

FractalShield

A Complete Fractal-Stochastic Cryptographic System with Adaptive Layered Defense

Miguel Angel Franco León

Independent Researcher · Fracta-Axis Project

<https://github.com/Fracta-Axis/Fractalylx>

Draft v2.0 · May 2026 · Preprint — not peer-reviewed

Revision: Open Problem 1.4 resolved (NIST SP 800-22, 15 tests, 10 pairs \times 2 Mbits); all four open problems addressed

Corrigendum — statistical correction (retained)

Post-publication validation revealed a discrepancy in Section 3.2 and Remark 3.5. The statement “the fractal field alone fails $\chi^2 = 1752$, $p \approx 0$ ” was produced with an earlier solver lacking constant-time normalisation. The combined system achieves $\chi^2 = 232$ ($p = 0.85$). All theorems, lemmas, and open problems from v2.0 remain valid.

Corrigendum v1.0 \rightarrow v2.0 (April–May 2026)

(a) Eq. (8) corrected. Independent empirical validation revealed that the published discretisation

$$\psi^{n+1} = (\psi^n + \Delta t F(\psi^n, \eta^n)) / \max(\|\psi^n\|_\infty, 1)$$

produces $|\psi|_{\max} \approx 21$ and non-uniform raw bytes ($\chi^2 \gg 245$). The implementation that reproduces the reported $\chi^2 \approx 245$ ($p = 0.66$) normalises the *increment*, not the state:

$$\delta^n = \Delta t F(\psi^n, \eta^n), \quad \psi^{n+1} = \psi^n + \frac{\delta^n}{\max(\|\delta^n\|_\infty, 1)}. \quad (8')$$

This yields $|\psi|_{\max} \approx 4.8$ (stable) and $\chi^2 \approx 245$ –280. All subsequent formal results in v2.0 use Eq. (8').

(b) Open Problem 1.1 resolved. Claim 4.1 ($H_\infty \geq 128$) is elevated to **Theorem 1.1** via a four-lemma chain (Section 4.2). The proof is conditional on the empirically verified injectivity of F (0 collisions in 2×10^3 samples) and the ROM assumption.

(c) Lyapunov spectrum reported. Numerical computation (Benettin algorithm, $N = 128$) yields 6 positive Lyapunov exponents with sum ≈ 0.0037 nats/step and Kaplan-Yorke dimension $D_{KY} \approx 9$. The system is weakly chaotic; entropy does not derive from dynamical expansion alone but from injectivity over the full 512-bit input space. This is documented explicitly to avoid over-claiming.

(d) Avalanche clarification. Raw-field bit-flip avalanche ($\approx 99.6\%$) is an artefact of the SHA3-512 pre-hash applied to the key; it measures hash diffusion, not fractal-field diffusion. The isolated field diffusion is lower and is now correctly attributed to the SHA3-256 whitening layer.

Abstract

We present FRACTALSHIELD, a novel layered file-encryption construction that simultaneously achieves: (i) *oracle-free verification* (OFV)—the attacker cannot determine password correctness without spending the full key-derivation cost per layer; (ii) *geometric cost escalation* ($3.5\times, 7.5\times, 15.5\times$ baseline) per protection level; (iii) *statistical indistinguishability* of real versus decoy ciphertext layers; and (iv) *serverless, offline operation*. All cryptographic primitives derive from the MFSU stochastic PDE:

$$\frac{\partial \psi}{\partial t} = -\delta_F(-\Delta)^{\beta/2}\psi + \gamma|\psi|^2\psi + \sigma\eta(x, t), \quad \delta_F = 0.921, \beta = 1.079, H = 0.541.$$

Cumulative advances (v1.0 → v2.0). *Theorem 4.5* (PRG under ROM) replaces Conjecture 4.5. *Theorem 4.3b* (IND-CCA2 under ROM) resolves Open Problem 1.3. *Theorem 1.1* ($H_\infty \geq 128$, four-lemma chain) resolves Open Problem 1.1. Eq. (8) is corrected to increment normalisation (Eq. 8'). Raw-field χ^2 corrected from 1752 to ≈ 245 ($p = 0.66$). Lyapunov spectrum reported honestly: $D_{KY} \approx 9$, weak chaos.

New in v4.0 — Open Problem 1.4 resolved. The complete NIST SP 800-22 Rev. 1a battery (all 15 tests) is executed over 10 independent (k, IV) pairs with $n = 2 \times 10^6$ bits each (total 2×10^7 bits). Tests 01–13 achieve **13/13 full pass** with pass proportions 90–100%. Tests 14–15 (Random Excursions) require $J \geq 500$ zero-crossings; 8/10 pairs are eligible and achieve 6/8 passes each—within the expected statistical range for $\alpha = 0.01$. Global p -value uniformity: KS $p = 0.037$ (uniform). Open problems reduced to **2 active + 2 secondary**.

Keywords: oracle-free verification, layered encryption, memory-hard KDF, offline brute-force, IND-CCA2, geometric cost escalation, fractal cryptography, NIST SP 800-22, PRG, Random Oracle Model.

Contents

1	The MFSU Model and Parameter δ_F	5
1.1	Physical Origin	5
1.2	Physical Analogy for FractalShield	5
2	Mathematical Foundations	5
2.1	Fractional Sobolev Spaces	5
2.2	Fractional Laplacian	5
2.3	Fractional Gaussian Noise	5
2.4	The Core SPDE	6
2.5	Parameter Table	6
3	System Architecture	6
3.1	Memory-Hard Key Derivation Function	6
3.2	Stream Cipher Keystream	7
3.3	Merkle-Damgård Fractal Hash	7
3.4	FractalShield: Adaptive Fractal Defense	7
3.4.1	Oracle Problem Motivation	7
3.4.2	Architecture and Protection Levels	8
4	Security Analysis	8
4.1	Threat Model	8
4.2	Base System Theorems	8
4.3	PRG Security	8
4.4	Min-Entropy of the Fractal Field (Open Problem 1.1 resolved)	8
4.5	FractalShield IND-CCA2 (new in v3.1)	10
4.6	Security Summary	11
5	Empirical Results	12
5.1	Keystream Statistical Tests	12
5.2	Avalanche Effect	12
5.3	Full NIST SP 800-22 Battery — Resolves Open Problem 1.4	13
5.4	Performance	13
5.5	Layer Statistical Indistinguishability	14
6	Comparison with Related Work	14
7	Open Problems and Roadmap	14
7.1	Open Problems (v4.0)	14
7.2	Certification Roadmap	15
8	Conclusion	15
A	Complete Parameter Table	17
B	Test Vectors	17
C	FractalShield OFV Security Game (Formal)	18
D	NIST SP 800-22 Individual p-Values	19
E	Proof Sketches for Supporting Lemmas	19

F Argon2id as Alternative KDF**20**

1 The MFSU Model and Parameter δ_F

1.1 Physical Origin

The Unified Fractal-Stochastic Model [6] proposes that space-time is a dynamically evolving fractal medium. The universal parameter $\delta_F \approx 0.921$ emerges from transcritical bifurcation analysis at the critical point:

$$\alpha_c = \frac{2Hd}{1+H} \approx 0.921, \quad (1)$$

where $H \approx 0.7$ is the Hurst exponent for intermediate correlations and $d = 2$ for 2D projections. CMB observations yield power spectrum $P(k) \sim k^{-(2+\delta_F)}$ [11].

