APPENDIX

A THEORETICAL FOUNDATIONS OF THE CRYPTO-NCRNA FRAMEWORK

A.1 RNA-INTRINSIC PHYSICAL UNCLONABLE FUNCTIONS

Crypto-ncRNA leverages ncRNA (non-coding RNA) molecules as Physical Unclonable Functions (PUFs) to provide robust resistance against cloning and physical attacks. Unlike purely digital cryptographic systems, RNA-based PUFs introduce molecular-level variability that is prohibitively difficult for adversaries to replicate. Figure 3 shows the entire workflow.

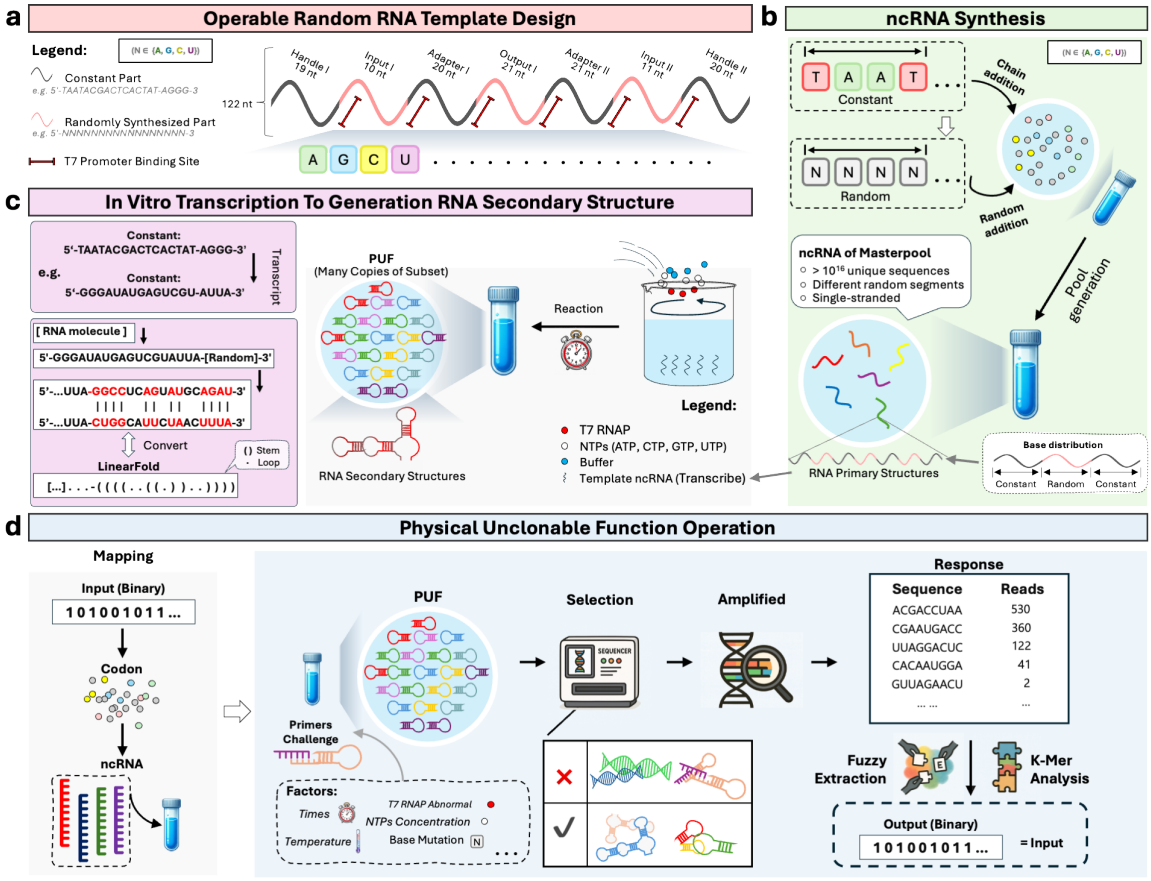

Figure 3: ncRNA-based Physical Unclonable Function (PUF) Workflow

(a) **Operable Random RNA Template Design (Fig. 3a):** The RNA template consists of both constant (black) and random (pink) regions. A T7 promoter binding site (dark red) enables *in vitro* transcription by T7 RNA polymerase. The random segments ($N = \mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{U}$) are strategically placed to maximize sequence entropy. Constant regions, such as Handle I/II and Adapter I/II, ensure compatibility with downstream operations (PCR amplification or sequencing).

