Weak Keys Break the BUFF Security of HAWK

Quang Dao[1]

Carnegie Mellon University & LayerZero Labs
qvd@andrew.cmu.edu

Abstract. HAWK is a signature scheme based on the module lattice isomorphism problem, and the only lattice-based candidate in the third round of NIST's call for additional post-quantum signatures. Its specification claims that HAWK achieves the BUFF (Beyond UnForgeability Features) security properties "as is", without applying the generic BUFF transform, citing the analysis of Aulbach, Düzlü, Meyer, Struck, and Weishäupl (PQCrypto '24).

We refute this claim for HAWK exactly as specified. Several of the BUFF games let the adversary register a public key of its own choosing, yet the HAWK reference verifier performs almost no validity check on a key beyond decoding it. We exhibit degenerate "weak" public keys under which the all-zero signature verifies for all random-oracle challenges except the negligible symmetry-breaking corner case, and use them to break the three BUFF properties whose games let the adversary supply both the verification key and the signature: message-bound signatures, malicious strong universal exclusive ownership, and weak non-resignability.

We trace these breaks to gaps in the BUFF proofs for HAWK of Aulbach et al.: one missing case analysis, and three steps that silently assume properties of the adversarially chosen keys as if they come from honest key generation. Honest key generation already enforces bounds that would reject our weak-key family on the attacks we exhibit. We prove that enforcing this norm floor at verification yields message-bound security for constant keys, but we make no claim that it fully restores BUFF security.

Keywords: HAWK · BUFF security · message-bound signatures · exclusive ownership · weak keys · post-quantum signatures.

1 Introduction

Standard unforgeability notions for signature schemes, such as EUF-CMA and SUF-CMA, give the adversary an honestly generated public key and ask it to forge under that key. A long line of work has shown that real protocols need more. Pornin and Stern observed that a signature valid under one public key may be claimed under a different public key, breaking the reverse direction of trust that protocols often assume [10]. Ristenpart and Yilek connected such exclusive-ownership failures to rogue-key attacks and proofs of possession [11]. Jackson, Cremers, Cohn-Gordon, and Sasse catalogued the protocol attacks that remain possible under EUF-CMA, including producing signatures under adversarial keys

without knowing the message, and computing weak key pairs for which a single signature verifies many messages [8]. These behaviors were formalized as the BUFF properties (Beyond UnForgeability Features) by Cremers, Düzlü, Fiedler, Fischlin, and Janson, who in the same paper gave a lightweight transform (the BUFF transform) that generically provides them for any unforgeable signature scheme [2].

HAWK is a lattice signature scheme based on the module lattice isomorphism problem [5], and the only lattice-based candidate in the third round of NIST's call for additional post-quantum signatures [9]. Its signatures are more compact than FALCON's at matching security, and its signing and verification use only fixed-point integer arithmetic, with no floating-point operations. The HAWK v1.1 specification states, in its advantages section, that "it has been shown that HAWK achieves the BUFF security properties as is" [12], citing the analysis of Aulbach, Düzlü, Meyer, Struck, and Weishäupl (henceforth ADMSW24) [1]. ADMSW24 marks HAWK as satisfying S-CEO, S-DEO, MBS, and wNR (defined in Section 2.1), and its Remark 7 asserts that the lighter PS-3 transform of Pornin and Stern [10] suffices for HAWK, so the full BUFF transform is unnecessary. PS-3 is the variant that hashes the message together with the public key into the challenge, which HAWK already does via $M = \mathsf{H}(m \,\|\, \mathsf{H}(pk))$. A subsequent paper of Düzlü and Struck builds on this, taking message-bound signatures as the right formalization of "the scheme has no effective weak keys" and using HAWK as a canonical lattice example where this holds [7].

Several of the BUFF games depart from EUF-CMA by letting the adversary output its own public key, not only an honest key-generation output. These games are what this paper exploits: the HAWK verifier accepts adversarial public keys that decode cleanly and pass the final quadratic-norm bound, yet face no public-key validity check in the specification.

1.1 Our results

We turn this gap into concrete breaks of three BUFF properties for HAWK as specified, and trace the breaks to the proofs that claimed otherwise.

(1) *Weak-key counterexamples* (Section 3). We construct public keys with constant $q_{00} = 1$ and $q_{01} = a$ that the HAWK codec decodes. With the zero signature $s_1 = 0$, the unmodified HAWK verifier accepts them on all random-oracle challenges except the negligible case $h_1 = 0$. From this family we give explicit counterexamples to MBS, M-S-UEO, and wNR on all three HAWK parameter sets. Each is accepted by the official C and Python reference verifiers and reproducible from our companion artifact [3].

(2) *The errors in the proofs* (Section 4). We trace the four HAWK proofs of ADMSW24, Section 5.1, to two kinds of error. One is a missing case analysis: a coordinate in the proof's NTRU preimage decomposition can be exactly zero, which the proofs never consider. The other three assume the honest key-generation magnitude and distribution of the key, neither of which the verifier enforces.