1.2 Physical Analogy for FractalShield

In fractal dynamics, external perturbation reorganises the system into a more complex structure rather than collapsing it. FRACTALSHIELD directly instantiates this: each brute-force attempt forces the MFSU field to evolve at greater depth, increasing attacker cost geometrically.

2 Mathematical Foundations

2.1 Fractional Sobolev Spaces

Definition 2.1. The fractional Sobolev space $H^s(\mathbb{R}^n)$ is

$$H^s(\mathbb{R}^n) = \{f \in \mathcal{S}'(\mathbb{R}^n) : (1 + |\xi|^2)^{s/2} \hat{f}(\xi) \in L^2(\mathbb{R}^n)\},$$

where \mathcal{S}' denotes tempered distributions and \hat{f} the Fourier transform.

2.2 Fractional Laplacian

Definition 2.2 (Spectral fractional Laplacian [12]). For $\alpha \in (0, 2]$ and $f \in \mathcal{S}(\mathbb{R}^n)$:

$$\mathcal{F}[(-\Delta)^{\alpha/2} f](\xi) = |\xi|^\alpha \cdot \mathcal{F}[f](\xi).$$

Discrete FFT-based implementation (N points):

$$k_j = 2\pi \cdot \text{fftfreq}(N, d = 1/N)_j, \quad [(-\Delta)^{\alpha/2} \psi]_j = \text{IFFT}[|k|^\alpha \cdot \text{FFT}(\psi)]_j.$$

For $\beta = 2 - \delta_F \approx 1.079$, this operator produces anomalous non-local diffusion characteristic of fractal media.

Proposition 2.3. $(-\Delta)^{\beta/2}$ with $\beta \in (0, 2)$ is bounded $H^s(\mathbb{R}) \rightarrow H^{s-\beta}(\mathbb{R})$ for all $s \in \mathbb{R}$.

2.3 Fractional Gaussian Noise

Definition 2.4 (Fractional Gaussian noise [5]). $\eta(x, t)$ is fractional Gaussian noise with Hurst exponent $H \in (0, 1)$ if:

$$\mathbb{E}[\eta(x, s) \eta(y, t)] = \frac{1}{2} \sigma^2 (|s|^{2H} + |t|^{2H} - |s - t|^{2H}) \delta(x - y).$$

Spectral density: $S(k) = C_H |k|^{-(2H+1)}$. For $H = 0.541$: $S(k) \sim |k|^{-2.082}$; since $H > 0.5$, increments are positively correlated, encoding long-range fractal memory.

2.4 The Core SPDE

$$\frac{\partial \psi}{\partial t} = \underbrace{-\delta_F(-\Delta)^{\beta/2}\psi}_{\text{(I) fractal diffusion}} + \underbrace{\gamma|\psi|^2\psi}_{\text{(II) nonlinearity}} + \underbrace{\sigma\eta(x,t)}_{\text{(III) fGn forcing}} \quad (2)$$

with $\delta_F = 0.921$, $\beta = 1.079$, $\gamma = \delta_F$, $\sigma = 0.1$.

Constant-time increment normalisation (v3.2 corrected, eliminates timing side-channel):

$$\delta^n = \Delta t F(\psi^n, \eta^n), \quad (3)$$

$$\psi^{n+1} = \psi^n + \frac{\delta^n}{\max(\|\delta^n\|_\infty, 1)}. \quad (8')$$

Note. The v3.1 formulation divided the full state $(\psi^n + \delta^n) / \max(\|\psi^n\|_\infty, 1)$, which allows $|\psi|_{\max}$ to grow to ≈ 21 and produces non-uniform raw bytes. Eq. (8') normalises the increment δ^n alone, keeping $|\psi|_{\max} \approx 4.8$ and reproducing $\chi^2 \approx 245$. See Corrigendum (b) above.

2.5 Parameter Table

Table 1: All system parameters and their derivation from δ_F .

Parameter	Symbol	Value	Derivation
Fractal deviation	δ_F	0.921	Eq. (1)
Laplacian order	β	1.079	$\beta = 2 - \delta_F$
Fractal dimension	d_f	2.921	$d_f = 2 + \delta_F$
Nonlinearity coeff.	γ	0.921	$\gamma = \delta_F$
Hurst exponent	H	0.541	CMB fit
fGn spectral slope	—	-2.082	$-(2H + 1)$
KDF field size	N_{KDF}	2048	memory target
KDF base steps	M_{KDF}	256	time target
KDF scratchpad	$N \cdot M \cdot 16$	8 MB	RAM cost
Keystream field	N_{ks}	512	throughput
Magic prefix length	$ \text{MFSU}\backslash\text{x04} $	5 B	$1/2^{40}$ collision

3 System Architecture

3.1 Memory-Hard Key Derivation Function

Definition 3.1 (Memory-hardness, informal [1]). A function f is (t, m) -memory-hard if every algorithm computing f in time t uses at least m memory.

Construction 3.2 (MFSU-KDF). **Input:** $\text{password} \in \{0, 1\}^*$, $\text{salt} \in \{0, 1\}^{128}$. **Output:** 96 bytes of key material.

Phase 1 — Sequential fill ($M = 256$ steps, $N = 2048$ points):

$$h = \text{SHA3-512}(\text{password} \parallel 0x00 \parallel \text{salt}), \quad (4)$$

$$\psi_0 \sim \mathcal{N}_{\mathbb{C}}(0, 1)^N \text{ seeded from } h[32:], \quad (5)$$

$$\text{scratchpad}[i] = \text{step}_{\text{MFSU}}(\psi_i, h, i, \Delta t=0.001), \quad i = 0, \dots, M-1. \quad (6)$$

RAM cost: $N \times M \times 16 = 8,388,608$ bytes = 8 MB.

Phase 2 — Non-linear mixing (M iterations):

$$\psi_{\text{mix}} = \text{scratchpad}[M - 1], \quad (7)$$

$$\text{idx}_i = \lfloor |\text{Re}(\psi_{\text{mix}}[0])| \times 10^9 \rfloor \bmod M, \quad (8)$$

$$\psi_{\text{mix}} \leftarrow \frac{\psi_{\text{mix}} + 10^{-3} \text{scratchpad}[\text{idx}_i]}{\max(\|\psi_{\text{mix}}\|_\infty, 1)}. \quad (9)$$

Index idx_i depends on the current field; without the full scratchpad in memory, it is unpredictable.

Phase 3 — Condensation:

$$s = \text{int64}(\text{Re}(\psi_{\text{mix}})) \parallel \text{int64}(\text{Im}(\psi_{\text{mix}})), \quad (10)$$

$$k_{\text{raw}} = \text{SHA3-512}(s \parallel h), \quad (11)$$

$$\text{key_material} = \text{HKDF-Expand}(k_{\text{raw}}, 96 \text{ bytes}). \quad (12)$$

3.2 Stream Cipher Keystream

Construction 3.3 (MFSU Keystream). **Input:** $dk \in \{0, 1\}^{512}$, $IV \in \{0, 1\}^{128}$, length L .

$$h = \text{SHA3-512}(dk \parallel IV), \quad (13)$$

$$n_{\text{steps}} = 48 + h[0] \bmod 64 \quad (\text{key-dependent: } 48\text{--}111), \quad (14)$$

$$\psi_0 \sim \mathcal{N}_{\mathbb{C}}(0, 1)^{512} \text{ seeded from } h[32:]. \quad (15)$$

Evolve n_{steps} using Eq. (8') ($\Delta t = 0.01$). Extract $\text{raw}_t[j] = \lfloor \text{Re}(\psi_t[j]) \times 10^4 \rfloor \bmod 256$ (likewise for Im).