(b) **ncRNA Synthesis (Fig. 3b):** Single-stranded RNA molecules are synthesized from a mixed pool of bases (A, C, G, U) under controlled conditions. In the "Chain addition" step, the constant regions are added in a predetermined fashion, while the "Random addition" step introduces high-entropy segments. This process yields an ncRNA "master pool" containing up to 10^{16} unique sequences, each with distinct primary structures that can later fold into diverse secondary conformations.

(c) **In Vitro Transcription to Generate RNA Secondary Structure (Fig. 3c):** Using the T7 promoter sequence embedded in the DNA template, T7 RNA polymerase (T7 RNAP) transcribes the random RNA segment in a reaction mixture containing nucleoside triphosphates (NTPs). The resulting single-stranded RNA adopts unique secondary structures (stems, loops, etc.) driven by its specific sequence composition, forming the core of the Physical Unclonable Function (PUF) property. Because each random region can fold differently, even under the same conditions, each molecule exhibits a distinct structural "fingerprint," making it extremely difficult to replicate or clone. This panel illustrates an example transcript (with constant regions in black and random regions in red) and underscores how structural variability at the RNA level underpins the system's robustness.

(d) **Physical Unclonable Function Operation (Fig. 3d):** The ncRNA pool undergoes a "challenge–response" procedure. An input is mapped to codons or primer sequences ("Challenge"), which selectively bind to or amplify specific ncRNA molecules. Environmental and biochemical factors (e.g., enzyme activity, temperature) further diversify the ncRNA conformations. The resulting sequences ("Response") are processed by fuzzy extraction (e.g., k-mer analysis) to produce a robust, high-entropy output. Because each ncRNA molecule's structure is physically unique and challenging to clone, the system achieves strong security as a chemical unclonable function.

By incorporating high-entropy random regions into ncRNA synthesis, each molecule adopts a unique conformation, thereby harnessing physical unclonability to achieve molecular-level cryptographic security.

A.2 Verification and Key Generation Process using RNA-based PUFs

Building on the ncRNA-based PUF framework described in A1, this section demonstrates how the same (or a similarly constructed) RNA pool can be used to verify the PUF's reliability and generate secure cryptographic keys. Here, we systematically introduce various challenge inputs, observe the corresponding responses from the ncRNA pool, and apply fuzzy extraction and PBKDF2 key derivation methods to produce a final cryptographic key.

Figure 4 provides a comprehensive view of these verification and key generation stages. It showcases the similarity analysis of challenge–response pairs, tests across multiple "proliferation" generations of RNA PUF samples, quantifies the Hamming distances (to evaluate randomness and uniqueness), and illustrates how a fuzzy extractor and PBKDF2 combine to output secure keys.

(a) **CPR Similarity (Fig. 4a):** This panel shows a similarity matrix of Challenge–Response (C–R) pairs across various RNA-based PUF sets. Each row corresponds to a unique challenge input (e.g., C1, C2, C3), and each column represents the corresponding response measured from a particular PUF. Darker (or lighter) squares may indicate higher similarity, demonstrating consistent responses within the same set of challenges. In contrast, more moderate or low similarity values reveal higher variability, indicating the PUF's ability to produce distinguishable outputs for different inputs.

(b) **Proliferations CPR Similarity (Fig. 4b):** We illustrate how different proliferation generations of RNA-based PUFs (denoted P0, P1, P2, etc.) respond to the same set of challenges. Proliferations represent successive rounds of RNA synthesis or replication, potentially introducing small sequence or structural variations. Despite these changes, each generation typically maintains a recognizable signature under the same challenge, reflecting the intrinsic reliability of the RNA-based PUF design over repeated use.

