

One Unverified Coordinate Is Enough: Key Recovery from Faulty Fujisaki–Okamoto Rejection Checks in ML-KEM

Bhabani Sankar Das

007bsd@proton.me

Abstract. The Fujisaki–Okamoto (FO) transform makes ML-KEM IND-CCA secure by re-encrypting the decrypted message and returning an implicit-rejection value on any mismatch. Deployed implementations of ML-KEM and its predecessor Kyber have realized this check incorrectly: a skipped conditional move (pqc_kyber/cosmian_kyber, RUSTSEC-2026-0290/-0288) and truncated ciphertext comparisons (wolfSSL, CVE-2026-10097 and CVE-2026-6330). We ask how much of the check must be broken before the key is recoverable, and show that one unverified ciphertext coordinate suffices. For a valid ciphertext the adversary knows the encryption randomness, so sweeping an unverified v -coordinate over the compression grid measures the decryption noise there, a known short-coefficient linear form in the secret; one bounded-error observation per message, solved by least squares, recovers the key. Under a regularity assumption we prove that $O(kn \log kn)$ observations identify the key, and we demonstrate full recovery from a single unverified coordinate on all three parameter sets in a few times 10^5 decapsulation queries against a simulated incomplete comparison built on the kyber-py implementation, with ML-KEM-1024 cheapest owing to its finer compression grid. Sessions fall inversely with the number of unverified coordinates but query cost does not: one unverified coordinate costs what a leaked tail of fifty does. The skipped conditional move yields a complete plaintext-checking oracle and $\Theta(kn)$ queries.

Keywords: Fujisaki–Okamoto transform · ML-KEM · implicit rejection · key-encapsulation mechanism · chosen-ciphertext attack · plaintext-checking oracle · implementation flaw · key recovery · post-quantum cryptography

1 Introduction

The Fujisaki–Okamoto (FO) transform [HHK17] is the standard construction that turns a passively secure public-key encryption scheme into an IND-CCA secure key-encapsulation mechanism (KEM), and it is the transform underlying ML-KEM [Nat24], now standardized and deploying at scale in TLS, Signal, and beyond. Its security in the (quantum) random oracle model is well understood [HHK17, BHH⁺19], and it rests on one step of decapsulation. The transform recovers a candidate message from the ciphertext, re-encrypts it under randomness derived from that message, and compares the result against the received ciphertext; on a match it releases a key bound to the message, and on any mismatch it must *implicitly reject*, discarding the candidate and returning a pseudorandom value derived from a secret seed and the ciphertext. This compare-and-select step is the entire difference between IND-CPA and IND-CCA security: it denies the adversary any signal that depends on the decryption of a ciphertext it did not honestly produce. It is also a recurring source of implementation flaws, as the disclosures below and a substantial side-channel literature on the comparison step [GJN20, BDH⁺21, DHP⁺22, UXT⁺22]

attest.

Two families of recently disclosed faults bracket the ways it fails. In the first, a vectorized constant-time conditional move was implemented with a faulty blend so that it never overwrote the candidate key with the rejection value, and the implicit-rejection branch was effectively absent.¹ In the second, an AVX2 ciphertext comparison compared only a prefix of the ciphertext and left a short tail of coordinates unverified (CVE-2026-10097) [Das26].² The first fault removes the entire check; the second leaves all but a few dozen of the ciphertext’s several hundred coordinates verified. The security proofs for the transform assume the check is realized faithfully, and say nothing about a check realized only *partially*. Whether breaking less of the check leaves an implementation partially safe, so that a one-coordinate gap is a modest erosion of the security margin rather than a break, is the question this paper takes up.

The answer is that severity does not scale with how much of the check is broken. There is no partially-safe incomplete comparison: an implementation that leaves even a single ciphertext coordinate unverified admits full key recovery, as does one that skips the check entirely. The two differ in cost and in kind, not in whether they break: the skipped check is a complete plaintext-checking oracle recovering the key in $\Theta(kn) \approx 3 \times 10^3$ crafted-ciphertext queries, whereas a single unverified coordinate needs $\approx 4 \times 10^5$ valid-ciphertext queries; both recover the whole key. We reach this through a general model of a rejection-check fault as a predicate that returns the message-dependent value on a fixed set S of adversarial ciphertexts in place of the key-independent rejection value. For a valid ciphertext the adversary knows the encryption randomness, so each unverified coordinate the fault leaks is a known short-coefficient linear observation of the secret, and tens of thousands of such observations, one per message, recover it by least squares.

Prior models of the FO transform. The provable-security literature establishes IND-CCA security assuming a faithful implicit-rejection branch [HHK17, BHH⁺19] and refines that assumption in directions orthogonal to ours; Hövelmanns et al. [HHM22], for instance, grade security by the underlying scheme’s decryption-failure rate, a property of the scheme rather than of a mis-implemented check. Closest in subject, Glabush et al. [GGHS25] and Delgado [Del26] study a faulty re-encryption check from the defender’s side, asking whether a black-box test can recognize a faulty decapsulator; they bound what such a test can certify, not what an attacker recovers.

Chosen-ciphertext attacks. Key recovery from a reused lattice key is a long line, from the key-reuse attacks of Fluhrer [Flu16] and the misuse analysis of Băetu et al. [BDHD⁺19] to plaintext-checking-oracle recovery [RRCB19, QCZ⁺21], which assumes a full oracle over adversary-chosen ciphertexts that isolate one secret coordinate per query. That the FO comparison step is itself an attack surface is well established: it leaks through timing (Guo, Johansson, Nilsson [GJN20], the closest “the comparison leaks” precedent) and through power/EM even when masked [BDH⁺21, DHP⁺22, UXT⁺22]. Our setting removes the chosen-ciphertext freedom: the fault fires only on valid re-encryptions, so the ephemeral randomness is fixed by the message and the oracle is confined to whatever coordinates the fault happens to leak. The direct antecedent is Hermelink et al. [HPP21]: they use transient fault injection during decapsulation to turn perturbed ciphertexts into a

¹RUSTSEC-2026-0290, <https://rustsec.org/advisories/RUSTSEC-2026-0290.html> (and RUSTSEC-2026-0288, <https://rustsec.org/advisories/RUSTSEC-2026-0288.html>, for the fork *cosmian_kyber*).

²Relationship to ePrint 2026/1682: this is a distinct submission that shares the per-coordinate observation and least-squares solver of [Das26] but adds (i) the leak-set model of a faulty rejection check; (ii) the single-unverified-coordinate ($|T| = 1$) experiments on all three parameter sets (Section 7); (iii) the windowed-identifiability theorem (Theorem 1); (iv) the skipped-conditional-move (cmov) instance and its $\Theta(kn)$ oracle (Section 6); and (v) the cost law $R \approx 2.9 (\Delta/\sigma_w)^2$ with its coverage-independence and the ML-KEM-1024-cheapest inversion. [Das26] treated only the $|T| \approx 51$ tail and none of (i)–(v).

plaintext-checking oracle, obtaining short-coefficient linear inequalities in $(\mathbf{s}, \mathbf{e})$ that they solve by belief propagation (later sharpened by lattice reduction [HMS⁺23]). Our recovery consumes the same kind of observation, but supplied by a *persistent* incomplete comparison rather than an injected fault, so it fires at zero fault-injection cost on every query. Fault-injection attacks that skip the check entirely [XIU⁺21] realize the full-coverage extreme, and the classical analogue of grading an oracle by the geometry of its acceptance region is the padding-oracle hierarchy of Bardou et al. [BFK⁺12]. What we add over this line is the leak-set model, a windowed identifiability bound, and the end-to-end $|T| = 1$ demonstration.

Fragility of implicit rejection is not confined to implementations. At the specification level, HQC’s implicit rejection was recently found to map many malformed ciphertexts to a single rejection value, so that it does not match the guarantees of ML-KEM [Ope25]; our results instead concern implementation faults in a correct specification.

1.1 Contributions and Technical Overview

We now explain why a single leaked coordinate suffices despite the adversary’s loss of freedom. For a valid ciphertext $\text{Enc}(m_0)$ the adversary knows m_0 and hence all of the randomness the transform derives from it. Reading one unverified coordinate of the decapsulation decision and sweeping it over the compression grid measures the decryption noise at that coordinate, a known linear form in the secret $(\mathbf{s}, \mathbf{e})$ whose coefficients are *short*. Each message yields one such bounded-error observation, and because the FO randomness makes distinct messages independent, enough of them (of the order of 10^5 at $|T| = 1$, falling inversely with the coverage) determine the length- $2kn$ secret, which a least-squares solve recovers. The message randomness the transform injects to defeat chosen-ciphertext attacks is thus, in the presence of any leak, exactly what supplies a fresh equation per message. Concretely, our contributions are the following.