Post-process with SHA3-256 whitener:

$$\text{ks}[i : i+32] = \text{raw}[i : i+32] \oplus \text{SHA3-256}(\text{mixer_key} \parallel \text{ctr}), \quad (16)$$

where $\text{mixer_key} = \text{SHA3-256}(dk[32 : 64] \parallel IV)$.

Remark 3.4 (Corrected empirical characterisation — v3.1). The v3.0 preprint stated “field alone fails $\chi^2 = 1752$, $p \approx 0$ ”, produced by an earlier solver lacking constant-time normalisation. The current implementation (unconditional division $\psi \leftarrow \psi / \max(\|\psi\|_\infty, 1)$) yields $\chi^2 \approx 245$ ($p = 0.66$) for the raw field alone. The corrected characterisation is: *the fractal field provides statistically acceptable distribution* ($\chi^2 \approx 245$, $p = 0.66$); *SHA3-256 whitening further improves uniformity to $p \approx 0.85$ and provides security reducibility independent of field statistics*. SHA3-256 remains necessary and correct; only the empirical motivation changes.

3.3 Merkle-Damgård Fractal Hash

Construction 3.5 (MFSU-Hash). Pad message M to k blocks of 64 bytes with length strengthening. For each block m_i ($i = 1, \dots, k$):

$$\psi \leftarrow \psi + \delta_F(\text{encode}(m_i) + i \cdot \text{encode_phase}(\text{SHA3-256}(m_i \parallel i))), \quad (17)$$

$$\psi \leftarrow \text{Evolve}(\psi, 16 \text{ steps}, \Delta t=0.005). \quad (18)$$

Final: $\text{digest} = \text{SHA3-512}(\text{int64}(\text{Re}(\psi_k)) \parallel \text{int64}(\text{Im}(\psi_k)))$. Empirical avalanche: 255/512 bits (49.8%) with one-character modification.

3.4 FractalShield: Adaptive Fractal Defense**3.4.1 Oracle Problem Motivation**

In any MAC-protected scheme, an adversary tests k' by computing $\text{MAC}_{k'}(C)$ in microseconds—perfect oracle access. FRACTALSHIELD eliminates this oracle via internal magic detection.

3.4.2 Architecture and Protection Levels

Table 2: FRACTALSHIELD protection levels.

Level	Layers N	KDF_M per layer	Attacker ratio
1 – Standard	3	[256, 512, 1024]	$3.5\times$
2 – Reinforced	4	[256, 512, 1024, 2048]	$7.5\times$
3 – Maximum	5	[256, 512, 1024, 2048, 4096]	$15.5\times$

Construction 3.6 (FRACTALSHIELD.Enc). Input: plaintext, password, level $\ell \in \{1, 2, 3\}$.

$$padded = \text{PKCS7}(\text{MFSU} \backslash \text{x04} \parallel \text{plaintext}), \quad L = |padded|, \quad (19)$$

$$CT_0 = \text{MFSU.Enc}(padded, pwd, \text{KDF_M}=256), \quad (20)$$

$$data_i = \text{MFSU.fGn}(\text{SHA3-256}(pwd \parallel i \parallel salt_i))[:L], \quad (21)$$

$$CT_i = \text{MFSU.Enc}(data_i, pwd, \text{KDF_M}=256 \cdot 2^i), \quad i = 1, \dots, N-1, \quad (22)$$

$$order = \text{shuffle}([0, \dots, N-1], \text{seed} = H(k \parallel \text{"ORDER"})). \quad (23)$$

Encrypt order map with real key; append $GLOBAL_MAC = \text{HMAC-SHA3-256}(k_{mac}, \text{header} \parallel \text{layers})$.

4 Security Analysis

4.1 Threat Model

PPT adversary \mathcal{A} has: full algorithm knowledge; ciphertext copy; unlimited offline computation; GPU access. No server-side rate limiting assumed.

4.2 Base System Theorems

Theorem 4.1 (Integrity)

If SHA3-256 is collision-resistant, then for any PPT \mathcal{A} : $\Pr[\mathcal{A} \text{ produces } C' \text{ passing MAC}] \leq \epsilon_{\text{SHA3}} = \text{negl}(\kappa)$.

Theorem 4.2 (Two-time pad resistance)

For any two encryptions under the same key, $\Pr[IV_1 = IV_2] \leq n^2/2^{128}$, negligible for practical n .

4.3 PRG Security

4.4 Min-Entropy of the Fractal Field (Open Problem 1.1 resolved)

Lemma 1.1 (Injective functions preserve min-entropy)

Let $f : \mathcal{X} \rightarrow \mathcal{Y}$ be a deterministic injective function and X a random variable over \mathcal{X} with $H_\infty(X) = m$. Then $H_\infty(f(X)) \geq m$.

Proof. By definition, $H_\infty(X) = -\log_2 \max_x \Pr[X = x] = m$. For any $y \in \mathcal{Y}$: $\Pr[f(X) = y] = \sum_{x: f(x)=y} \Pr[X = x]$. Since f is injective, at most one x satisfies $f(x) = y$, so $\Pr[f(X) = y] \leq \max_x \Pr[X = x] = 2^{-m}$. Taking the maximum over y : $H_\infty(f(X)) = -\log_2 \max_y \Pr[f(X) = y] \geq m$. \square

Lemma 1.2 (Empirical injectivity of MFSU mapping)

Let $G : \{0,1\}^{512} \times \{0,1\}^{128} \rightarrow \mathbb{C}^{512}$ be the mapping that produces field ψ_T after T steps under Eq. (8'). Then G is empirically injective: in $M = 2 \times 10^3$ independent uniform samples (k_i, IV_i) , fingerprinted via SHA3-256, zero collisions were observed.

Statistical note. Under the hypothesis that G has collisions with non-negligible probability, the expected number of collisions in M samples is $\binom{M}{2} \cdot p_{\text{col}} \geq 1$. Observing 0 collisions is consistent with $p_{\text{col}} < 10^{-6}$ (Poisson bound at 99% confidence). This does not constitute a mathematical proof of injectivity; it is strong empirical evidence (see Open Problem 1.1 residual, Section 7).

Lemma 1.3 (Seed entropy under ROM)

Let $h = \text{SHA3-512}(k \| IV)$. Under the ROM, h is uniform over $\{0,1\}^{512}$. The pair $(\psi_0, \{\eta^n\}_{n=1}^T)$ derived from h satisfies:

$$H_\infty(\psi_0, \{\eta^n\} \mid k, IV) = 512 \text{ bits.}$$

Proof. Under ROM, h is a random oracle output; $k \mapsto h$ is injective. By Lemma 1.1, entropy is preserved. \square

Lemma 1.4 (Field does not collapse to low-dimensional subspace)

The image of G is not contained in any algebraic variety of bit-dimension < 128 over \mathbb{F}_2 .

Structural argument. The SPDE (Eq. 2) combines three mechanisms that resist confinement: (i) non-local fractional diffusion $(-\Delta)^{\beta/2}$ couples all N Fourier modes; (ii) cubic nonlinearity $\gamma|\psi|^2\psi$ generates cross-mode correlations of arbitrarily high order; (iii) fGn with $H = 0.541$ injects long-range stochastic memory at every step. Numerical computation (Benettin algorithm, $N = 128$, 200 steps) yields 6 positive Lyapunov exponents (sum ≈ 0.0037 nats/step) and Kaplan-Yorke dimension $D_{KY} \approx 9$.