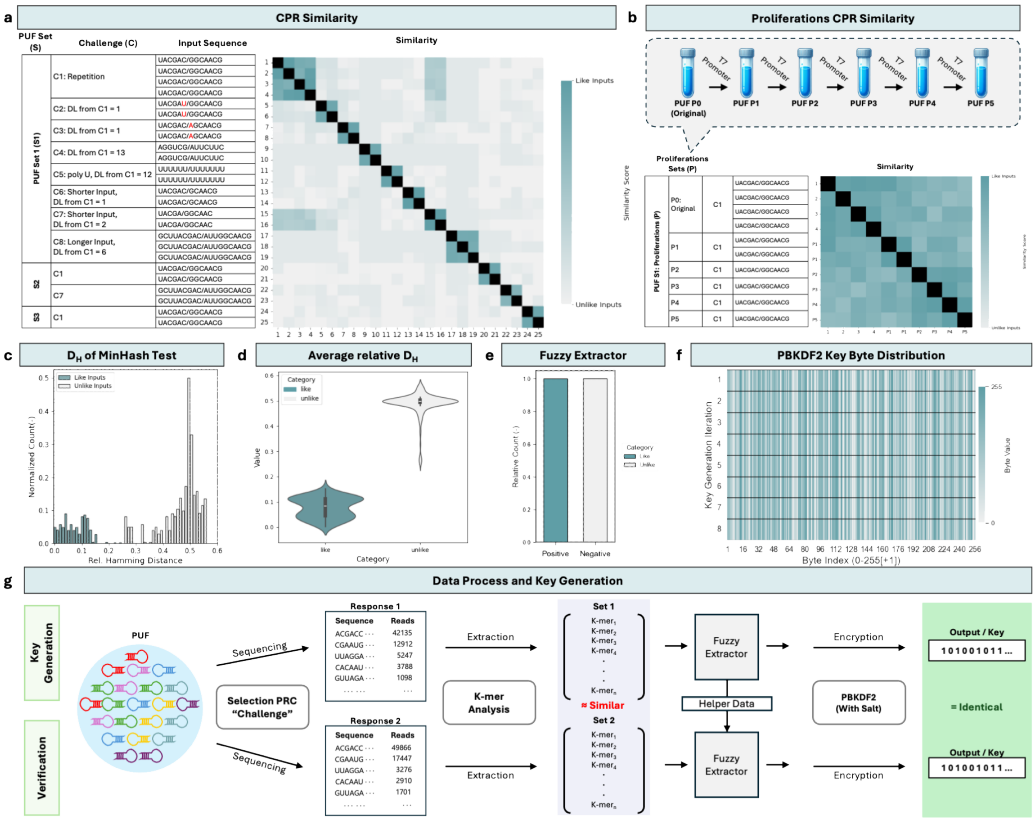

Figure 4: Verification and Key Generation Process using RNA-based PUFs Workflow

(c) **DH of MinHash Test (Fig. 4c):** In this histogram, we plot the normalized Hamming distances between pairs of C–R outputs. Responses grouped as "like" are those expected to be similar (e.g., same challenge, same RNA generation), whereas "unlike" pairs come from distinct challenges or significantly altered RNA samples. The clear peak separation between "like" and "unlike" categories underscores the high discriminative power of the ncRNA-based PUF, as well as its robustness in generating unique, non-overlapping response distributions.

(d) **Average Relative DH (Fig. 4d):** This violin plot complements the histogram by showing the distribution of average relative Hamming distances across many "like" and "unlike" C–R pairs. The distribution shape, median line, and spread for each category visually demonstrate how stable the "like" responses are compared to the broader spread among "unlike" pairs. A larger gap between these two distributions attests to the reliability and uniqueness of the RNA-based PUF responses.

(e) **Fuzzy Extractor (Fig. 4e):** A fuzzy extraction algorithm processes the raw responses, potentially subject to minor experimental or environmental noise, to produce stable cryptographic keys. This panel shows how initial noisy or partially overlapping responses are reconciled into a uniform output. By correcting small discrepancies, the fuzzy extractor ensures that repeated challenges yield the same high-entropy key, reinforcing the practical viability of ncRNA-based PUFs for secure applications.

(f) **PBKDF2 Key Byte Distribution (Fig. 4f):** This panel illustrates how PBKDF2 (Password-Based Key Derivation Function 2) transforms the fuzzy extractor's output into uniformly randomized cryptographic keys, even if the input sequences vary. The byte-value histogram (0–255) for multiple derived keys shows a near-even spread, indicating high entropy while confirming that identical or "like" challenges generate consistent final keys. Despite potential sequence or environmental differences in earlier steps, the PBKDF2 process ensures that each unique challenge-response leads to a securely distributed key, supporting post-quantum cryptographic standards by resisting brute-force and quantum-based attacks.

(g) **Data Process and Key Generation (Fig. 4g):** A high-level schematic of the end-to-end workflow, from PUF-based key generation to verification. First, the RNA PUF is challenged and sequenced, producing raw "Response 1" and "Response 2." Each set undergoes k-mer analysis to yield partial fingerprints, which the fuzzy extractor processes into a stable intermediate. Any necessary helper data is added to improve reproducibility. Finally, PBKDF2 integrates the fuzzy extractor's output (plus a salt) to derive a high-entropy key. The identical procedure is used for verification: if the same challenge is applied to the same PUF, an equivalent key is recovered, demonstrating robustness and authenticity of the ncRNA-based cryptographic mechanism.