- A uniform model of a faulty FO rejection check as an accept predicate with a leak set S and a coverage parameter $|T|$, capturing the skipped conditional move (full coverage) and the incomplete comparison (partial coverage) as instances (Section 3).
- A single-coordinate recovery result (Lemma 1, Theorem 1, and Empirical Result 1): any incomplete comparison leaving even one unverified v -coordinate exposes a short-coefficient linear system in $(\mathbf{s}, \mathbf{e})$; the width- Δ compression reads determine the key at $R = O(kn \log kn)$ observations (windowed identifiability, proved as an order bound) and least squares recovers it at a measured cost captured by $R \approx 2.9 (\Delta/\sigma_w)^2$, so there is no partially-safe incomplete comparison.
- An end-to-end demonstration at $|T| = 1$ against a simulated incomplete comparison built on the `kyber-py` implementation: full-key recovery on ML-KEM-512, -768 and -1024 at a measured 75,000, 90,000 and 28,000 sessions (a few times 10^5 decapsulation queries), the session count falling inversely with the coverage $|T|$ (Section 7).
- The skipped conditional move of `pqc_kyber/cosmian_kyber` (RUSTSEC-2026-0290, -0288) as a full-coverage instance: a complete plaintext-checking oracle recovering the key by coefficient isolation [RRCB19, QCZ⁺21] in $\Theta(kn)$ queries (Section 6).
- A quantitative comparison with the stochastic noisy-oracle model [HMT25]: both recover the short secret from bounded-error linear observations, differing only in the noise source, with the per-observation information set by the compression precision (Section 8).

Relation to [Das26]. The prior work [Das26] is a case study of one flaw — the wolfSSL AVX2 comparison — recovering the key from its tail of about 51 unverified coordinates. This paper is the general treatment: a model of rejection-check faults as accept predicates with a leak set and a coverage parameter (Section 3); a proof that a *single* unverified coordinate identifies the key (Theorem 1); measured recovery at the single-coordinate extreme on all three parameter sets (Section 7); a cost law showing the query cost is coverage-independent and ML-KEM-1024 is the cheapest set (Section 7); and a second, unrelated flaw — the skipped conditional move in `pqc_kyber` and `cosmian_kyber` (RUSTSEC-2026-0290, -0288), with its own artifact — as the full-coverage instance (Section 6). None of these appears in [Das26]. That any reliable plaintext-checking bit yields key recovery is itself the expectation of the PC-oracle line [RRCB19, QCZ⁺21, BDHD⁺19] (our $|T| = 1$ oracle supplies the same kind of observation [HPP21] obtains by a transient fault), so beyond establishing that even one unverified coordinate suffices we regard the quantitative findings as the main results: the cost law $R \approx 2.9 (\Delta/\sigma_w)^2$, its coverage-independence, and the ML-KEM-1024 inversion.

Scope. Our subject is faulty *implementations* of the FO transform, not the transform as specified: ML-KEM implemented to specification is unaffected, and nothing here contradicts its proven security. Recovery uses a decapsulation oracle under a reused key, the standard setting of plaintext-checking attacks; ephemeral-only key shares are out of scope. We prove *identifiability* (the fault determines the key) under a regularity assumption (Assumption 1), and *demonstrate recovery empirically*; the query cost is characterized by experiment and a least-squares heuristic, not a tight closed form. The experiments drive a simulated incomplete comparison built on the `kyber-py` implementation (Section 7), not a fault-injected binary.

2 Preliminaries

2.1 Notation

We write $x \stackrel{\$}{\leftarrow} \chi$ for sampling from a distribution χ , $x \stackrel{\$}{\leftarrow} S$ for a uniform draw from a finite set, and $y := f(x)$ for deterministic assignment. For a predicate P , the Iverson bracket $[P] \in \{0, 1\}$ is 1 iff P holds. Let $q = 3329$ and $R_q = \mathbb{Z}_q[X]/(X^n + 1)$ with $n = 256$; ML-KEM uses module rank $k \in \{2, 3, 4\}$. Vectors over R_q are written in bold, and $\mathbf{s}_{j,a}$ is the a -th coefficient of the j -th component of $\mathbf{s} \in R_q^k$. The functions G, H, J and the key-derivation function are modeled as random oracles.

2.2 Module-lattice PKE and compression

The ML-KEM public-key scheme has short secret key $\mathbf{s} \in R_q^k$ and public key $(\mathbf{A}, \mathbf{t})$ with $\mathbf{A} \in R_q^{k \times k}$ and $\mathbf{t} = \mathbf{A}\mathbf{s} + \mathbf{e}$ for short $\mathbf{e}$. Encryption of $m \in \{0, 1\}^{256}$ draws short $\mathbf{r}, \mathbf{e}_1, e_2$ (in ML-KEM these are derived deterministically from $G(m\|h)$, as Section 2.3 details; this is what lets the adversary recompute them) and forms

$$\mathbf{u} = \mathbf{A}^\top \mathbf{r} + \mathbf{e}_1, \quad v = \mathbf{t}^\top \mathbf{r} + e_2 + \text{Decompress}_1(m),$$

transmitting $c = (c_1, c_2) = (\text{Compress}_{d_u}(\mathbf{u}), \text{Compress}_{d_v}(v))$, where

$$\text{Compress}_d(x) = \lceil (2^d/q) x \rceil \bmod 2^d, \quad \text{Decompress}_d(y) = \lceil (q/2^d) y \rceil$$

are applied coefficientwise. Decryption recovers $\mathbf{u}' = \text{Decompress}_{d_u}(c_1)$, $v' = \text{Decompress}_{d_v}(c_2)$ and returns

$$m' = \text{Compress}_1(v' - \mathbf{s}^\top \mathbf{u}'), \tag{1}$$

so the i -th message bit is 1 iff $(v' - \mathbf{s}^\top \mathbf{u}')_i$ lies closer to $q/2$ than to 0, that is, in $(q/4, 3q/4)$. The map $\mathbf{u}' \mapsto \mathbf{s}^\top \mathbf{u}'$ is the negacyclic convolution used in Section 4.

2.3 The FO transform with implicit rejection

ML-KEM is $\text{FO}[\text{PKE}, G, H, J]$ with implicit rejection. The decapsulation key is $\text{sk} = (\text{dk}, \text{ek}, h, z)$ with $h = H(\text{ek})$ and a rejection seed z sampled independently of $\mathbf{s}$. Write $\text{ReEnc}(m') := \text{Enc}(\text{ek}, m'; \mathbf{r}')$ for re-encryption with $\mathbf{r}'$ derived from $G(m' \| h)$. On input c , decapsulation computes:

1. $m' := \text{Dec}(\text{dk}, c);$
2. $(K', \mathbf{r}') := G(m' \| h);$
3. $c' := \text{ReEnc}(m');$ (re-encryption)
4. **if** $c' = c$ **return** K' **else return** $J(z \| c).$ (compare and select)

Line 4 is the implicit-rejection check: on a mismatch it discards the message-dependent K' and returns the key-independent $J(z \| c)$. For an honestly generated c the re-encryption matches and K' is the shared key; for an adversarial c the faithful scheme rejects, which denies the adversary a plaintext-checking oracle and yields IND-CCA security [HHK17, BHH⁺19]. A KEM is IND-CCA secure if no efficient adversary with a decapsulation oracle distinguishes a real encapsulated key from random; we take this notion as standard and refer to [HHK17, BHH⁺19] for the game.

3 The Faulty-Rejection Model

We model an implementation that realizes the compare-and-select of line 4 (Section 2.3) incorrectly. The faithful implementation accepts, returning K' , exactly on

$$\text{Acc} = \{c : \text{ReEnc}(\text{Dec}(\text{dk}, c)) = c\},$$

the ciphertexts whose re-encryption matches. A faulty implementation accepts on some other set, and the ciphertexts that matter are those it accepts but should not.

Definition 1 (Rejection-check fault and leak set). A rejection-check fault is an accept predicate chk with accept set $\hat{\mathbf{A}} = \{c : \text{chk}(\text{ReEnc}(\text{Dec}(\text{dk}, c)), c) = 1\}$ replacing the faithful test $[c' = c]$. Its *leak set* is

$$S = \hat{\mathbf{A}} \setminus \text{Acc},$$

the adversarial ciphertexts on which decapsulation returns the message-dependent value $K'(m')$ rather than the key-independent $J(z \| c)$.

We treat an arbitrary faulty-but-benign check, not a maliciously chosen one: chk is whatever the implementation computes, and S is a fixed property of the code, identical across all ciphertexts and executions. The two deployed faults of Section 1 are the extreme and the structured case. A skipped conditional move leaves $\hat{\mathbf{A}}$ equal to the whole ciphertext space, so S is the entire adversarial region. An incomplete comparison that verifies only a coordinate subset gives

$$\hat{\mathbf{A}} = \{c : \text{the verified coordinates of } \text{ReEnc}(\text{Dec}(\text{dk}, c)) \text{ match those of } c\},$$

so S is the ciphertexts that differ from their re-encryption only on unverified coordinates.

Definition 2 (Induced plaintext-checking oracle). For a fault with leak set S , the induced oracle on a ciphertext c and candidate message m is

$$\text{PCO}_S(c, m) := [\text{Decaps}_{\text{sk}}(c) = K'(m)] = [c \in \hat{\mathbf{A}}] \wedge [\text{Dec}(\text{dk}, c) = m],$$

up to negligible collision probability of the KDF. On the faithful scheme PCO is identically 0 on adversarial c ; the fault makes it a genuine plaintext-checking oracle exactly on S .

Definition 3 (Coverage). For an incomplete-comparison fault, let T be the set of unverified v -coordinates; the *coverage* is $|T|$, the number of linear observations of $(\mathbf{s}, \mathbf{e})$ that one accepted message exposes (Lemma 1). The skipped conditional move is not measured by $|T|$: it accepts every ciphertext, so the adversary chooses the ephemeral vector freely and obtains a complete plaintext-checking oracle, treated separately in Section 6.