Key-generation properties versus verifier acceptance. HAWK key generation rejection-samples until $\|f\|^2 + \|g\|^2 \geq \ell_{\text{low}}$ (Table 1), yet the reference verifier performs almost no check that a decoded public key came from that process. Our weak keys violate this requirement while still decoding and passing verification. Section 5 records two predicates that capture these key-generation properties. The first is a norm floor $\mathsf{KeyNormCheck}$ matching key generation's minimum. The second is the finite algebraic identity $\mathsf{KeyIdentityCheck}$ needed to equate the verifier's modular norm with the intended quadratic form (Section 6). Every honest key passes the norm floor exactly, and the floor rejects all of our counterexamples. We prove that enforcing this bound would defeat the entire constant-q_{00} attack class on message-bound signatures when the norm is evaluated as the intended quadratic form. Section 6 states what is proved when these predicates are enforced and what remains open, including keys whose Fourier coordinates are uneven. The full BUFF transform of [2] provably yields the BUFF properties, but enlarges the signature and still leaves the verifier accepting arbitrary weak keys (Section 7).

Scope. We do not claim any weakness in HAWK's EUF-CMA or SUF-CMA security on honestly generated keys, for which we find no flaw in the security proof of the HAWK spec, nor improved attacks against the published parameter sets. The weak keys are adversarially chosen and are rejected by HAWK key generation; only the verifier accepts them. We also do not exhibit S-CEO or S-DEO counterexamples for this family; extended search finds none, and Section 6 explains why the games appear to survive even though the ADMSW24 proofs of them are unsound on the verifier-accepted domain.

Related work. The BUFF properties were formalized, together with a generic transform that provides them, by Cremers, Düzlü, Fiedler, Fischlin, and Janson [2], building on the exclusive-ownership line of Pornin and Stern [10], Ristenpart and Yilek [11], and Jackson et al. [8]. ADMSW24 [1] is the BUFF survey of the NIST additional-signature candidates whose HAWK analysis we examine here. For the structured-lattice schemes, Düzlü, Fiedler, and Fischlin add BUFF security to FALCON without increasing the signature size [6]; their hash-into-signature route is the FALCON analogue of the full-transform approach we discuss for HAWK in Section 7. Weak non-resignability carries its own subtleties: Don, Fehr, Huang, Liao, and Struck show that the original non-resignability notion is unachievable and repair it for the generic BUFF transform [4]. Our wNR counterexample is against the verifier-accepted key domain of an unmodified scheme rather than against a transformed scheme, so it is orthogonal to that repair.

2 Preliminaries

A signature scheme $\Sigma = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Verify})$ has key generation $(sk, pk) \leftarrow \mathsf{KGen}(1^\lambda)$ and signing $\sigma \leftarrow \mathsf{Sign}(sk, m)$. Verification $\mathsf{Verify}(pk, m, \sigma) \in \{0, 1\}$ is deterministic. We write λ for the security parameter and call a function negligible, $\mathsf{negl}(\lambda)$, if it vanishes faster than any inverse polynomial.

2.1 BUFF properties

The BUFF properties of Cremers, Düzlü, Fiedler, Fischlin, and Janson [2] formalize security against *maliciously generated* public keys, a regime that EUF-CMA and SUF-CMA do not address. We recall the five notions in the strong forms that ADMSW24 [1] uses for HAWK, each through a security experiment (Figure 1). All games are in the random-oracle model for Σ's hash functions.

