HAWK: an update

6th PQC Standardisation Conference, Gaithersburg (MD)

Joppe W. Bos[1], Olivier Bronchain[1], Léo Ducas[2,3], Serge Fehr[2,3], Yu-Hsuan Huang[2], Thomas Pornin[4], Eamonn W. Postlethwaite[5], Thomas Prest[6], **Ludo N. Pulles**[2], Wessel van Woerden[6]

26th September 2025

[1]NXP Semiconductors, [2]Cryptology Group, CWI, [3]Leiden University, [4]NCC Group, [5]King's College London, [6]PQShield

Eurasian sparrowhawk by Bart Hoekstra, CC-BY-NC-SA 4.0

HAWK: an intro

HAWK is the only lattice-based scheme in the 2nd round of NIST's additional standardisation process.

Why use HAWK?

Fast:	KeyGen: 3.5 ms[1]	Sign/Verify: < 0.1 ms
Compact:	$\lvert pk_{\text{pub}} \rvert = 1024$ bytes	$\lvert \text{sig} \rvert = 555$ bytes
Hardware friendly:	$\leq$ 12kiB RAM	no `float`/`double`

[1] on my laptop

HAWK: an intro

HAWK is the only lattice-based scheme in the 2nd round of NIST's additional standardisation process.

Why use HAWK?

Fast:	KeyGen: 3.5 ms[1]	Sign/Verify: < 0.1 ms
Compact:	$\lvert \mathsf{pk}_{\mathsf{pub}} \rvert = 1024$ bytes	$\lvert \mathrm{sig} \rvert = 555$ bytes
Hardware friendly:	$\leq$ 12kiB RAM	no `float`/`double`

[1] on my laptop

HAWK: an intro

The scheme, and its reference implementation are mature:

- isochronous implementation,
- BUFF-secure as is [ADM+24],
- simple samplers: centred binomial and discrete gaussian on two cosets of $\mathbb{Z}$.

Its security is based on relatively new problems...
but they are being actively researched and gaining maturity.

HAWK: an intro

The scheme, and its reference implementation are mature:

- isochronous implementation,
- BUFF-secure as is [ADM+24],
- simple samplers: centred binomial and discrete gaussian on two cosets of $\mathbb{Z}$.

Its security is based on relatively new problems...

but they are being actively researched and gaining maturity.

Update on LIP

Eurasian sparrowhawk by Bart Hoekstra, CC-BY-NC-SA 4.0

LIP

HAWK is *inspired* by the lattice isomorphism problem (LIP).

[DvW22] *On LIP, quadratic forms, remarkable lattices, and cryptography*
💡 Build cryptography from **non random lattices** with good geometry.

[BGPS23] *Just how hard are rotations of $\mathbb{Z}^n$?*
💡 LIP seems hard for $\mathbb{Z}^n$.

[DPPvW22] *HAWK: Module LIP makes Lattice Signatures Fast, Compact and Simple*
💡 Use **module structure** over cyclotomic rings and **simplify** KeyGen and Sign.

Does module structure or the simplification weaken security?

LIP

HAWK is *inspired* by the lattice isomorphism problem (LIP).

[DvW22] *On LIP, quadratic forms, remarkable lattices, and cryptography*
💡 Build cryptography from **non random lattices** with good geometry.

[BGPS23] *Just how hard are rotations of $\mathbb{Z}^n$?*
💡 LIP seems hard for $\mathbb{Z}^n$.

[DPPvW22] *HAWK: Module LIP makes Lattice Signatures Fast, Compact and Simple*
💡 Use **module structure** over cyclotomic rings and **simplify** KeyGen and Sign.

Does module structure or the simplification weaken security?

HAWK: private key recovery

Recover HAWK $\mathsf{sk}_{\mathsf{priv}}$ $\iff$ solve 'search module LIP' (smLIP) in rank 2

In particular,

setup: use the ring $R = \mathbb{Z}[\zeta]$, with $\zeta^n = -1 \qquad (n \in \{2^9, 2^{10}\})$,
sample $\mathbf{U} \in \mathrm{GL}_2(R)$, and compute $\mathbf{Q} = \mathbf{U}^*\mathbf{U}$,

problem: upon input $\mathbf{Q}$, output any $\mathbf{V} \in \mathrm{GL}_2(R)$ with $\mathbf{Q} = \mathbf{V}^*\mathbf{V}$.

HAWK was designed with [GS02, LS19] in mind: rank 1 'ideal' LIP is polynomial time.

