LaZer

Lattice-Based Zero-Knowledge Arguments for Lattice Relations

Patrick Steuer [1]

on behalf of the LaZer team:
Vadim Lyubashevsky [1], Ngoc Khanh Nguyen [2], Michał Osadnik [3], Gregor Seiler [1]

NIST Threshold Call – Preview Talks Round 2 · TCPT2 – 2026-07-08

Category **S6** · ePrint 2024/1846 · published at ACM CCS 2024

[1]IBM Research – Zurich [2]King's College London [3]Aalto University

What is LaZer?

LaZer (CCS'24)

A **library** for zero-knowledge and succinct proofs of statements about short secrets in module lattices.

- **Two proof systems:** LNP'22 — *linear-size* zero-knowledge, small for a single small- to medium-sized statement; LaBRADOR'23 — *succinct* (no ZK), short even for large statements.
- **Lineage:** LNP'22 + LaBRADOR'23 → LaZer'24.
- **Assumptions:** only M-SIS, M-LWE — transparent, no trusted setup, plausibly post-quantum.

Why prove lattice relations?

- Post-quantum crypto lives in module-ring arithmetic: **ML-KEM**, **ML-DSA**, **FN-DSA**, lattice **FHE** (e.g. TFHE).
- Threshold protocols repeatedly need to *prove* statements about lattice objects: “my key/share is well-formed”, “this ciphertext is valid”.
- Often this is just linear equations over the ring + a norm bound.

Circuit vs. native

Arithmetize → generic ZK

- re-encode norm bounds as a circuit
- re-encode equations as a circuit
- additional code to handle arithmetization

Prove natively (LaZer)

- re-encode norm check as equation
- lift* equations and aggregate
- reuse the application's ring arithmetic

* from application to proof-system ring

LNP interface

On a committed witness $(\mathbf{s}_1, \mathbf{m})$, $\mathbf{s}_1$ short — prove polynomial equations, plus a *separate norm constraint per witness part*:

equations

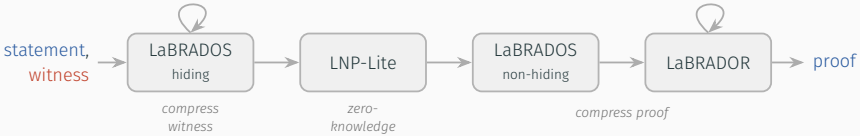
- **quadratic relations over** R_q — inner products $\sum \langle s_i, s_j \rangle + \cdots = 0$
- **quadratic relations over** $\mathbb{Z}_q$ — the same on *constant coefficients*

norms

- **binaryness** — coefficients in $\{0, 1\}$, used to prove exact ℓ_∞ norm
- **exact** ℓ_2 **norm** — $\|s\|_2 \leq B$
- **approximate norm** — $\|s\| \leq \psi B'$, slack $\psi \geq 1$ (approximate range proof – ARP)

Making LNP ZK proofs succinct

For *large* statements, wrap the linear-size proof so it stays short — proof size $<$ **100 KB** for arbitrarily large statements.



- LaBRADOS* (hiding) **compresses the witness**, recursively.
- LNP-Lite — LNP tailored to LaBRADOS outputs — adds **zero-knowledge**.
- LaBRADOS/LaBRADOR (non-hiding) **compress the proof**, recursively.

Zero-knowledge costs ∼**10 KB** on a ∼100 KB proof. *New work, in submission — extends* LaZer.

* a version of LaBRADOR with compressing outer commitments, hiding prover messages.

Succinct example: aggregate signatures

Combine **many** signatures into **one** short proof that every message was signed.

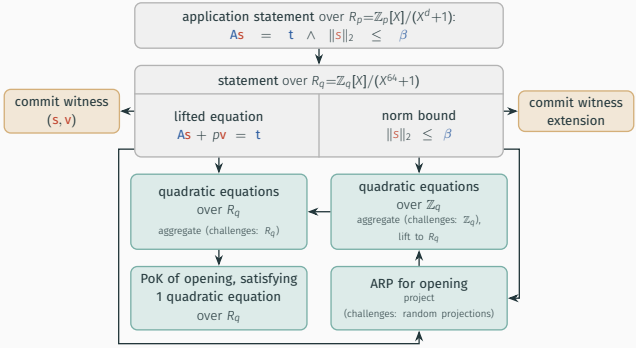
Per FALCON-512 signature i

know short $s_{1,i}, s_{2,i}$ with $a_i \cdot s_{2,i} + s_{1,i} = t_i \pmod{p}$, $p = 12289$, $d = 512$ — lifted with carry v_i, then aggregated over all i.