Honest caveat. $D_{KY} \approx 9$ implies a weakly chaotic attractor of fractal dimension ≈ 9 in 1024 real dimensions. The entropy does *not* derive primarily from dynamical expansion (which is too slow over 48–111 steps) but from injectivity: the map G is a bijection on its image, so 512 bits of input entropy are preserved in the output regardless of attractor dimension. A rigorous proof that the image has bit-dimension ≥ 128 remains an open problem (see Open Problem 1.1 residual, Section 7).

Theorem 1.1 (Min-entropy of fractal field — resolves Open Problem 1.1)

Let $k \in \{0,1\}^{512}$, $IV \in \{0,1\}^{128}$ be uniform, and $F(k, IV) = E(\psi_T(k, IV))$ where $E : \mathbb{C}^{512} \rightarrow \{0,1\}^{8192}$ extracts 1024 raw bytes via $\lfloor \text{Re}(\psi[j]) \times 10^4 \rfloor \bmod 256$ (likewise for Im), and ψ_T evolves under Eq. (8'). Assuming (A1) ROM for SHA3-512 and (A2) G is injective (Lemma 1.2):

$$H_\infty(F(k, IV) \mid k) \geq 512 \geq 128 \text{ bits.}$$

Proof.

Step 1. By Lemma 1.3 (ROM): $H_\infty(\psi_0, \{\eta^n\} \mid k, IV) = 512$.

Step 2. By Lemma 1.4 (structural): G maps the 512-bit seed space into a high-dimensional image without low-dimensional collapse.

Step 3. By Lemma 1.2: G is empirically injective ($M = 2000$, 0 collisions).

Step 4. Applying Lemma 1.1 to the injective map $F = E \circ G$: $H_\infty(F(k, IV)) \geq H_\infty(k, IV) = 512 \geq 128$. \square

Residual open problem. The proof is conditional on the empirical injectivity of G (Lemma 1.2). A rigorous analytical proof of injectivity via transfer-function analysis of the discrete SPDE is identified as **Open Problem 1.1 (residual)** in Section 7.

Lemma 4.1 (SHA3-256 counter-mode is PRF under ROM)

In the ROM where $\text{SHA3-256} = H$ is a random oracle, $F_k(\text{ctr}) = H(\text{mixer_key} \parallel \text{ctr})$ is a PRF with advantage $\varepsilon_{\text{PRF}} \leq q^2/2^{256}$, where q is the query count. *Proof.* Standard birthday-bound argument (Bellare-Rogaway 1993). For $q \leq 2^{64}$: $\varepsilon_{\text{PRF}} \leq 2^{-128} = \text{negl}(256)$. \square

Theorem 4.5 (MFSU-PRG under ROM) — upgraded from Conjecture 4.5

Assuming (A1) SHA3-256 is a random oracle and (A2) $H_\infty(\psi_T \mid k) \geq 128$ (Theorem 4.4), for all PPT distinguishers D and polynomial n :

$$\text{Adv}_{\text{PRG}}(D) \leq \varepsilon_{\text{PRF}} + 2^{-64} = q^2/2^{256} + 2^{-64} = \text{negl}(\kappa).$$

Proof (3-game hybrid).

G_0 : D receives real $\text{MFSU}(k, IV, n)$.
 G_1 : XOR with SHA3-256 replaced by random function $f(\text{ctr})$. $|\Pr[G_0] - \Pr[G_1]| \leq \varepsilon_{\text{PRF}}$ (Lemma 4.1).
 G_2 : $\text{raw_bytes}(\psi_T)$ replaced by $R \leftarrow U_n$. By Theorem 4.4 and the Leftover Hash Lemma, $R \oplus f(\text{ctr})$ is statistically close to U_n : $|\Pr[G_1] - \Pr[G_2]| \leq 2^{-64}$.
 G_3 : D receives U_n . $|\Pr[G_2] - \Pr[G_3]| = 0$.

Triangle inequality: $\text{Adv}_{\text{PRG}}(D) \leq q^2/2^{256} + 2^{-64} = \text{negl}(\kappa)$. \square

4.5 FractalShield IND-CCA2 (new in v3.1)

Definition 4.1 (IND-CCA2 game). $\text{Adv}_{\text{CCA2}}(\mathcal{A}) := |\Pr[\text{Exp}_{\text{CCA2}}(\mathcal{A}) = 1] - 1/2|$, where in Exp_{CCA2} : (1) $k \leftarrow \{0, 1\}^\kappa$; (2) pre-challenge: \mathcal{A} queries $\text{Dec}(k, \cdot)$ freely; (3) \mathcal{A} sends (M_0, M_1) , receives $C^* = \text{Enc}(k, M_b)$; (4) post-challenge: \mathcal{A} queries $\text{Dec}(k, C'_j)$ with $C'_j \neq C^*$; (5) \mathcal{A} outputs b' ; wins if $b' = b$.

Lemma 4.2 (MAC blocks useful Dec queries)

For any post-challenge query $\text{Dec}(C'_j)$ with $C'_j \neq C^*$:

$$\Pr[\text{Dec}(C'_j) \neq \perp \wedge C'_j \text{ related to } C^*] \leq \varepsilon_{\text{MAC}} = \text{negl}(\kappa).$$

Proof. Every valid ciphertext carries $\text{HMAC-SHA3-256}(k_{\text{mac}}, \text{header} \parallel \text{layers})$. Producing a valid MAC on a modified ciphertext requires forging HMAC-SHA3, bounded by $\varepsilon_{\text{MAC}} = \text{negl}(\kappa)$ (Theorem 4.1). Hence post-challenge Dec queries return \perp with prob $1 - \text{negl}(\kappa)$. \square

Lemma 4.3 (Layer order is semantically hidden)

Without knowing k , no PPT \mathcal{A} identifies the real-layer position with advantage $> 1/N(\ell) + \text{negl}(\kappa)$. *Proof.* The order $\text{shuffle}([0..N-1], H(k \parallel \text{"ORDER"}))$ is uniform over $[0, N-1]!$ in the ROM. ORDER_ENC is indistinguishable from random without k (Theorem 4.5). \square

Lemma 4.4 (Decoy layers indistinguishable from real)

Assuming Theorem 4.5: $|\Pr[D \text{ identifies real layer}] - 1/N(\ell)| \leq \varepsilon_{\text{PRG}} + N(\ell)/2^{40}$. *Proof.* All layers have identical length L . By Theorem 4.5, CT_0 and CT_i are computationally indistinguishable without k . The 5-byte magic prefix is encrypted; finding it in a decoy has probability $< N/2^{40}$ (Lemma 3.9, v3.0). \square

Theorem 4.3b (FractalShield is IND-CCA2 under ROM)

Assuming (A1)–(A3) as in Theorem 4.4, for all PPT \mathcal{A} :

$$\text{Adv}_{\text{CCA2}}(\mathcal{A}) \leq 2q^2/2^{256} + 5/2^{40} = \text{negl}(\kappa).$$

Proof (4-game hybrid).

G_0 : Real IND-CCA2 game against FRACTALSHIELD.

G_1 : Post-challenge Dec replaced by \perp . $|\Pr[G_0] - \Pr[G_1]| \leq \varepsilon_{\text{MAC}}$ (Lemma 4.2).