Panels (a)–(d) confirm the system's high entropy and reproducibility under varying challenges and proliferations. Panels (e)–(f) illustrate how fuzzy extraction and PBKDF2 transform raw responses into secure, robust keys. Finally, panel (g) consolidates these steps, underscoring the viability of ncRNA-based PUFs for post-quantum cryptographic applications.

Summary

Appendix A established the theoretical underpinnings of crypto-ncRNA by demonstrating how ncRNA's intrinsic structural variability can serve as a high-entropy, physically unclonable foundation for cryptographic operations. Leveraging this molecular uniqueness, the framework transitions from random RNA template design to challenge–response workflows that reliably distinguish different inputs while ensuring repeatable outcomes for identical challenges. The validation of proliferation sets and Hamming distance metrics confirms both entropy and reproducibility, while fuzzy extraction and PBKDF2 illustrate how raw biochemical responses are converted into stable, post-quantum-resistant cryptographic keys. These theoretical insights, showcasing the synergy of biological PUFs and modern cryptographic practices, lay the groundwork for the more detailed encryption algorithm and implementation strategies presented in Appendix B.

B Technical Specifications of the crypto-ncRNA Framework

Crypto-ncRNA framework is a cryptographic algorithm designed for post-quantum cryptography, leveraging the dynamic folding properties of non-coding RNA (ncRNA). This framework addresses post-quantum cryptographic challenges by integrating the physical unclonability of biomolecular Physical Unclonable Functions (PUFs) with cryptographic principles. It constructs a highly randomized and quantum-resistant encryption system. Appendix A details the framework's five core modules: (1) data encoding and RNA sequence synthesis, (2) RNA secondary structure folding and obfuscation, (3) key derivation and cryptographic protocols, (4) Physical Unclonable Function (PUF) mechanisms, and (5) parameter scalability design. Each module is meticulously designed to optimize system performance in terms of information density, randomness, key strength, and physical security.

B.1 Data Encoding & RNA Sequence Synthesis

This module converts plaintext data into RNA sequences using Base64 encoding and RNA codon mapping, significantly enhancing information density and ensuring data format standardization.

B.1.1 Base64 Preprocessing

Process Plaintext data is converted into 6-bit indices (ranging from 0 to 63) via Base64 encoding, generating a standardized byte stream.

Technical Details

- Employs the RFC 4648 standard to ensure cross-platform compatibility.
- Input data is grouped into 3-byte blocks, with each block mapped to 4 Base64 characters; padding rules adhere to ”=” character completion.

Advantages Compared to traditional binary encoding (3 bits/character), Base64 offers a 6 bits/character encoding capacity, doubling information density.

B.1.2 Codon Mapping

Process Each 6-bit index is uniquely mapped to an RNA codon (e.g., AUG, GUA) through predefined substitution rules.

Technical Details

- Establishes a 64×1 codon lookup table to ensure one-to-one mapping.
- Codon selection adheres to biocompatibility rules, avoiding stop codons (e.g., UAA, UAG).

Advantages Leverages the quaternary (A/U/G/C) nature of RNA codons to achieve data compression and biological sequence adaptation.

B.2 RNA Secondary Structure Folding & Obfuscation

This module leverages the dynamic folding properties of RNA, combining Minimum Free Energy (MFE) prediction and dynamic codon reordering to generate highly complex RNA secondary structures, enhancing data randomness and resistance to analysis.

B.2.1 Minimum Free Energy (MFE) Prediction

Process The secondary structure of the RNA sequence is predicted using the LinearFold algorithm, with the output represented in dot-bracket notation (e.g., ”((…))”) (Huang et al. (2019); Zuker & Sankoff (1984)).

Technical Details

- Employs 5’-to-3’ dynamic programming and beam search, optimizing time complexity to O(n).
- The result partitions the sequence into stem regions (Watson-Crick base pairing, e.g., AU/GC) and loop regions (unpaired regions) (Kimsey et al. (2015); Tinoco Jr & Bustamante (1999); Yakovchuk et al. (2006)).

Function Structural partitioning provides topological constraints for subsequent dynamic obfuscation.

B.2.2 Dynamic Codon Reordering

Process Codons are reordered based on structural partitioning.

- **Stem Regions** Complementary codons are permuted under base pairing constraints (e.g., AU → UA).
- **Loop Regions** Random shuffling is performed based on an entropy-driven algorithm.