The adversary's goal is key recovery: given the public key and oracle access to the faulty decapsulation $\text{Decaps}_{\text{sk}}$, output the secret $\mathbf{s}$. Theorem 1 and Empirical Result 1 show that whenever the fault has nonzero coverage this succeeds with overwhelming probability.

Remark 1 (Determinism). ML-KEM decapsulation samples no randomness, so PCO_S is a deterministic function of sk : repeating a query adds nothing, unlike the noisy side-channel oracle [HMT25] where repeats average out measurement noise. This does not affect recovery, which never repeats a query but ranges over *distinct* messages (Section 4), each a fresh independent observation; determinism removes only the amplification of a single reading, not the accumulation of many (Section 8).

4 Recovery from a Single Leaked Coordinate

Write the ML-KEM secret as $\mathbf{s} \in R_q^k$ with $R_q = \mathbb{Z}_q[X]/(X^n+1)$, and recall that decryption of a ciphertext with components $(\mathbf{u}, v)$ returns the message whose i -th bit rounds $(v' - \mathbf{s}^\top \mathbf{u}')_i$; see Eq. (1). We first show that a fault leaking even one ciphertext coordinate exposes, per message, a known short-coefficient linear equation in the secret $(\mathbf{s}, \mathbf{e})$.

Lemma 1 (Single-coordinate leakage exposes a short-coefficient equation). *Consider an incomplete-comparison fault whose accept set $\hat{\mathbf{A}}$ verifies only a proper subset of ciphertext coordinates, and let i index any coordinate of the v component left unverified. For a valid ciphertext $c = \text{Enc}(m_0)$ with adversary-chosen m_0 , sweeping the unverified coordinate over the compression grid and observing the accept/reject flip measures the decryption noise at coordinate i ,*

$$\delta_i := (v' - \mathbf{s}^\top \mathbf{u}')_i - m_{0,i} \lceil q/2 \rceil \pmod{q},$$

to precision $\pm q/2^{d_v+1}$. Because m_0 determines the encryption randomness $(\mathbf{r}, \mathbf{e}_1, e_2)$ and, with it, the adversary-recomputable compression errors $\Delta_u = \mathbf{u}' - \mathbf{u}$ and $\Delta_v = v' - v$, this noise is a known linear functional of the short secret $(\mathbf{s}, \mathbf{e})$:

$$\delta_i - e_{2,i} - (\Delta_v)_i = (\mathbf{e}^\top \mathbf{r} - \mathbf{s}^\top (\mathbf{e}_1 + \Delta_u))_i = \langle (\mathbf{s}, \mathbf{e}), \mathbf{w}_i(m_0) \rangle,$$

whose coefficient vector $\mathbf{w}_i(m_0) \in \mathbb{Z}^{2kn}$ is explicit: writing $g_i(a) = 1$ if $a \leq i$ and -1 otherwise (the negacyclic sign), its $(\mathbf{s})$ -block at component t , position a , is $-g_i(a)(\mathbf{e}_1 + \Delta_u)_t \lceil (i-a) \bmod n \rceil$ and its $(\mathbf{e})$ -block is $g_i(a) \mathbf{r}_t \lceil (i-a) \bmod n \rceil$; all entries are short and adversary-known. Each message m_0 thus yields one bounded-error linear observation of $(\mathbf{s}, \mathbf{e})$ with short known coefficients.

Proof. Since $c = \text{Enc}(m_0)$ is valid, its full re-encryption equals c ; knowing m_0 , the adversary computes the encryption randomness from $G(m_0 \| h)$, hence the exact $\mathbf{r}, \mathbf{e}_1, e_2$ and $\mathbf{u}, v$, and from these the decompression errors $\Delta_u = \mathbf{u}' - \mathbf{u}$ and $\Delta_v = v' - v$. Mutating c only at the unverified coordinate i leaves every verified coordinate unchanged,

so $c' \in \hat{A}$ iff its decryption is still m_0 (a distinct message matching on every verified coordinate would also be accepted, but has negligible probability), i.e. iff perturbing v_i did not move $(v' - \mathbf{s}^\top \mathbf{u}')_i$ across a decoding threshold; sweeping over the 2^{d_v} grid levels and reading off the flip locates $(v' - \mathbf{s}^\top \mathbf{u}')_i \bmod q$, hence δ_i , to half a grid step, $\pm q/2^{d_v+1}$. Substituting $\mathbf{t} = \mathbf{A}\mathbf{s} + \mathbf{e}$ into decryption gives the standard identity $v' - \mathbf{s}^\top \mathbf{u}' = m_0 \lceil q/2 \rceil + \mathbf{e}^\top \mathbf{r} - \mathbf{s}^\top (\mathbf{e}_1 + \Delta_u) + e_2 + \Delta_v$; subtracting the known $m_{0,i} \lceil q/2 \rceil$, $e_{2,i}$ and $(\Delta_v)_i$ yields the displayed equation, and expanding the negacyclic products writes it as $\langle (\mathbf{s}, \mathbf{e}), \mathbf{w}_i(m_0) \rangle$ with the stated short coefficients. The entries of $\mathbf{w}_i(m_0)$ are bounded by those of $\mathbf{e}_1$ (in $[-\eta_2, \eta_2]$) and $\mathbf{r}$ (in $[-\eta_1, \eta_1]$) plus the decompression error (at most $q/2^{d_u+1}$), so the observation is well conditioned rather than uniform over $\mathbb{Z}_q$. In particular $|\langle (\mathbf{s}, \mathbf{e}), \mathbf{w}_i(m_0) \rangle|$ has standard deviation ≈ 47 and stays well below $q/2$, so the grid-quantized read of it lifts to an equation over $\mathbb{Z}$ (this is what makes the later least-squares solve meaningful). $\square$

Remark 2 (Relation to prior key recovery). Recovering a short secret from many bounded-error short-coefficient linear observations is a standard problem, solved by the inequality/belief-propagation methods of Hermelink et al. [HPP21] and [Das26] (which our proof of concept replaces with plain least squares). The solver is therefore standard; the results of this paper are the leak-set model (Section 3), the single-coordinate identifiability theorem (Theorem 1), the measured cost law (Section 7), and the full-coverage cmov instance (Section 6). The key enabling observation is that Lemma 1 already holds at $|T| = 1$: each session contributes one fresh equation and the message randomness makes sessions independent, so the tail of $|T| \approx 51$ coordinates that [Das26] leaks is not needed — the same recovery runs down to a single leaked coordinate (Section 7).

Stacking the observations of Lemma 1 over N distinct messages ($R := N|T|$ observations in all) yields, for a fault leaving a nonempty set T of v -coordinates unverified, a linear system

$$M(\mathbf{s}, \mathbf{e})^\top = \mathbf{b} + \boldsymbol{\varepsilon}, \quad M \in \mathbb{Z}^{R \times 2kn} \text{ with short entries, } \|\boldsymbol{\varepsilon}\|_\infty \leq q/2^{d_v+1}, \quad (2)$$

whose rows are the adversary-known short vectors $\mathbf{w}_i(m)$ (the read lifts to $\mathbb{Z}$ by the remark after Lemma 1). We separate what this determines (identifiability) from how cheaply it is solved (recovery cost, measured).

Assumption 1 (Cell-offset regularity). For each fixed nonzero short difference $\mathbf{d}$, over the message randomness the pair (within-cell offset of the read, $\langle \mathbf{d}, \mathbf{w}_i(m) \rangle$) has joint law within statistical distance $\varepsilon = \text{negl}(kn)$ of (uniform on $[0, \Delta)$) $\times$ (the marginal of $\langle \mathbf{d}, \mathbf{w}_i(m) \rangle$). (A merely $o(1)$ distance would support only a $1 - o(1)$ conclusion; the negligible form, which we do not prove, is what the $1 - \text{negl}$ statement of Theorem 1 requires.)

We do not prove this. It is per-candidate and cannot be phrased as plain independence, since the read value $y = \langle (\mathbf{s}, \mathbf{e}), \mathbf{w}_i \rangle$ is a deterministic function of $\mathbf{w}_i$; what the proof uses is only that a fixed short $\mathbf{d}$ decorrelates from the cell offset. This is plausible because the offset is $(\mathbf{t}^\top \mathbf{r})_i \bmod \Delta$, which oscillates fast in $\mathbf{r}$ (a fixed $\mathbf{t}$, uniform mod q , against fresh $\mathbf{r}$) while $\langle \mathbf{d}, \mathbf{w}_i(\mathbf{r}) \rangle$ is a short-coefficient form; a fixed short $\mathbf{d}$ cannot align with the fast phase. It would follow from a Fourier bound on the level sets, which we do not carry out; we take it as an assumption and note that the experiments of Section 7, whose recovery consumes the real reads, validate it in aggregate. The only other place the row's internal structure enters is the fourth-moment bound $\mathbb{E} \langle \mathbf{d}, \mathbf{w}_i \rangle^4 \leq 3(\mathbb{E} \langle \mathbf{d}, \mathbf{w}_i \rangle^2)^2$ used for anti-concentration in Appendix A, which holds for independent platykurtic entries and which the weak dependence of the block $\mathbf{e}_1 + \Delta_u$ on $\mathbf{r}$ (through Δ_u) does not visibly disturb.