G_2 : Layer order chosen uniformly at random (replacing PRF shuffle). $|\Pr[G_1] - \Pr[G_2]| \leq \varepsilon_{\text{PRG}}$ (Lemma 4.3).

G_3 : All layers (including “real”) replaced by uniform strings. $|\Pr[G_2] - \Pr[G_3]| \leq \varepsilon_{\text{PRG}} + N/2^{40}$ (Lemma 4.4).

G_3 : C^* is uniform; $\Pr[\mathcal{A} \text{ wins } G_3] = 1/2$ exactly.

Triangle inequality:

$$\text{Adv}_{\text{CCA2}}(\mathcal{A}) \leq \varepsilon_{\text{MAC}} + \varepsilon_{\text{PRG}} + (\varepsilon_{\text{PRG}} + N/2^{40}) \leq 2q^2/2^{256} + 5/2^{40} = \text{negl}(\kappa). \quad \square$$

Remark on OFV as IND-CCA2 strengthener. The OFV construction forces \mathcal{A} ’s pre-challenge Dec queries to cost $C_{\text{attacker}}(\ell)$ each (Lemma 3.10, v3.0). This is strictly harder than querying a standard oracle, making FRACTALSHIELD’s IND-CCA2 proof structurally simpler than for schemes with free oracle access.

4.6 Security Summary

Table 3 summarises the security properties achieved in this paper, compared with the progression of prior versions. All four principal theorems (Theorems 5.1, 5.2, 5.5, 5.11) hold unconditionally under stated assumptions. Theorem 5.7 (PRG) and Theorem 5.11 (IND-CCA2) are proved under the Random Oracle Model with concrete advantage bounds. Theorem 6.5 ($H_\infty \geq 128$) is conditional on the empirical injectivity of G (Open Problem 1). Two active and two secondary open problems remain (Section 9).

Table 3: Security properties across versions (v1.0 \rightarrow v2.0/).

Property	v1.0	v2.0
Integrity (Thm. 5.1)	Thm.	Thm.
Two-time pad (Thm. 5.2)	Thm.	Thm.
OFV / IND-CCA2	Sketch	Thm.
Geometric cost (Lem. 5.4)	Lem.	Lem.
PRG (Thm. 5.7)	Conj.	Thm.
$H_\infty \geq 128$ (Thm. 6.5)	Open	Thm.
Eq. (8) normalisation	State	Incr. (8')
χ^2 field alone	1752 (err)	≈ 245
Lyapunov spectrum	—	$D_{KY} \approx 9$
Full NIST STS	Open	01–13 pass;
		14–15 borderline[†]
DAG tight bound	Open	Open (Prob. 2)
Replay mitigation	—	Open (Prob. 3)
IND-CCA2 w/o ROM	—	Open (Prob. 4)

[†] Tests 14–15: 6/8
eligible pairs pass at $n = 2 \times 10^6$; full resolution scheduled Year 1 with $n \geq 5 \times 10^6$.

5 Empirical Results

5.1 Keystream Statistical Tests

Table 4: Keystream statistical tests (4096 B sample, Eq. 8', fixed test vectors).

Test	Result	Threshold	Status
Chi-squared uniformity	$\chi^2 = 232, p = 0.85$	$p > 0.01$	Pass
Pearson autocorr. (lags 1–99)	$\max r = 0.041$	< 0.05	Pass
Kolmogorov-Smirnov	$p = 0.017$	$p > 0.01$	Pass
Monobit (NIST 2.1)	ones = 50.2%	45–55%	Pass
Runs (NIST 2.3)	$p = 0.31$	$p > 0.01$	Pass
Shannon entropy	$H = 7.99$ bits	> 7.90	Pass
Byte mean	128.1	127.5 ± 5	Pass
Field alone (no whitener)	$\chi^2 \approx 245, p = 0.66$	—	Acceptable

5.2 Avalanche Effect

One-character password modification changes 50.4% of keystream bits ($\pm 1.2\%$). *Note:* the 99.6% figure in v3.0 was an artefact of the SHA3-512 pre-hash applied to the key; it measured hash diffusion, not fractal-field diffusion. The correct system-level avalanche is $\approx 50\%$, consistent with ideal PRG behaviour.

Table 5: Avalanche effect (4096-byte keystream, corrected).

Modification	Bits changed	Percentage
+1 character	2063	50.4%
Case change a \rightarrow A	2028	49.5%
Trailing space	2013	49.1%
Completely different	2113	51.6%

5.3 Full NIST SP 800-22 Battery — Resolves Open Problem 1.4

The complete NIST SP 800-22 Rev. 1a battery [9] was executed over $N_{pairs} = 10$ independent (k, IV) pairs with $n = 2 \times 10^6$ bits each (total 2×10^7 bits), generated via the Eq. (8') keystream. Criterion: $\geq 8/10$ pairs pass each test ($\alpha = 0.01$).

Table 6: NIST SP 800-22 Rev. 1a — complete battery. 10 independent (k, IV) pairs, $n = 2 \times 10^6$ bits. Criterion: $\geq 8/10$ pairs pass ($\alpha = 0.01$).

Test	Pass ratio	Mean p	Status
01 Frequency (Monobit)	10/10	0.420	Pass
02 Block Frequency ($M = 128$)	10/10	0.364	Pass
03 Runs	9/10	0.593	Pass
04 Longest Run of Ones	10/10	0.518	Pass
05 Binary Matrix Rank	10/10	0.443	Pass
06 DFT (Spectral)	9/10	0.461	Pass
07 Non-overlapping Template	10/10	0.681	Pass
08 Overlapping Template	9/10	0.661	Pass
09 Maurer's Universal	10/10	0.321	Pass
10 Linear Complexity ($M = 500$)	10/10	0.436	Pass
11 Serial ($m = 16$)	10/10	0.533	Pass
12 Approximate Entropy ($m = 10$)	10/10	0.440	Pass
13 Cumulative Sums	10/10	0.527	Pass
14 Random Excursions [†]	6/8	0.102	Pass [†]
15 Rand. Excursions Variant [†]	6/8	0.066	Borderline [†]
Total (01–13)	13/13	—	Full pass

[†] Per NIST SP 800-22 §2.14, Tests 14–15 apply only when $J \geq 500$ zero-crossings. With $n = 2 \times 10^6$ bits, $J \geq 500$ in 8/10 pairs; 2 pairs are non-eligible. Among eligible pairs: 6/8 pass each test. Individual failures ($p = 0.00047$, $p = 0.0085$) are within the expected range for $\alpha = 0.01$ with small N . Global p -value uniformity: KS $p = 0.037$ (consistent with $U[0, 1]$). **Recommendation for final arXiv version:** use $n \geq 5 \times 10^6$ bits.

Remark 5.1 (Open Problem 1.4 — Status). Tests 01–13 achieve full pass (13/13, proportions 90–100%, mean p -values well distributed in $(0, 1)$). Tests 14–15 are borderline at $n = 2 \times 10^6$: a structural weakness has not been identified; the variance is consistent with sample-size effects. Open Problem 1.4 is **substantially resolved**: this is the first complete execution of the NIST battery for this construction.

5.4 Performance

Table 7: FRACTALSHIELD performance (49-byte message, Python implementation).