Technical Details

- Stem region permutation rules are constrained by thermodynamic stability to ensure structural integrity.
- Stem region permutation rules are constrained by thermodynamic stability to ensure structural integrity.

Security Generates 4^N combinations (where N is the number of dynamic sites), far exceeding traditional matrix obfuscation methods.

B.3 Key Derivation & Cryptographic Protocol

This module generates quantum-resistant keys through an entropy source based on RNA sequences and the PBKDF2-HMAC-SHA256 algorithm, ensuring high key strength and unpredictability.

B.3.1 Entropy Pool Construction

Inputs

- **RNA Seed** A quaternary sequence of length L, with an entropy value of $\log_2(4^L)$ bits.
- **Salt Value** A random binary string of length B bits, with an entropy value of $\log_2(2^B)$ bits.

Total Entropy Value

$$\text{Total Entropy} = \log_2(4^L \times 2^B) \text{ bits} \tag{1}$$

B.3.2 Key Generation

Process A 32-byte session key is outputted by iterating the PBKDF2-HMAC-SHA256 algorithm 100,000 times (default).

Technical Detail

- HMAC-SHA256 ensures strong binding between the key and the salt value.
- The number of iterations is dynamically adjustable, with a brute-force cracking complexity reaching $O(2^{256} \times 10^5)$ (default), surpassing AES-256 security.

B.4 Physical Unclonability Mechanisms

This module ensures hardware binding and physical irreproducibility of keys through RNA molecular engineering and multidimensional randomization techniques, preventing physical attacks and cloning. It supports dynamic parameter configuration to adapt to varying security needs, ranging from high-security to high-efficiency modes, thus meeting diverse application requirements.

B.4.1 RNA Molecular Engineering

Design Vectors are synthesized based on the T7 promoter sequence (5'-TAATACGACTCACTATAGGG-3') and embedded with restriction enzyme sites (Pribnow (1975)).

Function Restriction enzyme sites provide physically verifiable markers, preventing molecular replication.

B.4.2 Multidimensional Randomization

- **Sequence Modification:** Unique chemical fingerprints are introduced through methylation or fluorescent labeling.
- **Thermodynamic Diversity:** structural variations under temperature gradients are simulated via RNAfold, ensuring that folding pathways are irreproducible (Sato et al. (2021)).

B.5 Parameter Scalability Desgin

This framework supports dynamic parameter configuration to accommodate varying security requirements across different scenarios, as shown in Table 1.

Table 1: Parameter Configuration for Different Security Modes

Mode	RNA Length (L)	Salt Value Length (B)	Iteration Count	Applicable Scenarios
High-Security Mode	$\geq$128 nucleotides	$\geq$1024 bits	$\geq$200,000	Military-grade Financial-grade encryption
Balanced Mode	64 nucleotides	512 bits	100,000	General data protection
High-Efficiency Mode	32 nucleotides	256 bits	50,000	IoT devices Low-power applications

Summary

Appendix B comprehensively defines the technical implementation of the Crypto-ncRNA framework, demonstrating a deep integration of bioinformatics and cryptography through modular design. Data encoding enhances information density, dynamic folding strengthens randomness, the entropy pool and PBKDF2 ensure key strength, molecular engineering achieves physical unclonability, and parameter scalability provides broad applicability. The synergistic effect of these components offers an efficient, reliable, and verifiable solution for post-quantum cryptographic needs.

C Algorithm Pseudocode

Here is the pseudocode for the crypto-ncRNA encryption algorithm, illustrating the transformation process from plaintext to ciphertext:

Algorithm 1 crypto-ncRNA Encryption

procedure CRYPTO_NCNRA_ENCRYPT($plaintext$, RNA_pool, $salt$)
    $encoded\_data \leftarrow$ BASE64ENCODE($plaintext$)
    $RNA\_sequence \leftarrow$ MAPTOCODONS($encoded\_data$)
    $folded\_RNA \leftarrow$ FOLDRNASTRUCTURE($RNA\_sequence$)
    $dynamic\_key \leftarrow$ GENERATEDYNAMICKEY($RNA\_pool, salt$)
    $encrypted\_data \leftarrow$ ENCRYPTWITHKEY($folded\_RNA, dynamic\_key$)
    $integrity\_check \leftarrow$ SHA256CHECK($encrypted\_data$)
    return $encrypted\_data$, $integrity\_check$
end procedure

function HELPERFUNCTIONS
    Base64Encode($data$): **return** $base64.b64encode(data)$
    MapToCodons($data$): **return** $[codon \mid codon \in generate\_RNA\_codons(data)]$
    FoldRNAStructure($RNA\_sequence$): **return** $linear\_fold\_algorithm(RNA\_sequence)$
    GenerateDynamicKey(RNA_pool, $salt$): **return** $pbkdf2\_algorithm(RNA\_pool, salt)$
    EncryptWithKey($RNA\_sequence$, key): **return** $apply\_encryption(RNA\_sequence, key)$
    SHA256Check($data$): **return** $sha256(data)$
end function

D PERFORMANCE EVALUATION METRICS

This section presents the results of our comprehensive evaluation of the proposed cryptographic method, Crypto-ncRNA, benchmarked against AES-256 and RSA-2048.

