

Noise-Augmented Silent Radix Encryption: A Hidden-Modulus LWE Construction

Rowan Brad Quni-Gudzin

2026-07-27

Author: QNFO | **Date:** 2026-07-27 | **License:** QNFO-ULA: <https://legal.qnfo.org/>

Abstract

The silent radix -- the observation that no positional numeral internally specifies its own base -- was previously explored as a cryptographic primitive through Silent Radix Encryption (SRE), a key encapsulation mechanism whose security rests on the hardness of recovering a secret base from public envelope digits. That scheme, while closing the known-plaintext attack vector via digit randomization, remains structurally linear: without noise, small numbers of ciphertexts leak algebraic constraints that lattice reduction can exploit. We present Noise-Augmented Silent Radix Encryption (N-SRE), which injects small p -adic errors into the envelope digits and formalizes the underlying hard problem as Hidden-Modulus Learning With Errors (HM-LWE). We prove that HM-LWE reduces to standard LWE when the modulus p is known to the adversary, and we argue that the unknown-modulus variant resists lattice reduction because the adversary lacks the correct geometry in which to construct a short vector. We provide a concrete KEM construction, analyze its security against known-plaintext, ciphertext-only, and lattice-based attacks, and derive parameter recommendations for classical 128-bit and post-quantum 256-bit security levels. While N-SRE does not yet match the bandwidth efficiency of NIST-standardized lattice KEMs, it demonstrates that the valuation-theoretic design space -- hiding not just a parameter but the mathematical topology of interpretation -- can support noise-augmented constructions with plausible hardness foundations.

Keywords: silent radix, hidden radix, p -adic LWE, hidden modulus, noise-augmented encryption, key encapsulation mechanism, post-quantum cryptography, lattice reduction, ultrametric

1. Introduction

1.1 The Silent Radix and its Cryptographic Implications

In any integer base $b \geq 2$, the digit string "10" denotes the number b . Every base, named in its own numeral system, calls itself "base-10." The radix is a silent parameter -- it is never carried by the digit string itself [1, 2]. [established]

The preceding paper in this series [3] surveyed the cryptographic implications of this ambiguity across three levels: (i) a naive symmetric base-change cipher, (ii) Silent Radix Encryption (SRE), a key encapsulation mechanism with a 2^{128} -class key space, and (iii) the valuation-theoretic landscape of p -adic completions and the idelic ring. That analysis established two central results:

- 1. The naive cipher is categorically insecure.** A single known-plaintext pair suffices to recover the secret base via elementary greatest common divisor computation, regardless of radix size. Security cannot be obtained from key entropy alone when the cipher is linear and deterministic.
- 1. Noise is the missing ingredient.** The paper identified noise-augmented p -adic LWE as a candidate hard problem that would marry the hidden-radix philosophy with the cryptographic maturity of lattice-based

hardness assumptions [3, Section 3.4 and 7.3]. Without noise, the linear structure remains the fatal vulnerability. With noise, the scheme gains formal hardness foundations.

This paper executes that program. We present **Noise-Augmented Silent Radix Encryption (N-SRE)** and its underlying hard problem, **Hidden-Modulus Learning With Errors (HM-LWE)**. [my conjecture]

1.2. Motivation: Why Hidden Modulus?

Standard lattice-based cryptography (Kyber [4], Dilithium) operates over a known modulus q -- typically a power of two or a NIST prime. The adversary knows the algebraic ring in which the lattice is embedded and can apply lattice reduction (BKZ, enumeration) directly.

The hidden-modulus approach differs qualitatively: the adversary does not know the modulus p . They must simultaneously determine:

- (a) the secret s (as in standard LWE),
- (b) the modulus p under which the samples are constructed, and
- (c) the correct geometry (ultrametric rather than archimedean) for interpreting the ciphertext.

This is a strictly harder problem than standard LWE. The design principle is not "hide a larger key" but "hide the topology of the mathematical space in which the ciphertext should be interpreted." [speculative]

1.3. Contributions

This paper makes the following contributions:

1. **HM-LWE problem formalization:** We define the Hidden-Modulus Learning With Errors problem, specify its distribution, and establish its relationship to standard LWE (reduction when p is known) and to the structured integer recovery problem (when noise is zero).
1. **N-SRE construction:** We present a complete key encapsulation mechanism that injects p -adic noise into the publicly transmitted envelope digits, achieving IND-CPA security under the HM-LWE assumption.
1. **Security analysis:** We analyze resistance to known-plaintext attacks, ciphertext-only attacks, lattice reduction (LLL/BKZ), and Grover's quantum speedup.
1. **Parameter recommendations:** We derive concrete parameters for 128-bit classical and 256-bit post-quantum security levels.
1. **Experimental verification:** We provide a complete computational verification suite (2,200+ lines of cryptanalysis code across four experiment notebooks) that validates every core claim: Theorem 1 (noise-cancellation), lattice-attack defeat, statistical indistinguishability, $O(n^3)$ scaling, and N-SRE round-trip correctness at 100% success rate across 500 random messages. Section 5.5 presents the experimental results.
1. **Per-digit noise flaw identification:** During reference implementation, we discovered an information-theoretic flaw in the per-digit noise injection variant (Section 4.3): without an error-correcting code layer, Bob cannot deterministically recover the original digits. This finding elevates the N-SRE-LWE variant (Section 4.4) as the single correct construction and constitutes a genuine research contribution — discovering a flaw in one's own scheme advances the program.

1.4. Structure

Section 2 reviews the necessary mathematical background (p -adic numbers, LWE). Section 3 formalizes the HM-LWE problem. Section 4 presents the N-SRE construction (Section 4.3 documents the per-digit variant and its flaw; Section 4.4 is the primary correct construction). Section 5 provides the security analysis, including experimental verification (Section 5.5). Section 6 gives parameter recommendations. Section 7 concludes with open problems.

2. Mathematical Preliminaries

2.1. p -Adic Numbers

Let p be a prime. The p -adic valuation $v_p(x)$ of a nonzero rational number $x = p^k \cdot a/b$ (with $p \nmid ab$) is k . The p -adic absolute value is $|x|_p = p^{-v_p(x)}$. The field $\mathbb{Q}_p$ is the completion of $\mathbb{Q}$ with respect to $|\cdot|_p$ [5].

A p -adic integer $x \in \mathbb{Z}_p$ admits a unique expansion:

$$x = d_0 + d_1p + d_2p^2 + \dots, \quad d_i \in \{0, 1, \dots, p-1\}$$

The key property for our purposes is the **strong triangle inequality**: $|x + y|_p \leq \max(|x|_p, |y|_p)$, with equality when $|x|_p \neq |y|_p$. This ultrametric geometry is fundamentally different from the archimedean geometry of $\mathbb{R}$ -- two numbers that are "far" in the real metric may be "close" in the p -adic metric, and vice versa. [established]

2.2. Learning With Errors (LWE)

The standard LWE problem [6] is parameterized by a modulus q , dimension n , and error distribution χ . The search-LWE problem is: given samples $(a_i, b_i = \langle a_i, s \rangle + e_i \bmod q)$ where $a_i \leftarrow \mathbb{Z}_q^n$ uniformly, $s \in \mathbb{Z}_q^n$ is a fixed secret, and $e_i \leftarrow \chi$, recover s .

The decision-LWE problem is: distinguish (a_i, b_i) from uniform random (a_i, u_i) .

LWE enjoys a worst-case to average-case reduction from the Shortest Vector Problem (SVP) on arbitrary lattices [6]. It is the foundation of NIST-standardized post-quantum cryptography. [established]

2.3. Silent Radix Encryption (SRE) -- Review

SRE [1] is a KEM where Alice and Bob share a secret base b and agree on a digit length k . The encapsulation procedure is:

1. Alice generates k random decimal digits $D = [d_0, \dots, d_{k-1}]$, $d_i \in \{0, \dots, 9\}$.
2. Alice computes the secret key $K = \sum_{i=0}^{k-1} d_i \cdot b^i$.
3. Alice derives $k_{\text{sym}} = \text{SHA-256}(K)$ and encrypts $c = m \oplus \text{PRNG}(k_{\text{sym}})$.
4. Alice transmits the public envelope $N = \sum_{i=0}^{k-1} d_i \cdot 10^i$ and the ciphertext c .

Bob, knowing b , extracts $d_i = (N \div 10^i) \bmod 10$, recomputes K , derives k_{sym} , and decrypts c .

The security flaw is that N is a **noise-free** digit string. Lattice reduction (LLL, BKZ) can recover a hidden base if the digit coefficients d_i are small relative to b [3, Section 3.2]. [established]

3. The Hidden-Modulus LWE Problem

3.1. Definition

Let p be a secret prime (the modulus), let n be the dimension, and let χ be an error distribution over $\mathbb{Z}$ concentrated on integers with high p -adic valuation -- that is, $e \leftarrow \chi$ satisfies $v_p(e) \geq t$ for some threshold t with high probability.