HAWK: private key recovery

Recover HAWK $\mathsf{sk}_{\mathsf{priv}}$ $\iff$ solve 'search module LIP' (smLIP) in rank 2

In particular,

setup: use the ring $R = \mathbb{Z}[\zeta]$, with $\zeta^n = -1 \qquad (n \in \{2^9, 2^{10}\})$,
sample $\mathbf{U} \in \mathrm{GL}_2(R)$, and compute $\mathbf{Q} = \mathbf{U}^*\mathbf{U}$,

problem: upon input $\mathbf{Q}$, output any $\mathbf{V} \in \mathrm{GL}_2(R)$ with $\mathbf{Q} = \mathbf{V}^*\mathbf{V}$.

HAWK was designed with [GS02, LS19] in mind: rank 1 'ideal' LIP is polynomial time.

smLIP using any number field

More general: $R = \mathcal{O}_K$ where K is a number field.

Literature on smLIP$_K$:

[MPPW24] (heuristic) smLIP is polynomial time if K is totally real,

[APvW25] (heuristic*) smLIP is polynomial time if K has at least one real embedding,

[CME+25] smLIP reduces to solving a 'norm reduced principal ideal problem' (nrdPIP) in (for HAWK) a noncommutative quaternion algebra.

Cyclotomics are totally imaginary, hence the first two attacks do not apply.

The current understanding is: nrdPIP is fundamentally different to PIP, and there seems no easy way to generalise Gentry–Szydlo.

smLIP using any number field

More general: $R = \mathcal{O}_K$ where K is a number field.

Literature on smLIP$_K$:

[MPPW24] ~~(heuristic) smLIP is polynomial time if K is totally real,~~

[APvW25] ~~(heuristic*) smLIP is polynomial time if K has at least one real embedding,~~

[CME+25] smLIP reduces to solving a 'norm reduced principal ideal problem' (nrdPIP) in (for HAWK) a noncommutative quaternion algebra.

Cyclotomics are totally imaginary, hence the first two attacks do not apply.

The current understanding is: nrdPIP is fundamentally different to PIP, and there seems no easy way to generalise Gentry–Szydlo.

smLIP using any number field

More general: $R = \mathcal{O}_K$ where K is a number field.

Literature on smLIP$_K$:

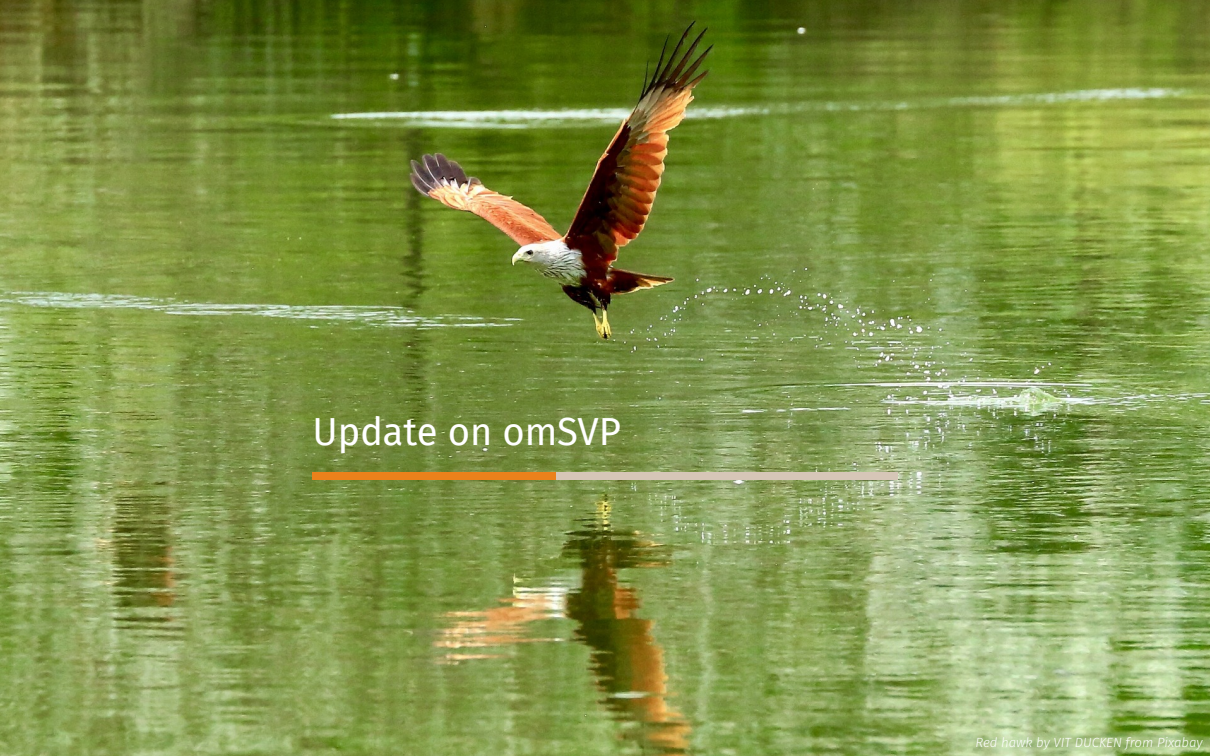
[MPPW24] ~~(heuristic) smLIP is polynomial time if K is totally real,~~

[APvW25] ~~(heuristic*) smLIP is polynomial time if K has at least one real embedding,~~

[CME+25] smLIP reduces to solving a 'norm reduced principal ideal problem' (nrdPIP) in (for HAWK) a noncommutative quaternion algebra.