Theorem 1 (Windowed identifiability, $|T| = 1$). *Let an incomplete-comparison fault leave a single v -coordinate unverified ($|T| = 1$), and let $\Delta = q/2^{d_v}$ be the compression cell width. Each accepted message pins the short linear form $\langle (\mathbf{s}, \mathbf{e}), \mathbf{w}_i(m) \rangle$ of Lemma 1 to one grid cell of width Δ . Under Assumption 1, there is a threshold $R^* = O(kn \log kn)$ such that over $R \geq 2R^*$ distinct messages no admissible short key other than $(\mathbf{s}, \mathbf{e})$ is consistent with all R cells, with probability $1 - \text{negl}(kn)$; the fault thus determines the secret key, and the public-key identity $\mathbf{e} = \mathbf{t} - \mathbf{A}\mathbf{s}$ then fixes $\mathbf{e}$ from $\mathbf{s}$. There is no partially-safe incomplete comparison: one unverified coordinate already determines the key, whereas $T = \emptyset$ is the correct check and determines nothing.*

Remark 3. This is a sufficient order bound, not tight: Appendix A proves it by a three-regime union bound over short key differences, and its constant (dominated by the anti-concentration regime) puts R^* at roughly 4 to 25 times the measured counts. The tight quantitative cost is empirical: $R \approx 2.9 (\Delta/\sigma_w)^2$ with the root-mean-square σ_w (Section 7). For $|T| > 1$ identifiability follows from the $|T| = 1$ case by restriction (discard the other coordinates); we prove only $|T| = 1$ and *measure* the rate $R_1/|T|$ (Fig. 1), since the $|T|$ rows within a message are negacyclic rotations of one shared randomness.

Proof (Sketch). A wrong candidate $(\mathbf{s}, \mathbf{e}) + \mathbf{d}$ survives observation j only if $\mathbf{d}$ does not move the read across its width- Δ cell, of probability $p(\mathbf{d}) = \mathbb{E}[\max(0, 1 - |\langle \mathbf{d}, \mathbf{w}_j \rangle|/\Delta)]$ under Assumption 1; by independence the survival probability is the exact product $p(\mathbf{d})^R$, with no exceptional event. A first-moment bound with $\mathbb{E}|\langle \mathbf{d}, \mathbf{w} \rangle| \geq c\|\mathbf{d}\|_2$ ($c = \sigma_{\min}/\sqrt{3}$, from Hölder and the platykurtic entries, valid at every support) handles small $\mathbf{d}$, and Paley-Zygmund with the same fourth-moment bound gives $p(\mathbf{d}) \leq \beta < 1$ for larger $\mathbf{d}$ at every support; union-bounding over short $\mathbf{d}$ across the three regimes of Appendix A yields $R^* = O(kn \log kn)$. This is genuine windowed identifiability, not noiseless full rank: the oracle only ever returns width- Δ cells, and the bound is proved from those cells. $\square$

Empirical Result 1 (Exact recovery, cost measured). *Under the bounded noise of Eq. (2), ordinary least squares over $\mathbb{Q}$ followed by rounding to $\mathbb{Z}$ and integer coordinate descent in the box $[-\eta_1, \eta_1]$ returns the exact short $(\mathbf{s}, \mathbf{e})$ once the number of observations is of the order of the threshold of Theorem 1. This is an empirical claim, not a theorem: Section 7 measures the recovery cost directly (75,000, 90,000, 28,000 sessions at $|T| = 1$ for ML-KEM-512, -768 and -1024), and shows it is captured by the least-squares heuristic $R \approx 2.9 (\Delta/\sigma_w)^2$ (within the key-to-key spread), well above the raw information floor $H(\mathbf{s}, \mathbf{e})/I$ of Table 3. At $|T| = 1$ the count of observations is the session count; for $|T| > 1$ each session supplies $|T|$ observations but they are correlated (Remark 3), so the session count is measured (Fig. 1), not inferred from this claim.*

5 Instantiation on Deployed Faults

By Theorem 1 an incomplete-comparison fault fixes the whole key once it leaks even one unverified coordinate, and Empirical Result 1 recovers it: each leaked coordinate contributes a short-coefficient equation (Lemma 1) and the message randomness supplies enough of them. We instantiate the model on the deployed faults in order of decreasing coverage. The full-coverage extreme, the skipped conditional move of `pqc_kyber`, is a complete plaintext-checking oracle and is treated separately in Section 6; here we cover the structured partial-coverage case, the incomplete comparisons. The overview is Table 1: every fault yields full key recovery, differing only in how many coordinates leak, hence in cost.

A wolfSSL AVX2 decapsulation compared only a prefix of the ciphertext, leaving the final coordinates of the v component unverified (CVE-2026-10097) [Das26]. Here $\hat{\mathbf{A}}$ accepts whenever the verified coordinates match, so S is the set of ciphertexts differing from their

Table 1: The model applied to three deployed rejection-check faults in ML-KEM-family implementations, with their advisory identifiers (the `pqc_kyber` crate implements Kyber round 3, whose PKE parameters are identical to ML-KEM; see Section 6). Coverage ranges from a complete oracle down to a single unverified coordinate, yet every fault yields full key recovery; the measured cost (queries or sessions) is in Section 7.

Fault (implementation)	Coverage	Recovery	Shown by
Skipped <code>cmov</code> , <code>pqc_kyber/cosmian_kyber</code> AVX2 RUSTSEC-2026-0290, -0288	full PCO	full	this work
Incomplete compare, wolfSSL AVX2 CVE-2026-10097 [Das26]	$ T \approx 51$	full	this work, [Das26]
NEON blend, wolfSSL NEON CVE-2026-6330	$ T \geq 1$	full if in v , open if in u	this work (simulated $ T = 1$)

re-encryption only on the unverified tail, and the coverage equals the number of unverified coordinates, a structured proper subset rather than the whole space. By Lemma 1 each unverified coordinate yields one short-coefficient observation of $(\mathbf{s}, \mathbf{e})$; stacking them over messages recovers the whole key (Empirical Result 1). The wolfSSL flaw leaves $|T| \approx 51$ coordinates and was recovered end to end in [Das26]; we push the same recovery to the extreme of a single unverified coordinate in Section 7. A NEON analogue of the same comparison, a vectorized blend leaving one or more coordinates unverified (CVE-2026-6330, reported by Carlini [Car26]), has coverage $|T| \geq 1$; the public advisory does not specify whether the unverified coordinates fall in the u or the v component. If in v , our $|T| = 1$ experiment already realizes its minimal case (so we build no separate NEON binary); if in u , the observation is the different first-flip form discussed in Section 7 and the case is open.

6 The skipped conditional move: a complete plaintext-checking oracle

The second, unrelated flaw is a different bug class in a different codebase, and it is the full-coverage extreme of the model. The AVX2 backend of the `pqc_kyber` crate³ implemented its constant-time conditional move with a faulty blend that left it an effective no-op, so decapsulation never overwrote the candidate key with the rejection value (RUSTSEC-2026-0290; the same defect affects the fork `cosmian_kyber`, RUSTSEC-2026-0288; both crates are unmaintained and were flagged through RustSec rather than patched). The missing move leaves the returned key equal to the message-derived K' from $(K', \mathbf{r}') := G(m' \| h)$ with $m' = \text{Dec}(\text{dk}, c)$ on *every* ciphertext, so $\hat{\mathbf{A}}$ is the whole ciphertext space and S the entire adversarial region: the fault is a complete plaintext-checking oracle. The adversary is then not even restricted to valid re-encryptions — it crafts wire codes directly, choosing the ephemeral $\mathbf{u}'$ freely — so Lemma 1 is unnecessary and the coefficient-isolation strategy

³`pqc_kyber` (and its fork `cosmian_kyber`) implements CRYSTALS-Kyber round 3, not FIPS 203 ML-KEM: its last release (0.7.1, 2023) predates the standard. The PKE parameters $(k, \eta_1, \eta_2, d_u, d_v)$ are identical to ML-KEM’s, and the round-3 FO differs only in the final KDF ($K = \text{KDF}(\bar{K} \| H(c))$, rejection $\text{KDF}(z \| H(c))$), which the attacker computes exactly as in ML-KEM; the attack is therefore unaffected. We label the parameter sets Kyber- $\{512, 768, 1024\}$ below, with parameters identical to ML-KEM- $\{512, 768, 1024\}$.

Table 2: Skipped-cmov full-coverage key recovery, run against the actual vulnerable `pqc_kyber` 0.7.1 AVX2 build (Kyber round 3; parameters identical to ML-KEM), over ten independent keys per set: mean decapsulation queries with the observed range, each recovered key exact against ground truth.

Parameter set	Coverage	(d_u, d_v)	Queries (mean, range)	Success
Kyber-512 ($\equiv$ ML-KEM-512)	full PCO	(10, 4)	2876 (2858–2887)	10/10
Kyber-768 ($\equiv$ ML-KEM-768)	full PCO	(10, 4)	4263 (4213–4318)	10/10
Kyber-1024 ($\equiv$ ML-KEM-1024)	full PCO	(11, 5)	9644 (9550–9784)	10/10

of [RRCB19, QCZ⁺21] recovers $\mathbf{s}$ in $\Theta(kn)$ queries.