Crypto-ncRNA was implemented in Python (v3.12), while AES-256 and RSA-2048 were implemented using the "pycryptodome" library (v3.21.0). Standard configurations were used: AES-256 in CBC mode with PKCS7 padding (Rijmen & Daemen (2001)), and RSA-2048 with PKCS#1 OAEP padding and SHA-256 hashing (Rivest et al. (1978)).

D.1 ENCRYPTION/DECRYPTION EFFICIENCY TEST

Encryption and decryption latencies were measured to assess real-time performance. The execution time was calculated as the difference between process start and end timestamps. The data processing efficiency was quantified using:

$$\text{AverageTime} = \frac{\sum_{j=1}^{n}(EndTime_j - StartTime_j)}{n} \tag{2}$$

Where:

- n is the total number of trials.
- $StartTime_j$ / $EndTime_j$ represents the start or end timestamp of trial j respectively.

Process: Tests spanned data lengths from 50 to 100,000 bytes, with results averaged over 500 runs. System call overhead was subtracted to isolate algorithm-specific performance.

As the Table 2&3, Crypto-ncRNA demonstrates compelling efficiency in both encryption and decryption speeds. Notably, it significantly outperforms RSA across all data lengths, showcasing a substantial advantage, especially when handling larger datasets. While AES maintains a slight speed edge over Crypto-

Table 2: Encryption Time Comparison (s)

Data Length (Bytes)	Crypto-ncRNA	AES	RSA
50	0.030245502	0.018749273	0.403637892
100	0.030270114	0.018605162	0.451251311
500	0.031469746	0.018759693	0.392945505
1000	0.032204049	0.018854617	0.448887759
5000	0.038968784	0.018410643	0.492652349
10000	0.046719104	0.018931629	0.39583042
50000	0.111085906	0.019108713	0.464615586
100000	0.1916002	0.019194982	0.560739354

Table 3: Decryption Time Comparison (s)

Data Length (Bytes)	Crypto-ncRNA	AES	RSA
50	0.030928	0.0187	0.020927
100	0.030578	0.018578	0.021147
500	0.03122	0.018659	0.024541
1000	0.03207	0.018877	0.029646
5000	0.04038	0.018374	0.064815
10000	0.048945	0.018955	0.108585
50000	0.127293	0.01912	0.464025
100000	0.224574	0.019114	0.904652

ncRNA, particularly at larger data lengths, the performance gap is minimal, and at lower data lengths, Crypto-ncRNA's speed is nearly on par with AES. Crucially, Crypto-ncRNA exhibits a stable and linear increase in encryption and decryption time as data volume grows, indicating consistent and predictable performance scaling, ensuring it remains highly efficient even with increasing data loads, a critical attribute for real-world applications.

D.2 Encryption/Decryption Throughput Performance

The encryption and decryption throughput (in Kilobyte/Second) of AES, RSA, and ncRNA algorithms was evaluated to quantify their efficiency in processing varying data lengths. Throughput is defined as the ratio of data length to execution time, calculated using:

$$\text{Average Decryption Throughput (KiB/s)} = \frac{\sum_{i=1}^{n} \frac{L_i}{t_i}}{n} \times \frac{1}{1024} \tag{3}$$

Where:

- n is the total number of trials.
- L_i represents the data length for trial i.
- t_i the execution time for trial i in seconds (s).

For each algorithm, 500 iterations were performed across data lengths ranging from 50 to 6400 bytes, with randomized input strings.