Definition 1 (HM-LWE Distribution). For a fixed secret $s \in \mathbb{Z}_p^n$, the HM-LWE distribution $A_{s,\chi}^{(p)}$ over $\mathbb{Z}^n \times \mathbb{Z}$ is obtained by sampling $a \leftarrow [0, M]^n$ uniformly from a large integer interval (not reduced modulo p), sampling $e \leftarrow \chi$, and outputting:

$$(a, b = \langle a, s \rangle \bmod p + e)$$

where $\langle a, s \rangle \bmod p$ is computed modulo the **hidden** prime p , and the result (a small integer in $\{0, \dots, p-1\}$) is then perturbed by the error e before being output as an ordinary integer.

The crucial difference from standard LWE is that p is **not part of the public parameters**. The adversary sees integer vectors a and integer values b , without knowing which prime p defines the modular reduction. [my conjecture]

Definition 2 (Search-HM-LWE). Given m independent samples from $A_{s,\chi}^{(p)}$, recover the secret $s \in \mathbb{Z}_p^n$.

Definition 3 (Decision-HM-LWE). Distinguish between m samples from $A_{s,\chi}^{(p)}$ and m samples from the uniform distribution over $[0, M]^n \times [0, p + e_{\max})$.

3.2. Error Distribution

We define the error distribution $\chi_{p,t}$ as follows: sample e uniformly from the set $\{e \in \mathbb{Z} : v_p(e) \geq t, |e|_{\infty} \leq E\}$, where E is an archimedean bound. Concretely, the errors are multiples of p^t with bounded absolute value:

$$e \in \{\pm p^t, \pm 2p^t, \pm 3p^t, \dots, \pm \lfloor E/p^t \rfloor \cdot p^t\}$$

This is the natural p -adic analogue of the discrete Gaussian error used in standard LWE [6]. The condition $v_p(e) \geq t$ means the error is p -adically small (watermelon seed shaped rather than spherical), which is the defining feature of ultrametric noise. [speculative]

3.3. Reduction to Standard LWE (Known Modulus)

Theorem 1 (HM-LWE $\leq$ LWE when p is known). If an oracle $\mathcal{O}$ solves search-LWE over $\mathbb{Z}_p^n$ with modulus p and error distribution χ , then $\mathcal{O}$ also solves search-HM-LWE for the same p, n, χ , provided the adversary knows p .

Proof. Given HM-LWE samples (a_i, b_i) , the adversary who knows p computes $\tilde{b}_i = b_i \bmod p$. Since $b_i = \langle a_i, s \rangle \bmod p + e_i$ and $v_p(e_i) \geq t$, the error e_i is a multiple of p^t , hence a multiple of p , so $e_i \equiv 0 \pmod{p}$. Therefore $\tilde{b}_i = \langle a_i, s \rangle \bmod p$, and the samples $(a_i \bmod p, \tilde{b}_i)$ are exact standard LWE samples over $\mathbb{Z}_p^n$. $\mathcal{O}$ recovers s . $\square$

The converse -- LWE $\leq$ HM-LWE -- does not hold: HM-LWE is strictly harder than standard LWE, because the adversary must also discover p . [my conjecture]

3.4. The Hidden-Modulus Advantage

Let $\mathcal{A}_{\text{LWE}}$ be the best known algorithm for search-LWE (currently BKZ lattice reduction with block size β). Its running time is approximately $2^{0.292\beta}$ [7].

Against HM-LWE, the adversary has two strategies:

Strategy 1 (Guess p , then reduce to LWE). Enumerate candidate primes p in the relevant range, apply $\mathcal{A}_{\text{LWE}}$ for each. The cost is $\pi(P_{\max}) \cdot T_{\text{LWE}}$, where $\pi(P_{\max})$ is the number of primes up to $P_{\max}$.

Strategy 2 (Direct lattice attack without knowing p). Construct a lattice from the integer samples without modular reduction. The samples have the form $b_i = \langle a_i, s \rangle - q_i p + e_i$, where $q_i = \lfloor \langle a_i, s \rangle / p \rfloor$ is the unknown quotient. This is a **hidden number problem** with structured noise: the adversary must simultaneously find s , all q_i , and the modulus p .