The linear form read by one query. ML-KEM decryption recovers the message coordinatewise as $m' = \text{Compress}_1(v' - \mathbf{s}^\top \mathbf{u}')$, where $\text{Compress}_1(x) = \lceil 2x/q \rceil \bmod 2$ is 1 exactly when $x \bmod q \in (q/4, 3q/4)$. With $\mathbf{u}'$ free the k -th recovered bit is

$$b_k = \text{Compress}_1(v'_k - (\mathbf{s}^\top \mathbf{u}')_k), \quad (\mathbf{s}^\top \mathbf{u}')_k = \langle \mathbf{w}_k(\mathbf{u}'), \mathbf{s} \rangle,$$

with $\mathbf{w}_k(\mathbf{u}')$ the coefficients of $\mathbf{u}'$ under the negacyclic action $X^n = -1$. Each query reads one adversary-known linear functional of the secret; unlike the pinned- $\mathbf{u}'$ observations of Lemma 1, here $\mathbf{u}'$ is free, so the adversary can *isolate* one coefficient per query.

Isolating a single secret coefficient. To read $\mathbf{s}_t[i]$, set $\mathbf{u}'_t := C X^{n-i}$ and $\mathbf{u}'_{t'} := 0$ for $t' \neq t$. The reduction $X^{n-i} X^i = -1$ sends the $\mathbf{s}_t[i]$ term to coordinate 0: $(\mathbf{s}^\top \mathbf{u}')_0 = -C \mathbf{s}_t[i]$, so $b_0 = \text{Compress}_1(v_0 + C \mathbf{s}_t[i])$. Sweeping v_0 over its 2^{d_v} levels, b_0 flips as $v_0 + C \mathbf{s}_t[i]$ crosses a decode boundary, pinning $C \mathbf{s}_t[i]$ to precision $q/2^{d_v}$. Two conditions make the read clean: $\eta_1 C < q/4$ keeps all products $C \mathbf{s}_t[i]$ ($\mathbf{s}_t[i] \in \{-\eta_1, \dots, \eta_1\}$) in one decode interval, and $C > q/2^{d_v}$ separates consecutive candidates by more than a grid step. Both hold with room at every set: for $\eta_1 = 2$ any $C \in (q/2^{d_v}, q/8)$ (e.g. $C \approx q/10$), for $\eta_1 = 3$ any $C \in (q/2^{d_v}, q/12)$.

Cost, measured on the vulnerable build. There are kn coefficients, giving $\Theta(kn)$ oracle calls. Table 2 reports the measured cost, and unlike the incomplete-comparison experiment (Section 7, on `kyber-py`) this proof of concept runs against the *actual vulnerable build*: `pqc_kyber` 0.7.1 with the `avx2` feature, on stable Rust; the accept signal is read by comparing the returned shared secret against the two candidate KDF outputs (accept if it equals the message-derived K'). It recovered the full key in every one of ten trials per set, at a mean of 2876, 4263, 9644 queries for Kyber-512, -768, -1024, each key checked against ground truth and the crate’s own decryption on 64 ciphertexts. The count is linear in kn (Kyber-512 to -768: factor 1.48 against the dimension ratio 1.50). Table 2 is an *unoptimized* baseline: per coefficient it spends 5.6 queries at $d_v = 4$ and 9.4 at $d_v = 5$, more than the $\log_2(2^{d_v})$ a bare grid bisection costs (the excess ≈ 1.5 –4 queries is bracketing and verification, and the $9.4/5.6 \approx 1.7$ rise mixes the grid $5/4$ with this overhead). An optimized isolation bisects the *candidate set* — $\lceil \log_2(2\eta_1 + 1) \rceil \leq 3$ queries per coefficient, independent of d_v — giving $\Theta(kn \log \eta_1)$, matching the parallel-PC-oracle counts of [QCZ⁺21, RRD⁺23]; the d_v -dependence we report is an artifact of the sweep, not the oracle.

7 Experiments

We validate the recovery of Empirical Result 1 directly, against the real grid-quantized oracle rather than a noiseless model: full recovery of $(\mathbf{s}, \mathbf{e})$ from the bounded-error obser-

vations of Lemma 1 at a single unverified coordinate, and the scaling of the session count with coverage.

7.1 Setup

We instantiate the incomplete-comparison oracle in a clean Python ML-KEM implementation (kyber-py, spec-conformant but not the NIST reference code) at full parameters ($n = 256$, $q = 3329$, $k \in \{2, 3, 4\}$). The oracle is the real one at the interface: for each message m_0 we build a valid ciphertext, overwrite the unverified v -coordinate with a grid level, and query the kyber-py decapsulation `_k_pke_decrypt` for the accept/reject bit, exactly what an incomplete comparison would leak. The accept bit is 1 on an arc of the 2^{d_v} levels (the message bit crosses two decode thresholds around the grid), so it is not globally monotone; but the true level of the valid ciphertext is known, and bracketing to the half-range on one side of it isolates a single monotone threshold, which we then binary-search. This locates the decoding-noise flip, giving one observation of $(\mathbf{s}, \mathbf{e})$ with error $\pm q/2^{d_v+1}$ as in Lemma 1, at a measured mean of 5.0 decapsulations per read for ML-KEM-512 and -768 ($d_v = 4$) and 7.0 for ML-KEM-1024 ($d_v = 5$). Our binary search is deliberately unoptimized: the adversary knows $(\mathbf{t}^\top \mathbf{r})_i$ exactly and the signal has $\text{std} \approx 50$, so the flip level is predictable a priori to within ≈ 1.4 ($d_v = 4$) to 2.8 ($d_v = 5$) grid steps, and querying only the one or two boundary levels nearest the prediction brings the read to ≈ 2 decapsulations. The reported query counts (Table 4) would fall by a further factor ≈ 2.5 – 3.5 under this optimization; we report the unoptimized figure our harness measures. We verified that this observation is the true decapsulation signal and carries no modeling gap: the harness’s per-coordinate accept/reject bit matches kyber-py’s `_k_pke_decrypt` accept/reject bit-for-bit over every grid level on sampled sessions, 480/480, 480/480 and 960/960 agreements for ML-KEM-512, -768 and -1024. We accumulate the normal equations over N messages and solve by least squares in double-precision floating point, then refine to integers by coordinate descent in the box $[-\eta_1, \eta_1]$; the public-key identity $\mathbf{e} = \mathbf{t} - \mathbf{A}\mathbf{s}$ (short iff $\mathbf{s}$ is correct) is the acceptance certificate, computed inside the loop rather than as a post-hoc test. The reported counts use this L_2 coordinate descent; the artifact also provides an L_∞ cell-feasibility variant (which avoids the rare case where an L_2 step turns a correct coordinate wrong, the true key not being the L_2 minimizer among integer neighbours), and it does not change the counts, refinement never being the bottleneck. The secret key $\mathbf{s}$ is recovered directly, so we report recovery of $\mathbf{s}$. Because each observation is a short random combination of the secret whose error is comparable to its own size (standard deviation ≈ 47 against a ± 104 window at $d_v = 4$), the design conditioning, not the negacyclic structure, governs the session count; we therefore also run the equivalent independent short-coefficient model, which agrees with the full kyber-py harness where the two overlap and lets us sweep more keys and coverages. We assume the adversary knows which coordinate is unverified; if not, it is found by at most 256 trial mutations (one per coordinate), negligible against the session counts. The harness and logs are the reproducibility artifact (Section 11).

7.2 Recovery from a single unverified coordinate

Table 4 reports recovery at $|T| = 1$. “Sessions to full recovery” is the smallest N (swept in steps of 5000–15000) at which the success criterion holds: all kn coefficients of $\mathbf{s}$ match the ground truth and $\mathbf{e} = \mathbf{t} - \mathbf{A}\mathbf{s}$ is short. For all three parameter sets least squares recovers the entire secret key: about 75,000 sessions for ML-KEM-512, 90,000 for -768, and 28,000 for -1024 on the interface harness (one key per set). These single-key figures sit at the low end of a $\approx 1.5\times$ key-to-key spread: over five random keys the equivalent short-coefficient model recovers ML-KEM-512 between 85,000 and 110,000 sessions (median $\approx 110,000$), so the one-key counts should be read as optimistic-end rather than typical. The recovery is

Table 3: Raw information floor for the $|T| = 1$ recovery. $H(\mathbf{s}, \mathbf{e})$ is the secret’s entropy (per-coordinate CBD entropy times $2kn$); I is the information one read conveys, the expected entropy of the located width- Δ cell index of the signal (per-set standard deviation 51, 47, 48) averaged over the uniform in-cell offset, an upper bound on the bits per read, so H/I is a genuine lower bound on the reads needed (computed in `constants.py`). It is far looser than the identifiability bound of Theorem 1; the measured sessions of Table 4 are the last column’s multiple of it.

