Samiullah *et al.*, "An Image Encryption Scheme Based on DNA Computing and Multiple Chaotic Systems," *IEEE Access*, vol. 8, pp. 25650–25663, 2020, doi: 10.1109/ACCESS.2020.2970981.

[5] B. S. W. Poetro, K. Adi, and A. P. Widodo, "System-Level Secure Key Management For DNA-Based Image Cryptography Using Ephemeral ECDH and HKDF-SHA256," *Eng. Technol. Appl. Sci. Res.*, vol. 16, no. 3, pp. 36418–36426, Jun. 2026, doi: 10.48084/etasr.17609.

[6] B. S. W. Poetro, K. Adi, and A. P. Widodo, "Autonomous Key Generation and Management using HKDF-SHA256 for Secure DNA-based Image Cryptography," in *2025 3rd International Conference on Computer System, Information Technology, and Electrical Engineering (COSITE)*, Dec. 2025, pp. 381–384. doi: 10.1109/COSITE68330.2025.11414297.

[7] H. Krawczyk and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)," May 2010. doi: 10.17487/rfc5869.

[8] A. Langley, M. Hamburg, and S. Turner, "Elliptic Curves for Security," Jan. 2016. doi: 10.17487/RFC7748.

[9] R. Serrano, C. Duran, M. Sarmiento, C.-K. Pham, and T.-T. Hoang, "ChaCha20–Poly1305 Authenticated Encryption with Additional Data for Transport Layer Security 1.3," *Cryptography*, vol. 6, no. 2, p. 30, Jun. 2022, doi: 10.3390/cryptography6020030.

[10] Y. Nir and A. Langley, "ChaCha20 and Poly1305 for IETF Protocols," Jun. 2018. doi: 10.17487/RFC8439.

[11] P. W. Adi, A. Sugiharto, M. M. Hakim, D. R. I. M. Setiadi, and E. Winarno, "Efficient fragile watermarking for image tampering detection using adaptive matrix on chaotic sequencing," *Intell. Syst. with Appl.*, vol. 26, p. 200530, Jun. 2025, doi: 10.1016/j.iswa.2025.200530.

[12] T. Mendis, F. Kandah, and L. Medury, "MATADOR: a magic matrix-based framework for tamper detection and image recovery," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 37, no. 5, p. 77, Jul. 2025, doi: 10.1007/s44443-025-00099-y.

[13] B. Jasra and A. Hassan Moon, "Color image encryption and authentication using dynamic DNA encoding and hyper chaotic system," *Expert Syst. Appl.*, vol. 206, no. December 2021, p. 117861, Nov. 2022, doi: 10.1016/j.eswa.2022.117861.

[14] H. Zhiqiang, A. Rauf, A. Nazir, F. Tchier, A. Aslam, and K. A. Tola, "Design and analysis of a secure image encryption algorithm using proposed non-linear RN chaotic system and ECC/HKDF key derivation with authentication support," *Sci. Rep.*, vol. 15, no. 1, p. 39951, Nov. 2025, doi: 10.1038/s41598-025-23592-w.

[15] R. Bhargav and P. Singh, "A structure based block-wise image authentication framework for secure transmission and tamper localisation," *Results Eng.*, vol. 31, p. 111483, Sep. 2026, doi: 10.1016/j.rineng.2026.111483.

[16] V. N. S. Kumaran, T. Manikandan, R. K. Dhanaraj, T. Al-Shehari, N. A. Alsadhan, and S. Selvarajan, "A secure medical image encryption technique based on DNA cryptography with elliptic curves," *Sci. Rep.*, vol. 15, no. 1, p. 20003, Jun. 2025, doi: 10.1038/s41598-025-03898-5.

[17] A. Hennache, M. L. Hennache, and S. M. A. Ghaly, "Improving the RSA Encryption for Images by Introducing DNA Sequence Encoding," *Eng. Technol. Appl. Sci. Res.*, vol. 14, no. 6, pp. 17786–17791, Dec. 2024, doi: 10.48084/etasr.8557.