- **1024** signatures $\rightarrow$ $\sim$**73 KB** proof, prover **0.6 s**.
- **100 000** signatures $\rightarrow$ $\sim$**73 KB** proof, prover **65 s**.

LNP technical overview

How to prove knowledge of committed short vectors satisfying linear equations + an ℓ_2 norm bound.



Fiat–Shamir $\rightarrow$ non-interactive.

Proof of a well-formed ML-KEM-1024 key

ML-KEM (Kyber), FIPS 203.

ML-KEM-1024 key generation

$R_p = \mathbb{Z}_p[X]/(X^d+1)$, $d = 256$, $p = 3329$, $k = 4$.
given public $\mathbf{A} \in R_p^{k\times k}$ (uniform, from a seed ρ),
sample $\mathbf{s}, \mathbf{e} \leftarrow \mathrm{CBD}_2^k$ (coeffs in $[-2,2]$, $\Pr[0] = \frac{3}{8}$, $\Pr[\pm1] = \frac{1}{4}$, $\Pr[\pm2] = \frac{1}{16}$),
compute the public key

$$\mathbf{t} = \mathbf{A}\,\mathbf{s} + \mathbf{e} \in R_p^k.$$

CBD = centred binomial.

statement

given $\mathbf{A}$ and $\mathbf{t}$, the prover **knows** $(\mathbf{s}, \mathbf{e})$ with

$$\|(s \mid e)\|_2 \le B, \qquad \mathbf{A}\,\mathbf{s} + \mathbf{e} = \mathbf{t}.$$

$B = 1.2\sqrt{2048} \approx 54$ (1.2× the expected ℓ_2 of (s, e)).

Proof of a well-formed ML-KEM-1024 key

equation

- R_p equation, lifted with carry $\mathbf{v}$ — **linear equation over** R_q: $\mathbf{A}\,\mathbf{s} + \mathbf{e} = \mathbf{t} + p\,\mathbf{v}$.

norm bounds

- $\mathbf{s}, \mathbf{e}$ — **exact** ℓ_2 bound $\|(\mathbf{s} \mid \mathbf{e})\|_2 \leq B$.
- the lift's carry $\mathbf{v}$ — **ARP** (no-wrap).

Proof of a well-formed Swoosh key

Swoosh: lattice NIKE (ePrint **2023/271**). *Active* security needs each party to prove its public key is an honest M-LWE sample.

M-LWE key generation

$R_p = \mathbb{Z}_p[X]/(X^d+1)$, $d = 256$, $p = 2^{214} - 255$, $N = 32$.
given public $\mathbf{A} \in R_p^{N \times N}$ (uniform, invertible),
sample $\mathbf{s}, \mathbf{e} \leftarrow \mathrm{CBD}_1^N$ ternary, $\Pr[\pm 1] = \frac{1}{4}$, $\Pr[0] = \frac{1}{2}$,
compute the public key

$$\mathbf{b} = \mathbf{A}\,\mathbf{s} + \mathbf{e} \in R_p^N.$$

CBD = centred binomial.

statement

given $\mathbf{A}$ and $\mathbf{b}$, the prover **knows** $(\mathbf{s}, \mathbf{e})$ with

$$\|(s \mid e)\|_2 \le B, \qquad \mathbf{A}\,\mathbf{s} + \mathbf{e} = \mathbf{b}.$$

Proof of a well-formed Swoosh key

equation

- R_p equation, lifted with carry $\mathbf{v}$ — **linear equation over** R_q: $\mathbf{A}\,\mathbf{s} + \mathbf{e} = \mathbf{b} + p\,\mathbf{v}$.

norm bounds

- $\mathbf{s}, \mathbf{e}$ — **exact** ℓ_2 bound $\|(s \mid e)\|_2 \leq B$.
- the lift's carry $\mathbf{v}$ — **ARP** (no-wrap).

Planned improvement: the Swoosh modulus $p \approx 2^{214}$ is large, enable commitments over a *smaller* modulus to reduce proof size.

Proof of a well-formed ML-KEM-1024 ciphertext

ML-KEM (Kyber) encryption, FIPS 203.