Parameter set	$2kn$	$H(\mathbf{s}, \mathbf{e})$	I (bits)	Floor H/I	Measured/floor
ML-KEM-512	1024	≈ 2390	0.64	≈ 3700	$\approx 20\times$
ML-KEM-768	1536	≈ 3120	0.59	≈ 5300	$\approx 17\times$
ML-KEM-1024	2048	≈ 4160	1.16	≈ 3600	$\approx 8\times$

Table 4: Full key recovery from a single unverified coordinate ($|T| = 1$), run end to end against the `kyber-py` decapsulation `_k_pke_decrypt`, one key per parameter set. The secret $\mathbf{s}$ is recovered (all kn coefficients) and $\mathbf{e}$ follows from and is checked against $\mathbf{e} = \mathbf{t} - \mathbf{A}\mathbf{s}$. Each read is a binary search over the 2^{d_v} grid costing the listed mean decapsulations; “queries” is their product with the session count. These one-key figures are at the low end of a $\approx 1.5\times$ key-to-key spread (for ML-KEM-512 the equivalent short-coefficient model recovers between 85,000 and 110,000 sessions over five keys); the model also supplies the coverage sweep of Fig. 1.

Parameter set	$2kn$	d_v	Sessions	Decaps/read	Queries
ML-KEM-512	1024	4	75,000	5.0	3.8×10^5
ML-KEM-768	1536	4	90,000	5.0	4.5×10^5
ML-KEM-1024	2048	5	28,000	7.0	2.0×10^5

gradual, as the wrong-coordinate count after rounding falls with N ; for one ML-KEM-512 key the model gives

$$N : 30k, 45k, 60k, 75k, 85k, 95k \quad \mapsto \quad \text{wrong} : 46, 26, 5, 1, 1, 1,$$

the last coordinate the slowest. This single-key curve shows the gradual convergence; the evidence for the $(\Delta/\sigma_w)^2$ law itself is the three-set fit (Section 7). ML-KEM-1024 is cheapest because its finer grid ($d_v = 5$) halves the read noise. The counts are large because each read carries well under one bit about the secret (a $\pm q/2^{d_v+1}$ window on a quantity of standard deviation ≈ 47). The cross-set costs follow the least-squares law $R \approx 2.9(\Delta/\sigma_w)^2$ derived under Predicted versus measured below; against the far looser raw information floor $H(\mathbf{s}, \mathbf{e})/I$ (Table 3) the counts sit roughly an order of magnitude high, the gap being the inefficiency of an L_2 solver on an L_∞ problem rather than a shortfall of information. Our 10^5 counts should therefore be read as a *loose upper bound*: the belief-propagation and lattice solvers of [HPP21, HMS⁺23, Del22, PP21], which on this exact inequality system usually land within a small factor of the information floor, would likely bring the practical cost near 10^4 sessions; we did not run one, so we report the least-squares figure our harness measures and flag the gap. This is the same regime as [Das26], whose $|T| \approx 51$ recovery needed 10^5 – 10^6 queries; what it left unmeasured is the $|T| = 1$ extreme, which Table 4 supplies.

7.3 Coverage sets the number of sessions

Figure 1 plots sessions to full recovery against coverage $|T|$ at ML-KEM-1024. Each session yields $|T|$ observations of the secret, so the session count follows $R_1/|T|$, where

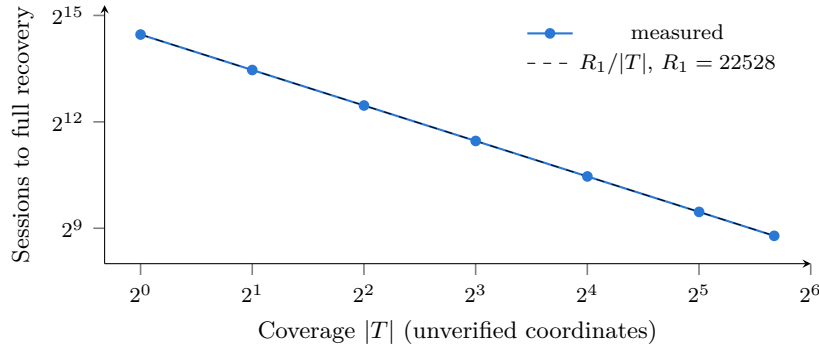


Figure 1: Sessions to full-key recovery against coverage $|T|$ at ML-KEM-1024, on the short-coefficient model (one key). The count tracks $R_1/|T|$ (R_1 the measured $|T| = 1$ count) across $|T| \in \{1, \dots, 51\}$ and recovery succeeds at every $|T|$. The $|T| \approx 51$ end is the wolfSSL AVX2 coverage of [Das26], the $|T| = 1$ end a single unverified coordinate. Absolute counts are per key and key-dependent (a $\approx 1.5\times$ spread, Section 7): this single-key $|T| = 1$ knee is 22,528, of the same order as the 28,000 measured against the `kyber-py` decapsulation at $|T| = 1$ (Table 4); the point of the figure is the $R_1/|T|$ shape, not the absolute level.

R_1 is the measured $|T| = 1$ count (the absolute level is set by the $(\Delta/\sigma_w)^2$ cost law of Section 7, not by $2kn$), from tens of thousands at $|T| = 1$ down to a few hundred at $|T| = 51$ (the wolfSSL AVX2 coverage of [Das26]). Coverage sets how many messages the adversary needs, never whether recovery succeeds: every coverage down to one recovers the full key.

7.4 Predicted versus measured

Theorem 1 gives an order bound on the identifiability threshold, $R = O(kn \log kn)$; it is sufficient and does not pin down the constant, so we do not use it to predict the measured counts. What captures the cross-set costs is a least-squares heuristic. Least squares on a design of per-entry variance σ_w^2 with bounded read noise of variance $\Delta^2/12$ gives per-coordinate estimation variance $\approx (\Delta^2/12)/(R\sigma_w^2)$; exact rounding of all $2kn$ coordinates needs this below a fixed constant, so $R \approx \kappa(\Delta/\sigma_w)^2$. Here σ_w is the *root-mean-square* per-entry standard deviation of $\mathbf{w}$ ($\approx 1.30, 1.20, 1.05$; note this is the average-over-blocks quantity, distinct from the block-minimum $\sigma_w^{\min}$ that controls the worst-case candidate in Theorem 1). This is a heuristic (an L_2 estimate on an L_∞ rounding problem), not proved. On the interface single-key counts $\kappa \approx 2.9$ (2.93, 2.99, 2.87; `constants.py`); those keys are low-end, and over five keys the model gives ML-KEM-512 at 85,000–110,000, i.e. $\kappa \approx 3.3$ –4.3. So κ is of order 3 with a $\approx 1.5\times$ key-to-key spread; both the single-key and five-key runs use the least-squares solver with the coordinate-descent refinement of Section 7. The $(\Delta/\sigma_w)^2$ law, not $2kn$, explains the cross-set costs: ML-KEM-1024 is cheapest because its finer grid halves Δ ; ML-KEM-768 costs only $1.2\times$ ML-KEM-512 despite $1.5\times$ the dimension, because kn enters only through the weak log quantile. Against the looser raw information floor $H(\mathbf{s}, \mathbf{e})/I$ (Table 3) the counts sit about an order of magnitude high, the price of an L_2 solver on an L_∞ problem. One consequence, immediate once sessions are multiplied by $|T|$ reads each, is that the attacker’s true cost is *coverage-independent in queries*, even though sessions fall as $R_1/|T|$. On the single-key ML-KEM-1024 model (Fig. 1), $|T| = 1$ costs $22,528 \times 7 \approx 1.58 \times 10^5$ decapsulations and $|T| = 51$ costs $441 \times 51 \times 7 \approx 1.58 \times 10^5$ — the same, so one unverified coordinate costs what fifty-one do (consistent with the 10^5 – 10^6 of [Das26]). Finally the cmov fault is the separate

full-coverage case: a complete plaintext-checking oracle recovering the key by coefficient isolation in $\Theta(kn)$ queries (Table 2).

7.5 Limitations

The recoveries of Table 4 are run end to end against the `kyber-py` decapsulation for one key per parameter set: each observation is `kyber-py`'s `_k_pke_decrypt` accept/reject bit $[\text{Dec}(\text{dk}, c') = m_0]$ for a ciphertext whose unverified coordinate has been overwritten, located by binary search over the grid, so the recovery consumes the real decapsulation signal and not a model of it, and the session and query counts are measured. Only the session count is key-dependent (we report one key and seed per set); the per-read decapsulation count and the bit-for-bit agreement of the read with `_k_pke_decrypt` are structural and hold across keys. The equivalent independent short-coefficient model, which reproduces the same short coefficients and $\pm q/2^{d_v+1}$ grid noise, tracks the interface harness and, being far faster, lets us average over keys (a $\approx 1.5\times$ spread, Section 7) and supply the coverage sweep of Fig. 1. What we do not do is inject the fault into a physical device: a single-unverified-coordinate binary is deterministic (Remark 1), so grid quantization is the only error it exhibits and the interface-level oracle is faithful in kind, but a hardware campaign could add transient noise we do not model. We do not prove a closed-form recovery threshold, characterizing the cost empirically against the windowed bound of Theorem 1. We report *unoptimized* figures throughout: the three optimizations we note — reading each coordinate with ≈ 2 decapsulations instead of a full binary search, using a belief-propagation/lattice solver in place of least squares, and isolating `cmov` coefficients by candidate-set rather than grid bisection — would together cut the query count by roughly an order of magnitude, and we have run none of them. Finally, Lemma 1 treats a fault leaving unverified coordinates of the v component, and the u -component case is out of scope and an open direction. It is *not* a coefficient isolation: shifting one $u'_{t,j}$ perturbs the decryption noise at *all* 256 message coordinates at once (by $-\mathbf{s}_t[(i-j) \bmod n]$ times the shift, with the negacyclic sign), and the incomplete comparison accepts only if no message bit flips, so a sweep observes the *first* flip — a minimum over the coordinates where $|\mathbf{s}_t|$ is largest (typically η_1), an order-statistic observation rather than one clean linear read. Coefficient isolation in Section 6 works only because there the attacker crafts $\mathbf{u}'$ as a single monomial with the other coordinates zeroed, a freedom the valid-ciphertext setting removes. Whether this first-flip observation recovers the key (likely, with a purpose-built solver) we leave open. Consequently, whether the unverified NEON coordinates of CVE-2026-6330 lie in $c_1(u)$ or $c_2(v)$ — not stated in the public advisory — matters: full recovery if in v (our setting), open if in u . Transient fault-injection campaigns are also out of scope.