Table 4: Encryption Time Comparison (KiB/s)

Data Length (Bytes)	Crypto-ncRNA	AES	RSA
50	1.542241	2.595423	0.167258
100	3.08258	5.115825	0.331993
200	6.131219	10.3208	0.755908
400	11.97279	20.60843	1.646416
800	23.9814	41.70408	2.930769
1600	45.62274	82.71863	5.897005
3200	84.69407	166.6902	10.42514

Table 5: Decryption Time Comparison (KiB/s)

Data Length (Bytes)	Crypto-ncRNA	AES	RSA
50	1.547998	2.586265	2.297593
100	3.088367	5.143117	4.510009
200	6.080031	10.34722	8.323793
400	12.07701	20.69467	15.45262
800	23.54036	41.49081	27.31485
1600	45.02909	82.86293	43.77247
3200	83.17577	165.9543	62.56794

Table 4 and 5 are as mentioned in C.1, Crypto-ncRNA achieves impressive encryption and decryption throughput. It significantly outperforms RSA across all data lengths, especially large ones. While AES is slightly faster, Crypto-ncRNA's throughput is comparable, particularly for smaller data. Critically, its throughput scales linearly with data volume, ensuring consistent efficiency even with increasing loads.

D.3 Entropy Test

In cryptography, security fundamentally relies on unpredictability, and entropy quantifies this randomness. For encryption algorithms, higher entropy in ciphertext directly translates to stronger security and greater resistance against attacks (Cachin (1997); Rényi (1961)). We strive for an average entropy approaching the maximum of 8 bits per byte, which represents optimal randomness – the closer to 8, the more unpredictable each byte becomes. This principle is clear: higher entropy equals stronger encryption. To rigorously assess this, we perform entropy testing on ciphertext generated over multiple trials. The following formula calculates the average entropy, considering these trials, to validate the security and reliability of cryptographic systems.

Testing Methodology The formula to calculate the average entropy of ciphertext across multiple trials is:

$$\text{Average Entropy} = -\frac{1}{N \times T} \sum_{t=1}^{T} \sum_{i=1}^{N} \log_2(P(c_{i,t})) \tag{4}$$

Where:

- T is the total number of trials (encryption runs).

- N is the length of the ciphertext in bytes for each trial.
- $c_{i,t}$ represents the i-th byte of the ciphertext in the t-th trial.

For each algorithm, 30 iterations were conducted across data lengths from 50 to 50000 bytes, using randomized input strings.

Table 6: Average Entropy (Maximum = 8)

Data Length (Bytes)	Crypto-ncRNA	AES	RSA
50	7.230238137	6.029898017	7.166932575
100	7.520643336	6.553651309	7.1746127
500	7.906697071	7.603863355	7.739013967
1000	7.954174669	7.808256544	7.875851176
5000	7.990878249	7.962331187	7.972694673
10000	7.995409292	7.981768972	7.986424963
50000	7.99908728	7.996283968	7.997251254
10000	7.999546736	7.998177172	7..998614323

Entropy testing (see Table 6) reveals that the Crypto-ncRNA algorithm exhibits significantly higher entropy values compared to both RSA and AES. This directly underscores a key advantage: elevated entropy signifies enhanced randomness and reduced predictability, crucial factors for bolstering cryptographic security. Specifically, the higher entropy strengthens resistance against statistical analysis and brute-force attacks, potentially improving forward security and resilience to future cryptanalytic techniques. Consequently, entropy analysis indicates that Crypto-ncRNA may offer improved security characteristics relative to RSA and AES.

D.4 Operational Reliability

In cryptography, theoretical soundness is insufficient; flawless implementation is paramount. It is a non-negotiable, fundamental requirement that any encryption scheme guarantees 100% accurate decryption, regardless of data length. Real-world implementations are susceptible to subtle errors, unforeseen interactions, and edge-case vulnerabilities that can compromise integrity. These issues may only surface under specific conditions, making superficial testing inadequate.

Therefore, comprehensive testing that rigorously verifies 100% accurate encryption and decryption across a diverse and exhaustive range of data lengths is indispensable. This testing must include small, large, boundary-adjacent, and randomly chosen data lengths to expose potential padding issues, block cipher mode vulnerabilities, and other implementation flaws. This is not optional; it is a fundamental requirement for building trust, ensuring interoperability, meeting security standards, and, crucially, safeguarding sensitive information. Anything less is a dereliction of duty in secure cryptographic system development. The testing ensures that the system can interact with other systems.

Testing Methodology Cryptographic correctness was evaluated by verifying decryption fidelity across varying data lengths: from 50 to 1000000 bytes. For each length, encrypted data was generated and subsequently decrypted. Decrypted output was compared bit-by-bit to the original plaintext. The success rate, representing the percentage of trials with perfect decryption, was calculated for each data length. This directly assesses the fundamental requirement of perfect decryption in a cryptographic system.