ML-KEM-1024 encryption

$R_p = \mathbb{Z}_p[X]/(X^d+1)$, $d = 256$, $p = 3329$, $k = 4$, $(d_u, d_v) = (11, 5)$.
given pk (A, t) and a message $\mathsf{m} \in \{0,1\}^{256}$,
sample $\mathsf{r}, \mathsf{e}_1 \leftarrow \mathrm{CBD}_2^k$, $\mathsf{e}_2 \leftarrow \mathrm{CBD}_2$,
compute

$$\mathsf{u} = \mathsf{A}^\top \mathsf{r} + \mathsf{e}_1, \qquad \mathsf{v} = \langle \mathsf{t}, \mathsf{r} \rangle + \mathsf{e}_2 + \lceil p/2 \rceil\, \mathsf{m},$$

ciphertext $(\mathsf{c}_1, \mathsf{c}_2) = \big(\mathrm{Compress}_{11}(\mathsf{u}),\ \mathrm{Compress}_5(\mathsf{v})\big)$.

statement

given pk and $(\mathsf{c}_1, \mathsf{c}_2)$, the prover **knows** $(\mathsf{m}, \mathsf{r}, \mathsf{e}_1, \mathsf{e}_2, \boldsymbol{\delta}_u, \boldsymbol{\delta}_v)$ with m binary, $\|(r \mid e_1 \mid e_2)\|_2 \le B$, $\|\delta_u\|_\infty \le 1$, $\|\delta_v\|_\infty \le 52$, and the decompressed ciphertext matches the encryption up to the rounding $\boldsymbol{\delta}_u, \boldsymbol{\delta}_v$: $\mathsf{u} = \mathrm{Decompress}_{11}(\mathsf{c}_1) + \boldsymbol{\delta}_u$, $\mathsf{v} = \mathrm{Decompress}_5(\mathsf{c}_2) + \boldsymbol{\delta}_v$.

Proof of a well-formed ML-KEM-1024 ciphertext

equations

two R_p equations lifted into R_q (carries $\mathsf{v}_u, \mathsf{v}_v$), roundings $\boldsymbol{\delta}_u, \boldsymbol{\delta}_v$ and their bit-decompositions (of $\boldsymbol{\delta}_u$ and the shifted $\boldsymbol{\delta}_v{+}52$) in the witness:

$$\mathsf{A}^\top \mathsf{r} + \mathsf{e}_1 = \mathrm{Decompress}_{11}(\mathsf{c}_1) + \boldsymbol{\delta}_u + p\,\mathsf{v}_u,$$

$$\langle \mathsf{t}, \mathsf{r} \rangle + \mathsf{e}_2 + \lceil p/2 \rfloor \mathsf{m} = \mathrm{Decompress}_5(\mathsf{c}_2) + \boldsymbol{\delta}_v + p\,\mathsf{v}_v.$$

norm bounds

- $\mathsf{r}, \mathsf{e}_1, \mathsf{e}_2$ — **exact** ℓ_2; m and both bit-decomps — **binaryness**.
- gadget reconstruction: $\langle g_u, \mathrm{bitdecomp}(\boldsymbol{\delta}_u) \rangle = \boldsymbol{\delta}_u$, $\langle g_v, \mathrm{bitdecomp}(\boldsymbol{\delta}_v{+}52) \rangle = \boldsymbol{\delta}_v + 52$.
- the lift's carries $\mathsf{v}_u, \mathsf{v}_v$ — **ARP** (no-wrap).

$\mathrm{bitdecomp}$ = decomposition wrt the gadget: $g_u{=}(1, -1)$ (ternary), $g_v{=}(1, 2, 4, 8, 16, 32, 41)$.

Proof of correct TFHE encryption

ZHEnith (ZAMA): proof of *correct TFHE encryption* — from ePrint **2025/699**, §4.11.1.