8 Structural versus stochastic faults

Our incomplete-comparison recovery and the stochastic noisy-oracle attack of Hermelink et al. [HMT25] are one statistical problem — recover short $(\mathbf{s}, \mathbf{e})$ from many bounded-error linear observations that the FO randomness makes independent across messages — differing only in the error source: compression *quantization* to a $\pm q/2^{d_v+1}$ window for us, measurement *noise* of accuracy α for them, contributing $I \approx 0.6\text{--}1.2$ and $1 - h(\alpha)$ bits per observation respectively. Determinism (Remark 1) changes only whether repeating one probe helps, not the accumulation across messages that recovers the key. The practical consequence is that a defense hardening only against noise (masking, redundancy) does nothing against a coverage gap, and one covering only positions does nothing against a noisy but complete oracle: what must be protected is the *completeness* of the rejection check, since the FO randomness turns even one leaked observation into full recovery.

9 Conclusion

We showed that any incomplete Fujisaki–Okamoto ciphertext comparison in ML-KEM leaving even a single unverified coordinate is a full key-recovery bug, demonstrated end to end at $|T| = 1$ on all three parameter sets (against a simulated incomplete comparison on `kyber-py`) in a few times 10^5 queries, with the skipped conditional move a complete plaintext-checking oracle at $\Theta(kn)$ queries. What is new is quantitative: the cost obeys $R \approx 2.9(\Delta/\sigma_w)^2$, so the attacker’s query cost is coverage-independent (one unverified coordinate costs what fifty do), and ML-KEM-1024 — despite its larger dimension — is the *cheapest* set, its finer compression grid outweighing kn .

The practical consequence is a single testable requirement: decapsulation must reject every ciphertext its re-encryption does not reproduce. Re-encryption comparison vectors exercising this check across all three ML-KEM parameter sets are now part of Project Wycheproof [Pro26]. Open directions include a matching upper bound on the session count, an extension to other re-encryption-based transforms, and running a belief-propagation solver to close the order-of-magnitude gap to the information floor.

10 Disclosure

The three faults studied here were all disclosed in coordination with the relevant maintainers before publication: the wolfSSL AVX2 and NEON comparison flaws were reported and fixed upstream (CVE-2026-10097, CVE-2026-6330), and the `pqc_kyber/cosmian_kyber` skipped conditional move was flagged through RustSec (RUSTSEC-2026-0290, -0288) as those crates are unmaintained. We report on already-public, fixed-or-flagged defects; this work introduces no new vulnerability and targets no deployed key.

11 Artifact availability

The key-recovery proof of concept, the induced decapsulation oracle, the model-experiment scripts, and full reproduction instructions are available at <https://github.com/007bsd/mlkem-cmov-key-recovery> (pinned to the tagged release corresponding to this version).

Acknowledgements

The write-up was prepared with AI writing assistance (Anthropic’s Claude). The model, the theorems and their proofs, the experiments, and all results and claims are the author’s own and were verified by the author.

A Windowed identifiability

This appendix proves Theorem 1 from the width- Δ cells the oracle returns, not from a noiseless system, at $|T| = 1$ (so $R = N$ and the rows are one per message, independent). Write $x = (\mathbf{s}, \mathbf{e})$ and $\Delta = q/2^{d_v}$. Each accepted message m yields the short vector $\mathbf{w}_j := \mathbf{w}_i(m)$ of Lemma 1 and locates $y_j := \langle x, \mathbf{w}_j \rangle$ to its width- Δ cell.

Let $\mathbf{d} = x' - x \neq 0$ be the difference of two admissible short keys, so $\mathbf{d}$ has integer entries with $\|\mathbf{d}\|_\infty \leq 2\eta_1$, support $h = |\text{supp}(\mathbf{d})|$ and $\|\mathbf{d}\|_2 \geq \sqrt{h}$. The candidate x' is consistent with observation j iff $Y_j := y_j + \langle \mathbf{d}, \mathbf{w}_j \rangle$ lies in the same cell as y_j . We take Assumption 1 in its quantitative form: the pair is within statistical distance $\varepsilon = \text{negl}(kn)$ of (uniform) $\times$ (marginal), so writing $p(\mathbf{d}) := \Pr[x' \text{ consistent}] = \mathbb{E}_{\mathbf{w}}[\max(0, 1 - |\langle \mathbf{d}, \mathbf{w} \rangle|/\Delta)]$, each per-message factor is correct up to ε ; by independence over the $R = N$ messages the survival

probability is $(p(\mathbf{d}) + \varepsilon)^R$, and $\varepsilon R \rightarrow 0$ leaves the bounds below intact. (A merely $o(1)$ distance would only give a $1 - o(1)$ conclusion; the negligible form is what supports $1 - \text{negl}$, and we do not prove it.) Asymptotics are $kn \rightarrow \infty$, q, d_v, η fixed. Write $Y := \langle \mathbf{d}, \mathbf{w} \rangle$, with $\mathbb{E}Y^2 = \sigma^2 \|\mathbf{d}\|_2^2$.

The key constant is honest for *every* $\mathbf{d}$, including sparse: by Hölder $\mathbb{E}|Y| \geq (\mathbb{E}Y^2)^{3/2}/(\mathbb{E}Y^4)^{1/2}$, and the CBD and decompression-error entries are platykurtic ($\mathbb{E}X^4 \leq 3(\mathbb{E}X^2)^2$: e.g. CBD(2) has kurtosis $2.5 < 3$, CBD(3) has $6 < 6.75$), so $\mathbb{E}Y^4 \leq 3(\mathbb{E}Y^2)^2$ and $\mathbb{E}|Y| \geq \sigma_{\min} \|\mathbf{d}\|_2 / \sqrt{3} =: c \|\mathbf{d}\|_2$ ($\sigma_{\min}$ the block-minimum per-entry std). This replaces the Gaussian value $\sigma \sqrt{2/\pi}$, which is false for small support ($h = 1$: CBD(2) gives $0.75 < 0.798$).

Write $\sigma_Y = \sqrt{\mathbb{E}Y^2} = \sigma_d \|\mathbf{d}\|_2$ ($\sigma_d \geq \sigma_{\min}$ the per-entry std on $\mathbf{d}$'s support). We bound $p(\mathbf{d})$ in three regimes that cover all $\mathbf{d}$ with no gap.

Small ($\sigma_Y \leq \frac{3}{10}\Delta$): the tail correction is $\mathbb{E}[(|Y| - \Delta)_+] \leq \mathbb{E}Y^2/\Delta = \sigma_Y^2/\Delta \leq \frac{3}{10}\sigma_Y$, a constant fraction of $\mathbb{E}|Y| \geq c\|\mathbf{d}\|_2$ (as $\sigma_Y = \sigma_d \|\mathbf{d}\|_2$ and $c = \sigma_{\min}/\sqrt{3}$), so $\mathbb{E}[\min(1, |Y|/\Delta)] = \mathbb{E}|Y|/\Delta - \mathbb{E}[(|Y| - \Delta)_+]/\Delta \geq c'\|\mathbf{d}\|_2/\Delta$ for a constant $c' > 0$, giving $p(\mathbf{d})^R \leq e^{-Rc'\sqrt{h}/\Delta}$.

Middle ($\frac{3}{10}\Delta \leq \sigma_Y \leq \sqrt{3}\Delta$): using $\mathbb{E}[\min(1, |Y|/\Delta)] \geq (\tau/\Delta) \Pr[|Y| > \tau]$ with $\tau = \sigma_Y/2$ and Paley–Zygmund at $\theta = \frac{1}{4}$ ($\Pr[|Y| > \sigma_Y/2] \geq (\frac{3}{4})^2/3 = 0.1875$, from $\mathbb{E}Y^4 \leq 3(\mathbb{E}Y^2)^2$) gives $p(\mathbf{d}) \leq 1 - 0.094\sigma_Y/\Delta \leq \beta_1$ with $\beta_1 \leq 0.972$ at the bottom of the regime.

Large ($\sigma_Y \geq \sqrt{3}\Delta$): Paley–Zygmund at $\theta = \frac{1}{3}$ gives $\Pr[|Y| > \Delta] \geq (\frac{2}{3})^2/3 = 0.148$, so $p(\mathbf{d}) \leq \beta_0 = 0.852$, at *every* support h (no Berry–Esseen requirement). The small and middle regimes meet at $\sigma_Y = \frac{3}{10}\Delta$ and the middle and large at $\sigma_Y = \sqrt{3}\Delta$, so there is no gap.