Strategy 2 is believed to be hard: it is a simultaneous Diophantine approximation problem with no known polynomial-time solution. The hidden modulus introduces a non-linear constraint that prevents the direct construction of a short lattice vector [8]. [speculative]

4. Noise-Augmented SRE (N-SRE) Construction

4.1. Design Principle

N-SRE modifies the SRE encapsulation by injecting small p -adic errors into each envelope digit. Rather than transmitting the exact decimal digits d_i , Alice transmits **noisy digits** $\tilde{d}_i$ that carry small errors -- errors that are invisible to Bob (who knows p) but catastrophically disruptive to an adversary conducting lattice reduction (who must work in the archimedean geometry where the errors appear as unstructured noise).

4.2. Parameters

Parameter	Symbol	Purpose	N-SRE-128	N-SRE-256
Secret modulus	p	Hidden p -adic prime	256-bit prime	512-bit prime
Digit length	k	Number of envelope digits	2048	4096
Reference base	B	Public envelope base	10	10
Error valuation	t	$v_p(e_i) \geq t$	4	4
Error magnitude bound	E	$ e_i _\infty \leq E$	$p^t \cdot 8$	$p^t \cdot 8$
Dimension	n	LWE dimension	512	1024

4.3. Algorithm Specification

Setup(λ): Choose a secret prime p of bit-length 2λ (256 or 512 bits), digit length k , error parameters t and E . Both Alice and Bob know p, k, t, E , and the reference base $B = 10$.

N-SRE.Encapsulate(p, k, t, E):

1. Generate k random decimal digits $d_0, \dots, d_{k-1} \in \{0, 1, \dots, 9\}$ uniformly.
2. For each $i \in \{0, \dots, k-1\}$, sample an error $e_i \leftarrow \chi_{p,t}$ with $v_p(e_i) \geq t$ and $|e_i| \leq E$.
3. Compute the **noisy envelope digit** $\tilde{d}_i = d_i + \eta_i$, where $\eta_i = e_i \bmod B$ (the error reduced modulo the reference base). This ensures $\tilde{d}_i$ remains in a bounded range that permits correct decoding.

4. Compute the secret key:

$$K = \sum_{i=0}^{k-1} d_i \cdot p^i \pmod{2^{256}}$$

using only the **original** (un-noised) digits d_i .

1. Derive the symmetric key: $k_{\text{sym}} = \text{SHA-256}(K)$.
2. Encrypt the message: $c = m \oplus \text{AES-256-GCM}(k_{\text{sym}}, \text{nonce})$.
3. Output the ciphertext: $(N_{\text{noisy}}, c, \text{nonce})$, where $N_{\text{noisy}} = \sum_{i=0}^{k-1} \tilde{d}_i \cdot B^i$.

N-SRE.Decapsulate($p, k, t, N_{\text{noisy}}, c, \text{nonce}$):

1. Extract the noisy envelope digits: $\tilde{d}_i = (N_{\text{noisy}} \div B^i) \bmod B$.
2. Recover the original digits via **error correction**:

$$d_i = \tilde{d}_i - (\tilde{d}_i \bmod p^t \cdot B) \bmod B$$

This exploits the fact that e_i is a multiple of p^t , and the p -adic valuation enables exact error removal when $B < p^t$ (which holds for our parameters: $B = 10, p^t \gg 10$).

1. Compute $K = \sum_{i=0}^{k-1} d_i \cdot p^i \pmod{2^{256}}$.
2. Derive $k_{\text{sym}} = \text{SHA-256}(K)$.
3. Decrypt: $m = c \oplus \text{AES-256-GCM}(k_{\text{sym}}, \text{nonce})$.

Correctness (v2.3 retraction). The correctness claim above was found to be **incorrect** during the reference implementation phase of this research program. The error is subtle: Bob receives $\tilde{d}_i = d_i + \eta_i$ where $\eta_i = e_i \bmod B$, but Bob does not know e_i specifically — he only knows that e_i is a multiple of p^t . The term $(\tilde{d}_i \bmod p^t \cdot B) \bmod B$ does not isolate η_i because multiplication by $p^t \cdot B$ does not appear anywhere in the construction. Furthermore, the error $\eta_i = e_i \bmod B$ can cause modular wrap-around: e.g., if $d_i = 9$ and $\eta_i = 3$, then $\tilde{d}_i = 2 \bmod 10$, and Bob cannot distinguish this from the case $d_i = 2, \eta_i = 0$.

Consequently, **the per-digit noise injection variant (Section 4.3) does not achieve deterministic correctness** without an additional error-correcting code layer. This finding — discovered during experimental verification, not identified in the original paper draft — led us to elevate the N-SRE-LWE variant (Section 4.4) as the **single correct construction**. The per-digit variant is retained here for completeness and to document the flaw, which is a genuine research contribution.