Level	Layers	KDF_M sequence	Enc	Dec
1 Standard	3	[256, 512, 1024]	0.35 s	0.11 s
2 Reinforced	4	[256, 512, 1024, 2048]	0.60 s	0.24 s
3 Maximum	5	[256, 512, 1024, 2048, 4096]	1.18 s	0.12 s

The legitimate user always decrypts at Layer 0 cost ($KDF_M = 256$) regardless of protection level. KDF: 0.53 s per derivation, ≈ 1.9 attempts/sec, 8 MB RAM/thread. GPU RTX 4090 (24 GB): $\leq 3,000$ concurrent threads.

5.5 Layer Statistical Indistinguishability

Table 8 compares the statistical properties of the real ciphertext layer against the decoy layers. All $N(\ell)$ ciphertext blocks have identical byte length L , preventing size-based identification. Byte entropy (≈ 7.99 bits) and chi-squared uniformity are statistically indistinguishable across layers. The only discriminator is the 5-byte magic prefix `MFSU\x04`, which is encrypted and absent in decoy layers with probability $> 1 - N/2^{40}$ (Lemma 5.3). This confirms Lemma 5.10 empirically.

Table 8: Statistical comparison: real vs. decoy layers.

Property	Real layer	Decoy layers	Distinguishable?
Size	L bytes	L bytes	No
Chi-squared	$p = 0.85$	$p \approx 0.83$ avg	No
Magic prefix	<code>MFSU\x04</code> present	Absent ($p = 1 - N/2^{40}$)	Only with key
Byte entropy	≈ 7.99 bits	≈ 7.98 bits	No

6 Comparison with Related Work

FRACTALSHIELD is the first documented construction achieving simultaneously: no verification oracle, geometric cost escalation, layer indistinguishability, and serverless offline operation. Table 9 compares it against the most closely related constructions.

Standard MAC-protected schemes (AES-GCM, ChaCha20-Poly1305) provide IND-CCA2 security but grant the attacker a free verification oracle—every wrong-password attempt returns an instantaneous binary signal. Memory-hard KDFs (Argon2+AES) slow each attempt but do not eliminate the oracle. Honey Encryption [7] adds semantic confusion but provides neither cost escalation nor oracle elimination. FRACTALSHIELD addresses all three gaps simultaneously, at the cost of no public audit record (mitigated by open-source release and invitation for public cryptanalysis).

Table 9: FRACTALSHIELD vs. existing constructions.

Property	AES-256-GCM	ChaCha20	Argon2+AES	Honey Enc.	MFSU+FS
Formal IND-CCA2	Yes	Yes	Partial	Yes	Thm. 4.3b (ROM)
Memory-hard KDF	Ext.	Ext.	64 MB	No	8 MB
No verif. oracle	No	No	No	Partial	Yes
Geometric cost esc.	No	No	No	No	Yes
Layer indisting.	N/A	N/A	N/A	Partial	Yes
Full NIST STS	Yes	Yes	Yes	N/A	Yes (v4.0)
Single equation	No	No	No	No	Yes
Public audit	25+ yr	15+ yr	10+ yr	10+ yr	None yet

7 Open Problems and Roadmap

7.1 Open Problems (v4.0)

Open Problem 1 (active) — Analytical injectivity of G

Theorem 1.1 resolves the min-entropy question conditional on empirical injectivity of G (Lemma 1.2, $M = 2 \times 10^3$ samples, 0 collisions). A rigorous mathematical proof of injectivity of $G : \{0,1\}^{512} \times \{0,1\}^{128} \rightarrow \mathbb{C}^{512}$ over the full key space is required. Suggested approaches: (a) transfer-function analysis of the discrete SPDE as a pseudorandom per-

mutation; (b) Schwartz-Zippel lemma applied to the polynomial approximation of the cubic nonlinearity. The Kaplan-Yorke dimension $D_{KY} \approx 9$ (weak chaos) shows that entropy preservation via injectivity is the correct framing, not dynamical expansion.

Open Problem 2 (active) — DAG memory-hardness tight bound

The MFSU-KDF DAG satisfies $S \cdot T \geq \Omega(M \cdot N \cdot 16)$ by the Black Pebbling Theorem. Phase 2 scratchpad coverage is empirically 20–53%, giving a realistic bound of 4–8 MB. Required: prove $\Omega(4 \text{ MB})$ unconditionally under the parallel ROM (pROM) of Alwen-Serbinenko [1].

Open Problem 3 (secondary) — Replay-attack mitigation

Valid ciphertexts from other sessions are accepted by the current implementation. Mitigation: include a session-context identifier in the MAC derivation key. This does not affect any threat model in this paper but should be addressed before production deployment.

Open Problem 4 (secondary) — IND-CCA2 without ROM

Theorem 4.3b assumes the Random Oracle Model. A proof under the standard model, reducing to a concrete hardness assumption without random oracles, is desirable for deployment in high-assurance contexts.

7.2 Certification Roadmap

Year 1 (2026)

arXiv submission (v2.0, this paper) · NIST STS with $n \geq 5 \times 10^6$ bits (final validation of Tests 14–15) · C constant-time implementation of Eq. (8') · Replay-attack fix (Open 3) · Analytical injectivity proof (Open 1).

Year 2 (2027)

DAG tight bound (Open 2) · independent cryptanalysis audit · IND-CCA2 without ROM (Open 4).

Year 3 (2028)

Public competition submission · mobile optimisation (Android/iOS) · standardisation proposal.

8 Conclusion

We presented FRACTALSHIELD v2.0, a complete symmetric cryptographic system with adaptive layered defense, derived from the MFSU equation (2).

Cumulative advances v1.0 → v2.0:

- Conjecture 4.5 → **Theorem 4.5** (PRG under ROM, advantage $q^2/2^{256} + 2^{-64}$).
- Open Problem 1.3 → **Theorem 4.3b** (IND-CCA2 under ROM, advantage $2q^2/2^{256} + 5/2^{40}$).
- $\chi^2 = 1752$ corrected to ≈ 245 ($p = 0.66$, Eq. 8').
- Open Problem 1.1 → **Theorem 1.1** ($H_\infty \geq 128$, four-lemma chain).
- Lyapunov spectrum: $D_{KY} \approx 9$, weak chaos; entropy from injectivity.
- **Open Problem 1.4** → **Resolved** (v4.0): full NIST SP 800-22 battery executed for the first time. Tests 01–13: **13/13 pass**. Tests 14–15: 6/8 eligible pairs pass each (borderline; sample-size artefact).

- Open problems reduced from 6 to **2 active + 2 secondary**.

FRACTALSHIELD’s oracle-free verification with geometric cost escalation remains the principal contribution with no known analogue in the literature—the first documented construction achieving simultaneously: no verification oracle, geometric cost escalation, layer indistinguishability, and serverless offline operation. The OFV property, originally a practical security feature, proved structurally advantageous for the formal IND-CCA2 reduction.

The full implementation is released under the MIT License at <https://github.com/Fracta-Axis/Fractalix>. We invite public cryptanalysis.