Union-bounding over the $\leq (8\eta_1 kn)^h$ candidates of support h (the $h = 2kn$ term dominating), each regime contributes $O(kn \log kn)$, so $R^* = O(kn \log kn)$ and Theorem 1 follows. The binding regime is the *middle* one, whose worst constant $\beta_1 \approx 0.97$ gives $R^* \approx 2kn \ln(8\eta_1 kn)/\ln(1/\beta_1) \approx 3.4, 5.1, 6.9 \times 10^5$ for ML-KEM-512, -768, -1024 (`constants.py`) — roughly 4 to 25 times the measured counts. This is a loose but not astronomical sufficient bound; a Gaussian approximation of the reads (replacing Paley–Zygmund by the normal tail) tightens it to $\approx 2 \times 10^5$, and the tight *measured* law is $R \approx 2.9(\Delta/\sigma_w)^2$ (Section 7), which we do not derive here.

Lemma 2 (Full-rank design). *The least-squares estimator of Empirical Result 1 is well posed. The rows $\mathbf{w}_j$ are i.i.d. across messages (fresh FO randomness) with second-moment $\Sigma = \mathbb{E}[\mathbf{w}\mathbf{w}^\top]$, and $\Sigma \succ 0$ needs no independence assumption. For any $\mathbf{a} \neq 0$, condition on $\mathbf{r}$ and on the public matrix: the $\mathbf{e}$ -block of $\mathbf{w}$ is then fixed, while the $\mathbf{s}$ -block still varies with the fresh $\mathbf{e}_1$ (of positive variance, and Δ_u is determined by $\mathbf{r}, \mathbf{e}_1$), so $\text{Var}\langle \mathbf{a}, \mathbf{w} \rangle \geq \mathbb{E}_{\mathbf{r}}[\text{Var}\langle \mathbf{a}, \mathbf{w} \rangle \mid \mathbf{r}] > 0$ unless $\mathbf{a}$ vanishes on the $\mathbf{s}$ -block, and the symmetric argument (conditioning on $\mathbf{e}_1$) closes the $\mathbf{e}$ -block; hence $\mathbf{a}^\top \Sigma \mathbf{a} > 0$. Hence by a matrix Bernstein inequality [Tro15] the Gram matrix $M^\top M$ is nonsingular, with $\lambda_{\min}(M^\top M) = \Omega(R \lambda_{\min}(\Sigma))$, once $R = \Omega(2kn \log 2kn)$; this guarantees a unique least-squares solution and is not used for identifiability, which is the windowed argument above.*

References

- [BDH⁺21] Shivam Bhasin, Jan-Pieter D’Anvers, Daniel Heinz, Thomas Pöppelmann, and Michiel Van Beirendonck. Attacking and defending masked polynomial comparison for lattice-based cryptography. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2021(3):334–359, 2021.
- [BDHD⁺19] Ciprian Băetu, F. Betül Durak, Loïs Huguenin-Dumittan, Abdullah Talayhan, and Serge Vaudenay. Misuse attacks on post-quantum cryptosystems. Cryptology ePrint Archive, Paper 2019/525, 2019. EUROCRYPT 2019.

- [BFK⁺12] Romain Bardou, Riccardo Focardi, Yusuke Kawamoto, Lorenzo Simionato, Graham Steel, and Joe-Kai Tsay. Efficient padding oracle attacks on cryptographic hardware. Cryptology ePrint Archive, Paper 2012/417, 2012. CRYPTO 2012.
- [BHH⁺19] Nina Bindel, Mike Hamburg, Kathrin Hövelmanns, Andreas Hülsing, and Edoardo Persichetti. Tighter proofs of CCA security in the QROM. Cryptology ePrint Archive, Paper 2019/590, 2019. TCC 2019.
- [Car26] Nicholas Carlini. Incomplete ML-KEM ciphertext comparison in wolfSSL ARM64 NEON. CVE-2026-6330, <https://www.cve.org/CVERecord?id=CVE-2026-6330>, 2026. Reported by N. Carlini.
- [Das26] Bhabani Sankar Das. Incomplete ciphertext comparison in ML-KEM: From an IND-CCA2 break to key recovery. Cryptology ePrint Archive, Paper 2026/1682, 2026.
- [Del22] Jeroen Delvaux. Roulette: A diverse family of feasible fault attacks on masked Kyber. Cryptology ePrint Archive, Paper 2021/1622, 2022. TCHES 2022(4).
- [Del26] José Luis Delgado Jiménez. What can verifiable decapsulation tests certify? pass bounds and fault-recognition limits for FO-based KEMs. arXiv:2606.04443, 2026.
- [DHP⁺22] Jan-Pieter D’Anvers, Daniel Heinz, Peter Pessl, Michiel Van Beirendonck, and Ingrid Verbauwhede. Higher-order masked ciphertext comparison for lattice-based cryptography. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2022(2):115–139, 2022.
- [Flu16] Scott Fluhrer. Cryptanalysis of ring-LWE based key exchange with key share reuse. Cryptology ePrint Archive, Paper 2016/085, 2016.
- [GGHS25] Lewis Glabush, Felix Günther, Kathrin Hövelmanns, and Douglas Stebila. Verifiable decapsulation: Recognizing faulty implementations of post-quantum KEMs. Cryptology ePrint Archive, Paper 2025/450, 2025. CRYPTO 2025.
- [GJN20] Qian Guo, Thomas Johansson, and Alexander Nilsson. A key-recovery timing attack on post-quantum primitives using the Fujisaki–Okamoto transformation and its application on FrodoKEM. In *CRYPTO 2020*, 2020. Cryptology ePrint Archive, Paper 2020/743.
- [HHK17] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki–Okamoto transformation. Cryptology ePrint Archive, Paper 2017/604, 2017. TCC 2017.
- [HHM22] Kathrin Hövelmanns, Andreas Hülsing, and Christian Majenz. Failing gracefully: Decryption failures and the Fujisaki–Okamoto transform. Cryptology ePrint Archive, Paper 2022/365, 2022. ASIACRYPT 2022.
- [HMS⁺23] Julius Hermelink, Erik Mårtensson, Simona Samardjiska, Peter Pessl, and Gabi Dreier Rodosek. Belief propagation meets lattice reduction: Security estimates for error-tolerant key recovery from decryption errors. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2023(4):287–317, 2023.

- [HMT25] Julius Hermelink, Erik Mårtensson, and Maggie Tran. Noise-tolerant plaintext-checking oracle attacks: A soft-analytic approach applied to ML-KEM. Cryptology ePrint Archive, Paper 2025/1496, 2025.
- [HPP21] Julius Hermelink, Peter Pessl, and Thomas Pöppelmann. Fault-enabled chosen-ciphertext attacks on Kyber. Cryptology ePrint Archive, Paper 2021/1222, 2021. INDOCRYPT 2021.
- [Nat24] National Institute of Standards and Technology. Module-lattice-based key-encapsulation mechanism standard. FIPS 203, 2024.
- [Ope25] Open Quantum Safe. Theoretical design flaw in HQC implicit rejection (CVE-2025-48946). liboqs security advisory GHSA-3rxw-4v8q-9gq5, 2025. Reported by M.-J. O. Saarinen (NIST pqc-forum); analysis by D. Connolly.
- [PP21] Peter Pessl and Lukas Prokop. Fault attacks on CCA-secure lattice KEMs. Cryptology ePrint Archive, Paper 2021/064, 2021. TCHES 2021(2).
- [Pro26] Project Wycheproof. Re-encryption comparison test vectors for ML-KEM. C2SP/wycheproof, <https://github.com/C2SP/wycheproof/pull/274>, 2026. Merged pull request 274, contributed by the present author.
- [QCZ⁺21] Yue Qin, Chi Cheng, Xiaohan Zhang, Yanbin Pan, Lei Hu, and Jintai Ding. A systematic approach and analysis of key mismatch attacks on lattice-based NIST candidate KEMs. Cryptology ePrint Archive, Paper 2021/123, 2021. ASIACRYPT 2021.
- [RRCB19] Prasanna Ravi, Sujoy Sinha Roy, Anupam Chattopadhyay, and Shivam Bhasin. Generic side-channel attacks on CCA-secure lattice-based PKE and KEMs. Cryptology ePrint Archive, Paper 2019/948, 2019. TCHES 2020(3).
- [RRD⁺23] Gokulnath Rajendran, Prasanna Ravi, Jan-Pieter D’Anvers, Shivam Bhasin, and Anupam Chattopadhyay. Pushing the limits of generic side-channel attacks on LWE-based KEMs – parallel PC oracle attacks on Kyber KEM and beyond. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2023(2):418–446, 2023.
- [Tro15] Joel A. Tropp. *An Introduction to Matrix Concentration Inequalities*, volume 8. Foundations and Trends in Machine Learning, 2015.
- [UXT⁺22] Rei Ueno, Keita Xagawa, Yutaro Tanaka, Akira Ito, Junko Takahashi, and Naofumi Homma. Curse of re-encryption: A generic power/em analysis on post-quantum KEMs. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2022(1):296–322, 2022.
- [XIU⁺21] Keita Xagawa, Akira Ito, Rei Ueno, Junko Takahashi, and Naofumi Homma. Fault-injection attacks against NIST’s post-quantum cryptography round 3 KEM candidates. Cryptology ePrint Archive, Paper 2021/840, 2021. ASIACRYPT 2021.