TFHE public-key encryption

$R_p = \mathbb{Z}_p[X]/(X^d+1)$, $d = 2048$, $p = 2^{64}$, $t = 32$, $\Delta = p/t = 2^{59}$.
given pk $(\mathbf{a}_{\mathrm{pk}}, \mathbf{b}_{\mathrm{pk}}) \in R_p \times R_p$ and a message $m \in \{0, \dots, t-1\}$,
sample $\vec{r} \leftarrow \{0,1\}^d$, $\vec{e}_1 \leftarrow \mathrm{TUniform}(17)^d$, $e_2 \leftarrow \mathrm{TUniform}(17)$,
compute the ciphertext $(\mathbf{a}, b)$:

$$\mathbf{a} = \mathbf{a}_{\mathrm{pk}} \overleftarrow{\mathbf{r}} + \mathbf{e}_1, \qquad b = \langle \vec{b}_{\mathrm{pk}}, \vec{r} \rangle + \Delta\, m + e_2 \pmod p.$$

$\overleftarrow{\mathbf{r}}$ = $\mathbf{r}$ in reversed coefficient order, TUniform(17) = uniform on $[-2^{17}, 2^{17}]$, endpoints at half weight.

statement

given pk and ciphertext, the prover **knows** $(\vec{r}, \vec{e}_1, e_2, m)$ with

$$\vec{r} \in \{0,1\}^d, \qquad \|(e_1|e_2)\|_2 \le B, \qquad m \in \{0, \dots, t-1\},$$

satisfying both encryption equations.

Proof of correct TFHE encryption — route 1

equations

- R_p equation, lifted with carry $\mathbf{v}_1$ — **linear equation over** R_q in a σ_{-1}-image: $\overleftarrow{\mathbf{r}} = X^{d-1}\,\sigma_{-1}(\mathbf{r})$.
- $\mathbb{Z}_p$ equation, lifted with carry v_2 — **one ct-claim**: $\mathrm{ct}\big(\sigma_{-1}(\mathbf{b}_{\mathrm{pk}})\,\mathbf{r} + \Delta\, m + e_2 + p\, v_2 - b\big) = 0$.
- **message range**: commit $\mathrm{bitdecomp}(m)$, add the ct-claim $\mathrm{ct}\big(\sigma_{-1}(\mathrm{pow})\,\mathrm{bitdecomp}(m) - m\big) = 0$, $\mathrm{pow} = (1, 2, \ldots, 2^{\ell-1})$, $\ell = \log_2 t$.

norm bounds

- $\vec{r} \in \{0,1\}^d$ — **binaryness**.
- $\mathrm{bitdecomp}(m)$ — **binaryness**.
- $\|(e_1|e_2)\|_2 \le B$ — **exact** ℓ_2.
- the lift's carries $\mathbf{v}_1, v_2$ — **ARP** (no-wrap).

Proof of correct TFHE encryption — route 2

Move the reversal to b_{pk}

r is *internal randomness*, and reversal is a bijection on $\{0,1\}^d$ — so *sample* the reversed vector directly, $\mathsf{r}' := \overleftarrow{\mathsf{r}} \leftarrow \{0,1\}^d$. The two equations become

$$\mathsf{a} = \mathsf{a}_{\mathrm{pk}}\,\mathsf{r}' + \mathsf{e}_1, \qquad b = \langle \overleftarrow{\vec{b}}_{\mathrm{pk}}, \vec{r}' \rangle + \Delta\, m + e_2 \pmod{p}.$$

- R_p equation — **linear equation over** R_q, the reversal is gone from the witness.
- $\mathbb{Z}_p$ equation — the reversal now sits on the *public* $\overleftarrow{\vec{b}}_{\mathrm{pk}}$, absorbed into the statement: one ct-claim $\mathrm{ct}\big(\sigma_{-1}(\overleftarrow{\mathsf{b}}_{\mathrm{pk}})\,\mathsf{r}' + \Delta\, m + e_2 + p\, v_2 - b\big) = 0$.
- **message range**: ct-claim $\mathrm{ct}\big(\sigma_{-1}(\mathrm{pow})\,\mathrm{bitdecomp}(m) - m\big) = 0$, with $\mathrm{bitdecomp}(m)$ binary.
- remaining range conditions as in route 1.

Potential optimizations (varying confidence in these)

- **Sum-check compression** (mainly for succinct proofs) (RoKoko, ePrint 2026/575)
- **Iterative rejection sampling** (can similar techniques be used e.g. for the masked projections?) — lower $\sigma \Rightarrow$ smaller proofs *and* fewer aborts (Gärtner, CRYPTO 2025)
- **Commitments over a smaller modulus** (see Swoosh slide)
- **General performance improvements** (algorithmic or assembly level optimizations)

Recap

LaZer provides **succinct ZKPoKs** for short secrets in module lattices.

Questions / feedback

Which relations do you need proved?

`lazer-zkp@googlegroups.com` ePrint 2024/1846