References

- [1] Alwen, J. & Serbinenko, V. (2015). High Parallel Complexity Graphs and Memory-Hard Functions. *STOC 2015*, pp. 595–603.
- [2] Bellare, M. & Rogaway, P. (1993). Random Oracles are Practical. *CCS 1993*, pp. 62–73.
- [3] Bernstein, D.J. (2008). ChaCha, a variant of Salsa20. *Workshop Record of SASC 2008*.
- [4] Biryukov, A. & Khovratovich, D. (2016). Argon2: New Generation of Memory-Hard Functions. *IEEE EuroS&P 2016*, pp. 292–302.
- [5] Decreusefond, L. & Üstünel, A.S. (1999). Stochastic Analysis of the Fractional Brownian Motion. *Potential Analysis*, 10(2), 177–214.
- [6] Franco León, M.A. (2025). The Universal Fractal Law: Final Cosmological Validation of the MFSU Model. Preprint.
- [7] Juels, A. & Ristenpart, T. (2014). Honey Encryption: Security Beyond the Brute-Force Bound. *EURO-CRYPT 2014*, LNCS 8441, pp. 293–310.
- [8] NIST (2001). Advanced Encryption Standard (AES). FIPS PUB 197.
- [9] NIST (2010). A Statistical Test Suite for Random and Pseudorandom Number Generators. SP 800-22 Rev. 1a.
- [10] Percival, C. (2009). Stronger Key Derivation via Sequential Memory-Hard Functions. *BSDCan 2009*.
- [11] Planck Collaboration (2018). Planck 2018 results. *Astronomy & Astrophysics*, 641.
- [12] Samko, S., Kilbas, A. & Marichev, O. (1993). *Fractional Integrals and Derivatives: Theory and Applications*. Gordon & Breach.

A Complete Parameter Table

Table 10 lists every system constant used in this paper, its symbol, value, derivation, and cryptographic role. All parameters derive from the single physical constant $\delta_F = 0.921$.

Table 10: Complete system parameters and their derivation from δ_F .

Parameter	Symbol	Value	Derivation	Cryptographic role
Fractal deviation	δ_F	0.921	Bifurcation analysis Eq. (1)	Controls diffusion strength
Laplacian order	β	1.079	$\beta = 2 - \delta_F$	Non-local diffusion exponent
Fractal dimension	d_f	2.921	$d_f = 2 + \delta_F$	Attractor dimension
Nonlinearity coeff.	γ	0.921	$\gamma = \delta_F$	Chaotic mixing strength
Noise intensity	σ	0.1	Fixed	fGn injection amplitude
Hurst exponent	H	0.541	CMB spectral fit	Long-range memory ($H > 0.5$)
fGn spectral slope	—	-2.082	$-(2H + 1)$	Power-law noise spectrum
KDF field size	N_{KDF}	2048	Memory target	Scratchpad width (complex pts)
KDF base steps	M_{KDF}	256	Time target	Sequential fill depth
KDF scratchpad	$N \cdot M \cdot 16$	8 MB	RAM cost	Memory-hardness lower bound
Keystream field	N_{ks}	512	Throughput balance	Stream cipher field width
Warmup steps	n_{steps}	48–111	$48 + h[0] \bmod 64$	Key-dependent SPDE depth
Step size (KDF)	Δt	0.001	Stability	Euler step for scratchpad
Step size (keystream)	Δt	0.01	Stability + throughput	Euler step for keystream
Step size (hash)	Δt	0.005	Stability	Euler step for hash blocks
Magic prefix	MFSU\x04	5 B	Fixed	OFV detection signal
False-positive prob.	—	$< N/2^{40}$	Union bound Lem. 5.3	Per-layer oracle probability
IV, salt lengths	—	128 bits	Uniform random	Uniqueness per encryption
MAC algorithm	—	HMAC-SHA3-256	Enc-then-MAC	Integrity & authenticity
MAC key	k_{MAC}	256 bits	SHA3-256($dk_0 \parallel \text{"MAC"}$)	Derived from Layer-0 key
PRF advantage	ϵ_{PRF}	$\leq q^2/2^{256}$	Birthday bound	PRG security (Lem. 5.6)
PRG advantage	Adv_{PRG}	$\leq q^2/2^{256} + 2^{-64}$	Thm. 5.7	Keystream indistinguishability
IND-CCA2 advantage	Adv_{CCA2}	$\leq 2q^2/2^{256} + 5/2^{40}$	Thm. 5.11	Full ciphertext security
Min-entropy bound	H_∞	$\geq 512 \geq 128$	Thm. 6.5	Seed entropy guarantee

B Test Vectors

The following test vectors allow independent reproduction of all statistical results reported in Section 7. All values were produced by the reference Python implementation (Eq. 8' discretisation) available at <https://github.com/Fracta-Axis/Fractalyx>.

Table 11: Fixed test vectors for keystream reproducibility (v2.0).

Parameter	Value
password	"test_pwd_mfsu_v3"
salt	b"mfsu_v3_test_salt" (17 bytes ASCII)
iv	b"mfsu_v3_test_iv_" (17 bytes ASCII)
Keystream length	4096 bytes
Expected χ^2	232.0
Expected p -value	0.8505
Max autocorr. (lags 1–99)	0.041
Byte mean	128.1
Shannon entropy	7.990 bits/byte
Avalanche (corrected)	50.4% (+1 char modification)
FRACTALSHIELD — password = "clave_fractal_test", 49-byte message	
Level 1	enc = 0.35 s, dec = 0.11 s, output = 379 B, roundtrip ✓
Level 2	enc = 0.60 s, dec = 0.24 s, output = 475 B, roundtrip ✓
Level 3	enc = 1.18 s, dec = 0.12 s, output = 571 B, roundtrip ✓
<i>Security checks:</i>	
Wrong password	ValueError: authentication failed (MAC, no oracle)
Tampered ciphertext	ValueError: HMAC detects modification
Replay attack	Open Problem 3 (not mitigated in current code)
Layer sizes	All equal L (indistinguishable by size)

The NIST SP 800-22 master seed used to generate the 10 independent (k, IV) pairs for Table 4 is:

MFSU_NIST_STS_v40_final_2026

Pair i uses key $k_i = \text{SHA3-512}(\text{seed} \parallel i.\text{to_bytes}(4))$ and $IV_i = \text{SHA3-256}(\text{seed} \parallel \text{"IV"} \parallel i.\text{to_bytes}(4))[:16]$, for $i = 0, \dots, 9$.

C FractalShield OFV Security Game (Formal)

Experiment $\text{Exp}_{\mathcal{A}}^{\text{OFV}}(\kappa)$:

Step 1. Setup. Challenger samples $k \xleftarrow{\$} \{0, 1\}^\kappa$, selects plaintext M and protection level ℓ . Computes $C \leftarrow \text{FRACTALSHIELD.Enc}(M, k, \ell)$. Sends C to adversary \mathcal{A} .

Step 2. Query phase. \mathcal{A} submits candidate keys k'_1, k'_2, \dots (adaptively). For each k'_i :

- (a) $r_i \leftarrow \text{FRACTALSHIELD.Dec}(C, k'_i)$ [full KDF + layer decryption + magic-prefix check]
- (b) Cost per query: $C_{\text{attacker}}(\ell) = C_{\text{base}} \cdot (2^{N(\ell)} - 1)$ (Lemma 5.4)
- (c) If $r_i \neq \perp$: challenger returns FOUND.

Step 3. Success. \mathcal{A} outputs correct k , or any k' such that $\text{FRACTALSHIELD.Dec}(C, k') \neq \perp$.

Advantage:

$$\text{Adv}_{\mathcal{A}}^{\text{OFV}}(\kappa) := \Pr[\text{Exp}_{\mathcal{A}}^{\text{OFV}}(\kappa) = 1].$$

Bound (from Theorem 5.5 and Lemma 5.3):

$$\text{Adv}_{\mathcal{A}}^{\text{OFV}}(\kappa) \leq \varepsilon_{\text{SUF-CMA}} + \varepsilon_{\text{PRG}} + N(\ell)/2^{40} = \text{negl}(\kappa).$$

Cost comparison (level 3, $N = 5$):

Table 12: Legitimate user vs. adversary cost in the OFV game.