[18] E. Winarno, K. Nugroho, P. W. Adi, and D. R. I. M. Setiadi, "Combined Interleaved Pattern to Improve Confusion-Diffusion Image Encryption Based on Hyperchaotic System," *IEEE Access*, vol. 11, pp. 69005–69021, 2023, doi: 10.1109/ACCESS.2023.3285481.

[19] E. Winarno, W. Hadikurniawati, K. Nugroho, and V. Lusiana, "Integrating Quadratic Polynomial and Symbolic Chaotic Map-Based Feistel Network to Improve Image Encryption Performance," *IEEE Access*, vol. 12, pp. 106720–106734, 2024, doi: 10.1109/ACCESS.2024.3436558.

[20] E. Winarno, K. Nugroho, P. W. Adi, and D. R. I. M. Setiadi, "Integrated dual hyperchaotic and Josephus traversing based 3D confusion-diffusion pattern for image encryption," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 35, no. 9, p. 101790, Oct. 2023, doi: 10.1016/j.jksuci.2023.101790.

[21] Y. Sanjalawe, A. Al-Daraiseh, S. Al-E'mari, and S. N. Makhadmeh, "FileCipher: A Chaos-Enhanced CPRNG-Based Algorithm for Parallel File Encryption," *Algorithms*, vol. 19, no. 2, p. 119, Feb. 2026, doi: 10.3390/a19020119.

[22] C. W. Chuah, N. Z. Harun, and I. R. A. Hamid, "Key derivation function: key-hash based computational extractor and stream based pseudorandom expander," *PeerJ Comput. Sci.*, vol. 10, p. e2249, Aug. 2024, doi: 10.7717/peerj-cs.2249.

[23] A. Saini and R. Sehrawat, "Enhancing Data Security through Machine Learning-based Key Generation and Encryption," *Eng. Technol. Appl. Sci. Res.*, vol. 14, no. 3, pp. 14148–14154, Jun. 2024, doi: 10.48084/etasr.7181.

[24] M. Bellare and C. Namprempre, "Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm," *J. Cryptol.*, vol. 21, no. 4, pp. 469–491, Oct. 2008, doi: 10.1007/s00145-008-9026-x.

[25] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication," Feb. 1997. doi: 10.17487/rfc2104.

[26] V. Tanksale, "Efficient Elliptic Curve Diffie–Hellman Key Exchange for Resource-Constrained IoT Devices," *Electronics*, vol. 13, no. 18, p. 3631, Sep. 2024, doi: 10.3390/electronics13183631.

[27] Q. H. Dang, "Recommendation for applications using approved hash algorithms," Gaithersburg, MD, 2012. doi: 10.6028/NIST.SP.800-107r1.

[28] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. 29, no. 2, pp. 198–208, Mar. 1983, doi: 10.1109/TIT.1983.1056650.

[29] T. Rahman, A. Khandakar, and M. E. H. Chowdhury, "Tuberculosis (TB) Chest X-ray Database." IEEE DataPort, 2020. doi: 10.21227/mps8-kb56.

[30] T. Rahman *et al.*, "Reliable Tuberculosis Detection Using Chest X-Ray With Deep Learning, Segmentation and Visualization," *IEEE Access*, vol. 8, pp. 191586–191601, 2020, doi: 10.1109/ACCESS.2020.3031384.

[31] Y. Wu, J. P. Noonan, and S. Agaian, "NPCR and UACI Randomness Tests for Image Encryption," *Cyber Journals Multidiscip. Journals Sci. Technol. J. Sel. Areas Telecommun.* , pp. 31–38, 2011, Accessed: Sep. 27, 2019. [Online]. Available: https://pdfs.semanticscholar.org/2b47/9abce221135af6065f9f8352e09cbfb5733a.pdf