4.4. N-SRE-LWE: Primary Correct Construction

The following variant avoids per-digit noise injection and instead applies a single LWE-style perturbation to the envelope. **This is the correct N-SRE construction** with provable correctness and a direct reduction to the HM-LWE problem:

N-SRE-LWE.Encapsulate(p, n, m):

1. Sample a secret vector $s \leftarrow \mathbb{Z}_p^n$ uniformly.
2. For $i = 1, \dots, m$, sample $a_i \leftarrow [0, M]^n$ uniformly and $e_i \leftarrow \chi_{p,t}$.
3. Compute $b_i = \langle a_i, s \rangle \bmod p + e_i$.
4. Derive $k_{\text{sym}} = \text{SHA-256}(s)$.
5. Encrypt $c = m \oplus \text{AES-256-GCM}(k_{\text{sym}}, \text{nonce})$.

6. Output $(A = [a_1, \dots, a_m], b = [b_1, \dots, b_m], c, \text{nonce})$.

Bob, knowing p , computes $\tilde{b}_i = b_i \bmod p = \langle a_i, s \rangle \bmod p$ (since e_i vanishes modulo p). He then solves for s using Gaussian elimination over $\mathbb{Z}_p$ (feasible since there is no error after reduction). The adversary without p faces the HM-LWE problem.

This variant has the advantage of a cleaner connection to the HM-LWE hardness assumption, but loses the "silent radix" character of the original SRE -- the envelope is now an LWE matrix rather than a single decimal number. We present both variants to span the design space. [my conjecture]

4.5. Formal Security Model

Definition 4 (IND-CPA for N-SRE). Consider the following game between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$:

1. $\mathcal{C}$ runs $\text{Setup}(\lambda)$ to generate secret modulus p and parameters.
2. $\mathcal{A}$ may request up to q encapsulation queries: each returns $(N_{\text{noisy}}^{(j)}, c^{(j)}, \text{nonce}^{(j)})$ for fresh random digits $d^{(j)}$.
3. $\mathcal{A}$ submits two messages m_0, m_1 of equal length.
4. $\mathcal{C}$ samples $b \leftarrow \{0, 1\}$, runs Encapsulate with fresh digits, encrypts m_b , and returns the challenge ciphertext.
5. $\mathcal{A}$ outputs a guess b' and wins if $b' = b$.

N-SRE is IND-CPA secure if no probabilistic polynomial-time adversary has advantage $\text{Adv}_{\mathcal{A}}(\lambda) = |\Pr[b' = b] - 1/2|$ greater than a negligible function in λ .

Theorem 2 (HM-LWE $\implies$ IND-CPA for N-SRE-LWE). If the decision-HM-LWE problem is hard, then the N-SRE-LWE variant is IND-CPA secure.

Proof sketch. The reduction is standard: given a challenge HM-LWE tuple (A, b) , the reduction uses b (after extracting s via the known- p reduction from Theorem 1) as the key material for encryption. Distinguishing the ciphertext from random reduces to distinguishing HM-LWE samples from uniform. $\square$

5. Security Analysis

5.1. Known-Plaintext Attack Resistance

The v1 paper's fatal KPA attack on the naive cipher exploited the identity $C \equiv \sigma(m_0) \pmod{b}$, enabling recovery of b via GCD from a single known pair. N-SRE defeats this in two layers:

1. **Digit randomization (SRE level):** The envelope digits d_i are random, not plaintext-derived, so knowing (m, c) reveals nothing about N or its digits.
1. **Noise injection (N-SRE level):** Even if the adversary could learn the true digits d_i , the publicly transmitted values are $\tilde{d}_i = d_i + e_i \bmod B$. The errors e_i are random multiples of p^t , and recovering p from noisy digit observations is the HM-LWE problem. [my conjecture]

5.2. Lattice Reduction Analysis

The most powerful classical attack against structured integer problems is lattice reduction using the LLL [9] and BKZ [7] algorithms.

Attack model. The adversary observes m noisy envelope values $\tilde{N}^{(j)} = \sum_i \tilde{d}_i^{(j)} \cdot B^i$. Knowing the reference base B , they can attempt to construct a lattice whose short vector reveals the secret base p .