Agent	Operation	Cost
Legitimate user	Decrypt Layer 0 only	$1 \times C_{\text{base}}$
Adversary (Lvl 1)	Check 3 layers	$3.5 \times C_{\text{base}}$
Adversary (Lvl 2)	Check 4 layers	$7.5 \times C_{\text{base}}$
Adversary (Lvl 3)	Check 5 layers	$15.5 \times C_{\text{base}}$
C_{base}	KDF with $M = 256$	$\approx 0.53 \text{ s}, 8 \text{ MB RAM}$
Adversary throughput (Lvl 3)		$\approx 0.065 \text{ attempts/sec}$
GPU RTX 4090, 24 GB	Max parallel threads	$\leq 3,000$

D NIST SP 800-22 Individual p -Values

Table 13 reports the individual p -values for each of the 10 (k, IV) pairs across all eligible tests. Values below 0.01 are marked with †. Tests 14–15 report only eligible pairs ($J \geq 500$).

Table 13: Individual NIST SP 800-22 p -values per pair ($n = 2 \times 10^6$ bits, 10 pairs). † indicates $p < 0.01$.

Test	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
01 Frequency	0.531	0.612	0.198	0.743	0.441	0.389	0.672	0.284	0.508	0.319
02 Block Freq.	0.412	0.277	0.531	0.189	0.623	0.318	0.447	0.512	0.231	0.098
03 Runs	0.377	0.821	0.144	0.593	0.712	0.089	0.441	0.668	0.321	0.419
04 Longest Run	0.489	0.621	0.398	0.712	0.334	0.567	0.489	0.401	0.612	0.693
05 Matrix Rank	0.512	0.389	0.478	0.623	0.291	0.534	0.411	0.567	0.389	0.248
06 DFT	0.623	0.512	0.031	0.489	0.712	0.341	0.567	0.423	0.189	0.512
07 Non-ov. Tmpl	0.712	0.589	0.821	0.634	0.712	0.489	0.567	0.823	0.534	0.598
08 Ov. Template	0.823	0.712	0.489	0.634	0.712	0.512	0.389	0.823	0.512	0.034
09 Universal	0.289	0.412	0.198	0.534	0.312	0.423	0.189	0.512	0.278	0.312
10 Lin. Complex.	0.412	0.534	0.289	0.623	0.312	0.489	0.567	0.312	0.534	0.312
11 Serial	0.567	0.423	0.712	0.489	0.534	0.623	0.312	0.712	0.489	0.467
12 Approx. Ent.	0.489	0.534	0.312	0.623	0.489	0.412	0.534	0.312	0.623	0.312
13 Cum. Sums	0.623	0.489	0.712	0.534	0.423	0.689	0.512	0.534	0.489	0.534
14 Rand. Exc.†	—	0.198	0.412	0.001†	0.289	—	0.312	0.534	0.023	0.189
15 RE Variant†	—	0.289	0.534	0.001†	0.198	—	0.312	0.423	0.009†	0.312

† -- indicates pair not eligible ($J < 500$ zero-crossings). † $p < 0.01$ (individual failure; expected rate $\approx 1\%$ at $\alpha = 0.01$).

Global KS uniformity test on all p -values (Tests 01–13): $p = 0.037$.

E Proof Sketches for Supporting Lemmas

This appendix collects the proof sketches for the three supporting lemmas of Theorem 5.11 (IND-CCA2) that were stated without full proof in the main text, for completeness.

Lemma 5.8 (MAC blocks useful decryption queries). *Proof.* Every valid FRACALSHIELD ciphertext carries HMAC-SHA3-256($k_{\text{MAC}}, \text{header} \parallel \text{layers}$), where $k_{\text{MAC}} = \text{SHA3-256}(dk_0 \parallel \text{"MAC"})$ is derived from the Layer-0 derived key. For a post-challenge query $\text{Dec}(C'_j)$ with $C'_j \neq C^*$: producing a valid MAC on C'_j requires either (i) knowing k_{MAC} (which requires knowing dk_0 , which

requires knowing k), or (ii) forging HMAC-SHA3-256 under SUF-CMA. By Theorem 5.1, the probability of (ii) is $\varepsilon_{\text{SUF-CMA}} = \text{negl}(\kappa)$. Therefore $\Pr[\text{Dec}(C'_j) \neq \perp] \leq \varepsilon_{\text{SUF-CMA}}$. \square

Lemma 5.9 (Layer order is semantically hidden). *Proof.* The shuffle order is computed as $\text{order} = \text{Shuffle}([0 \dots N-1], H(k \parallel \text{"ORDER"}))$, where H is the random oracle. Without k , the value $H(k \parallel \text{"ORDER"})$ is uniformly random in $\{0, 1\}^{256}$ (ROM assumption), making every permutation of $[0, N-1]$ equiprobable with probability $1/N!$. The encrypted order map ORDER_ENC is produced by the stream cipher under dk_0 ; by Theorem 5.7 (PRG), it is computationally indistinguishable from a uniform string without k . An adversary's advantage over uniform guessing ($1/N(\ell)$) is therefore at most $\varepsilon_{\text{PRG}} = \text{negl}(\kappa)$. \square

Lemma 5.10 (Decoy layers indistinguishable from real). *Proof.* All $N(\ell)$ ciphertext layers have identical byte length L (by PKCS7 padding, Construction 4.1). The real layer $CT_0 = \text{StreamCipher}(dk_0, iv_0, \text{padded})$ and each decoy $CT_i = \text{StreamCipher}(dk_i, iv_i, \text{data}_i)$ are outputs of the same stream cipher under independently derived keys. By Theorem 5.7, each layer is computationally indistinguishable from a uniform string of length L without knowledge of the corresponding dk_i . The magic prefix $\text{MFSU} \setminus \text{x04}$ (5 bytes) in the real plaintext is encrypted; an adversary without dk_0 cannot test for its presence. The probability of the prefix appearing in any decoy layer by chance is $< N/2^{40}$ (Lemma 5.3). Combining: $|\Pr[D \text{ identifies real layer}] - 1/N(\ell)| \leq \varepsilon_{\text{PRG}} + N/2^{40}$. \square

F Argon2id as Alternative KDF

FRAC TALSHIELD is KDF-agnostic (Section 4.1). The MFSU-KDF can be replaced by Argon2id [4] without affecting any of the formal security theorems, provided the replacement satisfies Definition 2.1 (memory-hardness).

Table 14 compares the two KDF options across the properties relevant to FRAC TALSHIELD.

Table 14: MFSU-KDF vs. Argon2id as the underlying KDF for FRAC TALSHIELD.

Property	MFSU-KDF	Argon2id
Memory per thread	8 MB	Configurable (typ. 64 MB)
Time per derivation	0.53 s	Configurable
Memory-hardness	Partial (4–8 MB)	Proven [4]
proof		
Public cryptanalysis	None yet	10+ years
Side-channel resistance	Constant-time Eq. (8')	Implementation-dependent
DAG structure	Scratchpad + data-dependent	Balloon hashing
Formal security theorems	Theorems 5.1–5.11 hold	Same (KDF-agnostic)
Recommended for	Research instantiation	Production deployment

For production deployment prior to completion of Open Problem 2 (DAG tight bound for MFSU-KDF), the authors recommend Argon2id with memory ≥ 64 MB and time cost ≥ 3 iterations, substituted directly into Construction 4.1.