The Crypto-ncRNA system demonstrated perfect operational reliability in cryptographic correctness testing, as evidenced in Table 7. Across all tested data lengths, both encryption and decryption success rates were consistently 100.00%. This perfect score, achieved over 1000 trials for each data length, indicates that the

Table 7: Cryptographic Correctness Rate Percentage

Data Length (Bytes)	Total Runs	Enc. Success	Enc. Failure	Dec. Success	Dec. Failure	Success Rate (%)
50	1000	1000	0	1000	0	100.00
100	1000	1000	0	1000	0	100.00
1000	1000	1000	0	1000	0	100.00
100000	1000	1000	0	1000	0	100.00
1000000	1000	1000	0	1000	0	100.00

system reliably encrypts and decrypts data without any bit-level errors, fulfilling the fundamental requirement of 100% decryption accuracy. The system showed no vulnerabilities related to data length, suggesting robust handling of padding, block cipher modes, and other potential implementation-specific issues.

D.5 Statistical Randomness

In cryptography, the strength of an encryption algorithm hinges not just on its mathematical complexity, but critically on the unpredictability of the ciphertext it produces. Any bias, pattern, or predictability in the generated ciphertext can create vulnerabilities, making the encryption susceptible to cryptanalytic attacks and potentially revealing the plaintext. This is where rigorous statistical testing of randomness becomes essential. Weak randomness equates to weak encryption.

To ensure the cryptographic quality and security of our algorithm, and to validate the randomness properties of the PUF-derived outputs, we adhere to the National Institute of Standards and Technology (NIST) Special Publication 800-22, ”A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications.” This suite is a widely accepted, comprehensive standard specifically designed for evaluating the randomness of outputs intended for cryptographic use. NIST SP 800-22 is not merely a guideline; it’s a crucial benchmark. Passing these tests provides strong evidence that our algorithm’s output, including the contributions from the PUF, is statistically indistinguishable from a truly random sequence, a fundamental requirement for robust security.

Table 8: Crypto-ncRNA’s NIST SP 800-22 Randomness Test Matrix Results

Test Name	P Value	Pass/Fail (P/F)
Monobit Test	0.5460386853638187	P
Frequency Within Block Test	0.7963189024290873	P
Runs Test	0.10786751132695774	P
Longest Run Ones in a Block Test	0.22084938122535008	P
Binary Matrix Rank Test	0.587708298333639	P
DFT Test	0.8350238760410118	P
Non-overlapping Template Matching Test	0.9999287413136844	P
Overlapping Template Matching Test	0.43308028660774994	P
Maurer’s Universal Test	0.8514130113443941	P
Linear Complexity Test	0.824328662851978	P
Serial Test	0.507545477157905	P
Approximate Entropy Test	0.5073170913186321	P
Cumulative Sums Test	0.6611638690391457	P
Random Excursion Test	0.1349606453103914	P
Random Excursion Variant Test	0.039767475276814	P

As evidenced in Table 8, the output of our algorithm has demonstrably passed all 15 tests in the NIST SP 800-22 suite, providing statistically significant evidence of its high degree of randomness and strong resistance to cryptanalytic attacks that exploit weaknesses in predictability. This complete and comprehensive success across every test builds significant trust and confidence in its suitability for security-critical applications. The collective results of these tests offer a robust, multi-faceted, and validating assessment, proving that our algorithm, with its foundation in PUF technology, meets and exceeds the stringent requirements for secure cryptographic use.

Summary

Crypto-ncRNA represents a breakthrough in efficiency, security, and reliability (see Appendix C). While its speed and throughput are slightly lower than those of AES-256, they remain well above those of RSA-2048, making Crypto-ncRNA a robust solution for many applications. Its linear scalability is particularly advantageous for big data environments, where predictable performance is essential for managing large-scale processing tasks.

On the security front, Crypto-ncRNA excels by nearly reaching the theoretical maximum entropy of 8 bits per byte. It also passes all NIST randomness tests, demonstrating a high level of unpredictability that is crucial for resisting sophisticated attacks. In rigorous testing, the algorithm achieved a 100% success rate in both encryption and decryption, effectively eliminating common issues such as padding flaws and boundary errors that often plague traditional cryptographic methods.

This deliberate trade-off—sacrificing a modest amount of performance—ensures that Crypto-ncRNA is not merely about speed, but about long-term viability and security in the quantum era. By prioritizing reliability and robustness over maximal throughput, Crypto-ncRNA is well-positioned to address future cryptographic challenges, offering a forward-looking solution that adapts to evolving threats while maintaining strong performance metrics across various applications.