Cyclotomics are totally imaginary, hence the first two attacks do not apply.

The current understanding is: nrdPIP is fundamentally different to PIP, and there seems no easy way to generalise Gentry–Szydlo.



Update on omSVP

Red hawk by VIT DUCKEN from Pixabay

SUF-CMA security of HAWK

One more short vector problem (omSVP)
reduces to
SUF-CMA security of HAWK in qROM.

(if you forge a HAWK signature, you solve omSVP)

Solving omSVP leaves the *design* of HAWK unsupported

(if you find another short vector, you need to find a hash preimage)

SUF-CMA security of HAWK

One more short vector problem (omSVP)
reduces to
SUF-CMA security of HAWK in qROM.

(if you forge a HAWK signature, you solve omSVP)

Solving omSVP leaves the *design* of HAWK unsupported

(if you find another short vector, you need to find a hash preimage)

omSVP in detail

omSVP captures the idea that,

> given an oracle providing short lattice vectors $\mathbf{v}_i$,
> returning a **'fresh'** and similarly short lattice vector is hard.

Here, 'fresh' accounts for structure.

HAWK v1.0: not 'fresh' is multiplying $\mathbf{v}_i$ with a root of unity, ζ^j.

[LJPW24] There is an efficiently computable *'symplectic automorphism'* ω.
💣 This solves omSVP: take $\mathbf{v}_1$ from the oracle, and output $\omega(\mathbf{v}_1)$.

HAWK v1.1: $\zeta^j \mathbf{v}_i$ and $\zeta^j \omega(\mathbf{v}_i)$ are not 'fresh'.

[vGP25] Any other automorphism makes smLIP easy to solve *(later today)*
☑ omSVP v1.1 cannot be trivially won.

omSVP in detail

omSVP captures the idea that,

> given an oracle providing short lattice vectors $\mathbf{v}_i$,
> returning a **'fresh'** and similarly short lattice vector is hard.

Here, 'fresh' accounts for structure.

HAWK v1.0: not 'fresh' is multiplying $\mathbf{v}_i$ with a root of unity, ζ^j.

[LJPW24] There is an efficiently computable *'symplectic automorphism'* ω.
💣 This solves omSVP: take $\mathbf{v}_1$ from the oracle, and output $\omega(\mathbf{v}_1)$.

HAWK v1.1: $\zeta^j\mathbf{v}_i$ and $\zeta^j\omega(\mathbf{v}_i)$ are not 'fresh'.

[vGP25] Any other automorphism makes smLIP easy to solve *(later today)*
☑ omSVP v1.1 cannot be trivially won.

Update on Key Generation

Key Generation

- Given $f, g \in R$, HAWK key generation requires one to find $F, G \in R$ with

$$fG - gF = 1.$$

  (Similar to FALCON except we use 1 instead of prime q.)

- Solve recursively by passing to fields of half the degree until reaching $\mathbb{Z}$ then lift solutions [PP19]. Use size reduction to find a *small* lift F, G.

Improvements

By tuning this intermediate size reduction, the works [Por23] and [Por25] were able to reduce the cycle count and RAM usage.

Table 1: Performance of HAWK-512 key generation

	Prior	[Pornin25]	
x86 w/ AVX2	7.68 Mcyc	6.11 Mcyc	↓ 20%
ARM Cortex M4	47.92 Mcyc	38.56 Mcyc	↓ 20%
RAM usage	14 kiB	12 kiB	↓ 14%

Wrapping up

Northern harrier by Iulian Ursache from Pixabay

Cryptanalysis conclusion

- At present, key recovery requires either a single real embedding or solving nrdPIP in a noncommutative quaternion algebra. It is not known how to efficiently perform the latter.
- Finding any further automorphisms would practically recover the secret key, but this seems as hard as finding a short vector ([vGP25] *later today*).
- Solving omSVP does not imply signature forgeries against HAWK, only that its SUF-CMA security is unsupported by a reduction.

Open research directions

1. How hard is nrdPIP in the HAWK case?
2. How hard is omSVP concretely, assuming no further automorphisms?
3. Can the (relative) simplicity of discrete Gaussians over $\mathbb{Z}$ and $\mathbb{Z} + 1/2$ lead to plausibly efficient masking?
   ([GR24] provides an initial side-channel study with practical countermeasures.)

Thank you!