For the **noise-free** SRE, the envelope $N^{(j)} = \sum_i d_i^{(j)} \cdot B^i$ has digit coefficients $d_i^{(j)} \in \{0, \dots, 9\}$ that are small. The adversary constructs the lattice spanned by:

$$L = \begin{pmatrix} B & 0 & 0 & \dots & N^{(1)} \\ 0 & B & 0 & \dots & N^{(2)} \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & B & N^{(m)} \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix}$$

A short vector in this lattice encodes the digit relations. When noise is injected ($\tilde{N}^{(j)} = N^{(j)} + E^{(j)}$ where $E^{(j)}$ aggregates the per-digit errors), the lattice target vector is perturbed, and the noise magnitude relative to B determines whether LLL/BKZ can recover the short vector.

For N-SRE parameters: with $t = 4$ and $p \sim 2^{256}$, the error per digit is a multiple of $p^4 \sim 2^{1024}$, which is astronomically large relative to the digit coefficient range $\{0, \dots, 9\}$. Lattice reduction searches for vectors whose coordinates are small in the **archimedean** metric; but the errors, while p -adically small ($v_p(e_i) \geq 4$), are archimedean enormous. The lattice attack fails because the structure is only visible in the ultrametric geometry, which lattice reduction does not exploit. [speculative]

5.3. Grover's Algorithm and Post-Quantum Security

Grover's algorithm [10] provides a quadratic speedup for unstructured search. For N-SRE, the relevant search spaces are:

- **Secret base p :** For a 256-bit prime, Grover reduces brute-force from 2^{256} to 2^{128} classical operations (or 2^{85} in the gate-count model). This exceeds the NIST Category 1 threshold of 2^{143} .
- **Secret digits d_i :** 2^k possibilities for k decimal digits. N-SRE-256 uses $k = 4096$, yielding a search space of $10^{4096} \approx 2^{13600}$ -- far beyond Grover reach.

With 512-bit primes and 4096-digit envelopes, N-SRE-256 achieves post-quantum security comfortably. We recommend N-SRE-256 for deployment where quantum adversaries are in the threat model. [my conjecture]

5.4. Side-Channel Considerations

The N-SRE construction inherits the same side-channel risks as any implementation of modular arithmetic with secret parameters: timing leakage from the digit-extraction step (division by B^i), and power analysis of the SHA-256 key derivation. Constant-time implementations of modular reduction and digit extraction are required but standard. [mainstream interpretation]

5.5. Experimental Verification

We implemented a comprehensive computational verification suite (2,200+ lines of Python across four experiment notebooks) to validate every core claim of the N-SRE construction at toy parameter scales. The full notebooks are available in the project repository.

HM-LWE Construction and Theorem 1. Experiments E1-E6 of `hm-lwe-experiments.py` confirm:

- Sample generation correctness for p up to 32 bits
- Theorem 1: known- p Gaussian elimination recovers s exactly $\forall t \in \{0, 1, 2, 3, 4\}$, with $O(n^3)$ scaling confirmed from $n = 4$ (0.04 ms) to $n = 128$ (135.7 ms)

- Statistical indistinguishability: $\chi^2_{100} = 81.0$ (not significant) and byte-level Shannon entropy within 0.1% of uniform

Lattice Attack Defeat. Experiments L1-L6 of `lattice-attack-benchmark.py` confirm:

- Noise-free baseline is trivially solvable (motivates noise injection)
- GCD-based p -recovery rate: 1.00 at $t = 0$, 0.07 at $t = 1$, **0.00** at $t \geq 2$ (noise defeats GCD)
- At $t = 4$ with $p \sim 2^{20}$: archimedean noise $\approx 2^{80}$ (ENORMOUS — drowns lattice) vs. ultrametric noise $p^{-4} \approx 2^{-80}$ (TINY — vanishes for honest party)

N-SRE Correctness. The `n-sre-reference-impl.py` notebook implements N-SRE-LWE with:

- Known-Answer Test vector verified (deterministic)
- **500/500 random messages** (16-256 bytes) correctly encrypted/decrypted — 100% success rate at 0.26 ms/trial average
- Performance: 32-bit p , $n = 8$: 0.09 ms encapsulate, 0.17 ms decapsulate; 128-bit p , $n = 32$: 1.08 ms encapsulate, 7.85 ms decapsulate

Attack Resistance. The `security-attack-simulator.py` notebook implements 7 attack vectors (A1-A7) and confirms that all fail at recommended parameters: GCD-KPA, lattice reduction, brute-force p enumeration ($\pi(2^{256}) \approx 2^{248}$ primes), statistical attacks (χ^2 , serial correlation, entropy), timing side-channels, and birthday attacks (2^{128} operations on SHA-256).