Questions?

https://hawk-sign.info

team@hawk-sign.info

Bibliography – i

[ADM+24] Thomas Aulbach, Samed Düzlü, Michael Meyer, Patrick Struck, and Maximiliane Weishäupl, *Hash your keys before signing - BUFF security of the additional NIST PQC signatures*, Post-Quantum Cryptography - 15th International Workshop, PQCrypto 2024, Part II (Markku-Juhani Saarinen and Daniel Smith-Tone, eds.), Springer, Cham, June 2024, pp. 301–335.

[APvW25] Bill Allombert, Alice Pellet-Mary, and Wessel van Woerden, *Cryptanalysis of rank-2 module-LIP: A single real embedding is all it takes*, EUROCRYPT 2025, Part II (Serge Fehr and Pierre-Alain Fouque, eds.), LNCS, vol. 15602, Springer, Cham, May 2025, pp. 184–212.

[BGPS23] Huck Bennett, Atul Ganju, Pura Peetathawatchai, and Noah Stephens-Davidowitz, *Just how hard are rotations of $\mathbb{Z}^n$? algorithms and cryptography with the simplest lattice*, EUROCRYPT 2023, Part V (Carmit Hazay and Martijn Stam, eds.), LNCS, vol. 14008, Springer, Cham, April 2023, pp. 252–281.

[CME+25] Clémence Chevignard, Guilhem Mureau, Thomas Espitau, Alice Pellet-Mary, Heorhii Pliatsok, and Alexandre Wallet, *A reduction from hawk to the principal ideal problem in a quaternion algebra*, EUROCRYPT 2025, Part II (Serge Fehr and Pierre-Alain Fouque, eds.), LNCS, vol. 15602, Springer, Cham, May 2025, pp. 154–183.

[DPPvW22] Léo Ducas, Eamonn W. Postlethwaite, Ludo N. Pulles, and Wessel van Woerden, *Hawk: Module LIP makes lattice signatures fast, compact and simple*, ASIACRYPT 2022, Part IV (Shweta Agrawal and Dongdai Lin, eds.), LNCS, vol. 13794, Springer, Cham, December 2022, pp. 65–94.

Bibliography – ii

[DvW22] Léo Ducas and Wessel van Woerden, *On the lattice isomorphism problem, quadratic forms, remarkable lattices, and cryptography*, EUROCRYPT 2022, Part III (Orr Dunkelman and Stefan Dziembowski, eds.), LNCS, vol. 13277, Springer, Cham, May / June 2022, pp. 643–673.

[GR24] Morgane Guerreau and Mélissa Rossi, *A not so discrete sampler: Power analysis attacks on hawk signature scheme*, IACR Transactions on Cryptographic Hardware and Embedded Systems **2024** (2024), no. 4, 156–178.

[GS02] Craig Gentry and Michael Szydlo, *Cryptanalysis of the revised NTRU signature scheme*, EUROCRYPT 2002 (Lars R. Knudsen, ed.), LNCS, vol. 2332, Springer, Berlin, Heidelberg, April / May 2002, pp. 299–320.

[LJPW24] Hengyi Luo, Kaijie Jiang, Yanbin Pan, and Anyu Wang, *Cryptanalysis of rank-2 module-LIP with symplectic automorphisms*, ASIACRYPT 2024, Part IV (Kai-Min Chung and Yu Sasaki, eds.), LNCS, vol. 15487, Springer, Singapore, December 2024, pp. 359–385.

[LS19] Hendrik W. Lenstra Jr. and Alice Silverberg, *Testing isomorphism of lattices over CM-Orders*, SIAM Journal on Computing **48** (2019), no. 4, 1300–1334.

[MPPW24] Guilhem Mureau, Alice Pellet-Mary, Georgii Pliatsok, and Alexandre Wallet, *Cryptanalysis of rank-2 module-LIP in totally real number fields*, EUROCRYPT 2024, Part VII (Marc Joye and Gregor Leander, eds.), LNCS, vol. 14657, Springer, Cham, May 2024, pp. 226–255.

Bibliography – iii

[Por23] Thomas Pornin, *Improved key pair generation for falcon, BAT and hawk*, Cryptology ePrint Archive, Report 2023/290, 2023.

[Por25] _____, *Improved (again) key pair generation for falcon, BAT and hawk*, Cryptology ePrint Archive, Report 2025/1239, 2025.

[PP19] Thomas Pornin and Thomas Prest, *More efficient algorithms for the NTRU key generation using the field norm*, PKC 2019, Part II (Dongdai Lin and Kazue Sako, eds.), LNCS, vol. 11443, Springer, Cham, April 2019, pp. 504–533.

[vGP25] Daniël M. H. van Gent and Ludo N. Pulles, *HAWK: Having automorphisms weakens key*, IACR Communications in Cryptology **2** (2025), no. 2.