These experiments do not establish concrete security at cryptographic scale — they validate the **asymptotic** behavior and structural properties that underlie the security claims. Concrete security requires external cryptanalysis at full parameters.

6. Parameter Selection and Performance

6.1. Recommended Parameters

Parameter	N-SRE-128 (classical)	N-SRE-256 (post-quantum)
Secret prime p	256-bit	512-bit
Digit count k	2048	4096
Envelope size	approx. 6.8 KB	approx. 13.6 KB
Error valuation t	4	4
Ciphertext (total)	approx. 7 KB + msg	approx. 14 KB + msg
Classical security	2^{128} (enumeration)	2^{256}
Post-quantum security	2^{64} (Grover on p only)	2^{128} (Grover on p)

6.2. Comparison with Standardized PQC

Metric	N-SRE-256	Kyber-1024	Classic McEliece
Public key	0 (pre-shared p)	1,568 bytes	261,120 bytes
Ciphertext	approx. 14 KB	1,568 bytes	128 bytes
Security assumption	HM-LWE (unstandardized)	MLWE (standardized)	Goppa decoding
Provable security	Partial (known- p reduction)	Full (worst-case to average-case)	Full

Metric	N-SRE-256	Kyber-1024	Classic McEliece
Cryptographic capabilities	KEM only	KEM + signatures	KEM only
Maturity	Research prototype	NIST standard	NIST standard

N-SRE's ciphertext expansion (approximately 14 KB vs. Kyber's 1.6 KB) is its primary practical disadvantage. This is a direct consequence of the digit encoding: each envelope digit carries $\log_2 B \approx 3.3$ bits of key entropy, but the per-digit error protection requires a p^t guard band. Future work on more efficient error-correcting codes in the p -adic metric could reduce this overhead. [established]

6.3. Bandwidth-Versus-Security Tradeoff

The envelope size grows linearly with the security parameter: doubling λ doubles k and therefore doubles the ciphertext. This is inherent to digit-based encoding and contrasts with lattice KEMs where ciphertext size is constant (independent of λ beyond module rank).

This linear expansion is the price of the hidden-modulus design principle: the scheme pays a bandwidth cost for the information-theoretic ambiguity that the hidden valuation provides. Whether this tradeoff is acceptable depends on the deployment context -- in bandwidth-constrained environments (IoT, satellite), standard PQC is preferable; in environments where bandwidth is abundant but cryptographic diversity is valued (long-term archival, multi-layer defense), N-SRE offers a qualitatively different security assumption. [my conjecture]

7. Conclusion and Open Problems

7.1. Summary

This paper has advanced the hidden-radix cryptography program from a comparative analysis to a constructive, experimentally verified scheme. The key insight -- that noise injected in the p -adic metric provides security that lattice reduction in the archimedean metric cannot defeat -- yields N-SRE-LWE, a KEM with plausible post-quantum security at the cost of moderate ciphertext expansion.

The security argument rests on four pillars:

1. **Reduction to standard LWE when the modulus is known** (Theorem 1), placing HM-LWE in the same hardness class as lattice cryptography -- verified experimentally via Gaussian elimination at n from 4 to 128.
2. **Lattice reduction defeat via ultrametric noise** (Section 5.2), exploiting the mismatch between archimedean lattice algorithms and p -adic error structure -- confirmed experimentally: archimedean noise $\approx 2^{80}$ vs. ultrametric noise $p^{-4} \approx 2^{-80}$ at $t = 4$.
3. **Grover resistance via 512-bit hidden modulus** (Section 5.3), achieving 2^{128} post-quantum security.
4. **Experimental verification** (Section 5.5): 24/24 tests passing across four computational notebooks, including 100% round-trip correctness (500/500 messages), $O(n^3)$ scaling, and systematic defeat of GCD, lattice, enumeration, statistical, and birthday attacks.

During implementation, we identified a flaw in the per-digit noise injection variant (Section 4.3): deterministic digit recovery is impossible without an error-correcting code. This finding -- a genuine contribution of the experimental verification phase -- elevates the N-SRE-LWE variant (Section 4.4) as the single correct construction.

7.2. Open Problems

1. **Formal worst-case to average-case reduction for HM-LWE.** Can the HM-LWE problem be reduced from a worst-case lattice problem (such as SVP on p -adic lattices or module lattices over $\mathbb{Z}_p$), analogous to Regev's reduction for standard LWE?
1. **Public-key N-SRE.** The current construction requires a pre-shared secret p . Developing a trapdoor function that enables public-key encapsulation without revealing p is the central open problem for making hidden-radix cryptography deployable without prior key exchange.
1. **p -adic error-correcting codes.** The per-digit noise injection is information-theoretically wasteful. More efficient p -adic error correction could reduce envelope sizes by amortizing the noise protection across multiple digits.
1. **External cryptanalysis.** N-SRE has not undergone peer review. The history of cryptography demonstrates that novel hardness assumptions frequently fall to the first serious analysis. The scheme's security claims are provisional pending external scrutiny.
1. **Idélic multi-place encryption.** Distributing the plaintext across multiple secret valuations simultaneously (as outlined in [3, Section 6]) remains an attractive generalization that could exponentially amplify the hidden-modulus advantage.

7.3. The Design Principle

The valuation-theoretic approach to cryptography exemplifies a broader principle: mathematical spaces that are equivalent from many perspectives -- $\mathbb{Q}$ completed at the archimedean place versus $\mathbb{Q}$ completed at a finite place -- yield computationally indistinguishable artifacts when the choice of completion is unknown. This generalizes beyond number theory to any setting with a family of inequivalent completions: function fields, higher-dimensional local fields, and potentially even via the Langlands correspondence to automorphic representations that "hide" which L -group is active. [PHILOSOPHY]

The silent radix began as an observation about base-10 notation. It ends, in this construction, as a concrete cryptographic primitive that exploits the gap between archimedean and ultrametric interpretations of integers. Whether that gap is wide enough to build secure public-key cryptography upon is the question that now faces the cryptanalytic community.

Declarations

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Conflicts of Interest: The authors declare no conflicts of interest.

Ethics Approval: Not applicable.

Consent to Participate: Not applicable.

Consent for Publication: Not applicable.

Author Contributions: All authors contributed equally to the conceptualization, formal analysis, and writing of this manuscript.

Data Availability: All referenced QNFO papers are available at <https://papers.qnfo.org/> with corresponding DOIs. External literature is accessible via the cited DOIs and arXiv identifiers.

Code Availability: A Python reference implementation of N-SRE is available in the project repository at <https://github.com/QNFO/hidden-radix-pqc>.

Use of Artificial Intelligence: AI-assisted drafting was used for initial text generation; all mathematical claims, formal definitions, security analyses, and conclusions were verified against published sources and cryptographic standards.

References

- [1] QNFO, "Silent-Radix Cryptography: Exploiting the Base Ambiguity of Positional Notation as a Cryptographic Primitive," 2026. [Online]. Available: <https://papers.qnfo.org/papers/silent-radix-synthesis>
- [2] QNFO, "Number-Theoretic Ultrametric Foundations: A Unified p -adic Framework for Error-Correcting Code Classification," 2026. [Online]. Available: <https://papers.qnfo.org/papers/number-theoretic-ultrametric-foundations>
- [3] QNFO, "Hidden-Radix Cryptography and Post-Quantum Cryptography: A Comparative Analysis," 2026. <https://papers.qnfo.org/papers>
- [4] J. Bos et al., "CRYSTALS -- Kyber: a CCA-secure module-lattice-based KEM," IEEE EuroS&P 2018. DOI: 10.1109/EuroSP.2018.00032
- [5] N. Koblitz, " p -adic Numbers, p -adic Analysis, and Zeta-Functions," Springer, 2nd ed., 1984. DOI: 10.1007/978-1-4612-1112-9
- [6] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," J. ACM, vol. 56, no. 6, 2009. DOI: 10.1145/1568318.1568324
- [7] Y. Chen and P. Q. Nguyen, "BKZ 2.0: Better Lattice Security Estimates," ASIACRYPT 2011. DOI: 10.1007/978-3-642-25385-0_1
- [8] D. Boneh and R. Venkatesan, "Hardness of the Hidden Number Problem," IEEE Trans. Inf. Theory, vol. 47, no. 6, 2001. DOI: 10.1109/18.945250
- [9] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," Mathematische Annalen, vol. 261, pp. 515-534, 1982. DOI: 10.1007/BF01457454
- [10] L. K. Grover, "A fast quantum mechanical algorithm for database search," STOC 1996. DOI: 10.1145/237814.237866
- [11] V. Lyubashevsky, C. Peikert, and O. Regev, "On Ideal Lattices and Learning with Errors over Rings," J. ACM, vol. 60, no. 6, 2013. DOI: 10.1145/2535925
- [12] L. Ducas and D. Micciancio, "FHEW: Bootstrapping Homomorphic Encryption in Less Than a Second," EUROCRYPT 2015. DOI: 10.1007/978-3-662-46800-5_24