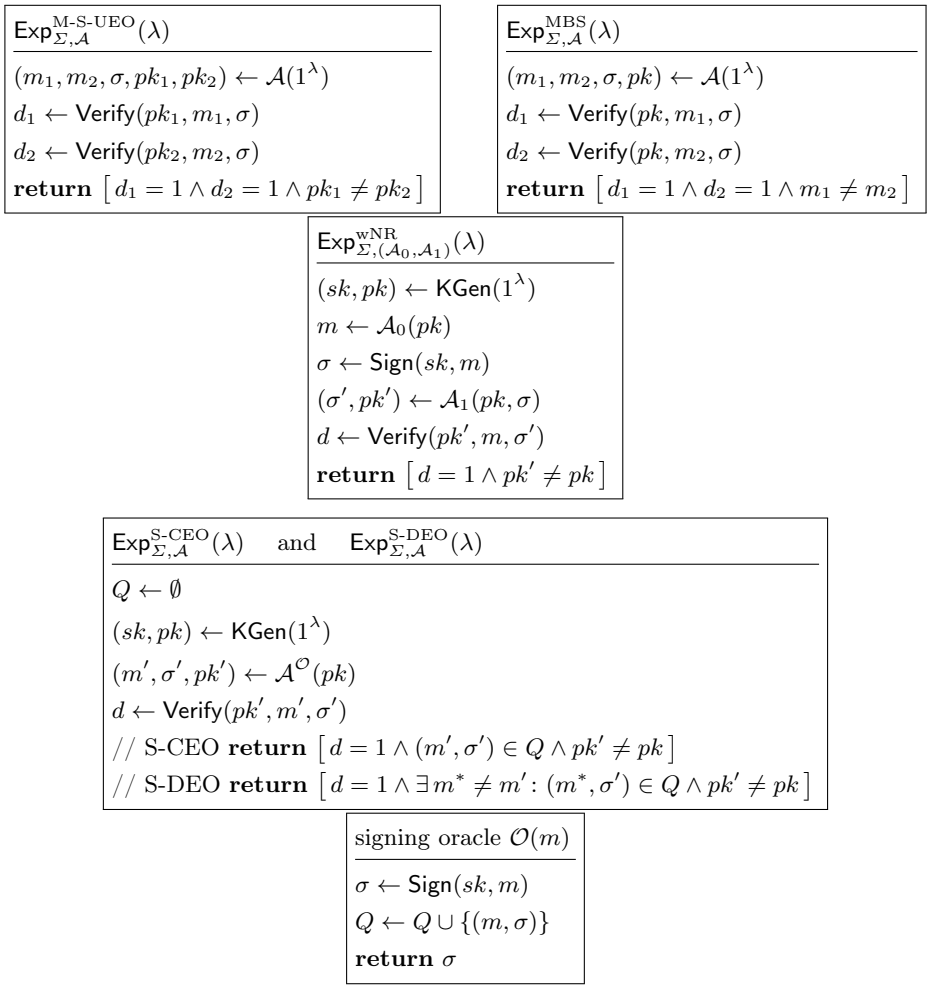

Fig. 1. The BUFF experiments, following [2,1]. S-CEO and S-DEO share the signing oracle $\mathcal{O}$ and differ only in the winning predicate.

For $X \in \{\text{M-S-UEO}, \text{MBS}, \text{S-CEO}, \text{S-DEO}\}$ and a PPT adversary $\mathcal{A}$, define $\mathsf{Adv}^X_{\Sigma,\mathcal{A}}(\lambda) = \Pr[\mathsf{Exp}^X_{\Sigma,\mathcal{A}}(\lambda) = 1]$. For wNR and PPT $(\mathcal{A}_0, \mathcal{A}_1)$, define

$\mathsf{Adv}^{\mathrm{wNR}}_{\Sigma,(\mathcal{A}_0,\mathcal{A}_1)}(\lambda) = \Pr[\mathsf{Exp}^{\mathrm{wNR}}_{\Sigma,(\mathcal{A}_0,\mathcal{A}_1)}(\lambda) = 1]$. The scheme *provides* the property if the corresponding advantage is $\mathsf{negl}(\lambda)$ for every PPT adversary. Informally:

M-S-UEO *Malicious strong universal exclusive ownership.* One signature σ verifies under two distinct public keys, both chosen by the adversary with no secret key; the two messages may be equal.

MBS *Message-bound signatures.* One key-signature pair (pk, σ) verifies two distinct messages.

S-CEO / S-DEO *Strong conservative / destructive exclusive ownership.* Given an honest pk and a signing oracle, the adversary outputs a fresh key $pk' \neq pk$ under which a previously obtained signature still verifies. In S-CEO the message is the same one it queried; in S-DEO it is a different one.

wNR *Weak non-resignability.* Given only an honest signature on a high-entropy message it never learns, the adversary re-signs that message under a fresh key $pk' \neq pk$.

Three of the five games, M-S-UEO, MBS, and wNR, let the adversary supply the public key(s) under which verification must succeed, with no honestly generated secret behind them; S-CEO and S-DEO start from an honest key but the winning key pk' is again adversarial. This adversarial-key freedom is what our counterexamples exploit. We use the strong notions throughout, following ADMSW24; recall that S-CEO and S-DEO together are equivalent to S-UEO, which M-S-UEO implies [2]. The wNR game is the achievable weakening of the original non-resignability notion of [2]. Don, Fehr, Huang, Liao, and Struck showed that the original notion, in which $\mathcal{A}_1$ also receives auxiliary information about m, is unachievable for high-entropy messages, and wNR removes that auxiliary input [4]. For wNR, this entropy and no-state condition is part of the adversary class: $\mathcal{A}_0$'s message distribution, conditioned on pk, must carry super-logarithmic min-entropy, and no state is passed from $\mathcal{A}_0$ to $\mathcal{A}_1$. Without that restriction, $\mathcal{A}_1$ could simply learn or guess the message.

2.2 HAWK verification

HAWK works over the cyclotomic ring $\mathcal{R} = \mathbb{Z}[X]/(X^n + 1)$ with $n = 2^k$, with Hermitian adjoint a^* given by complex conjugation in the power-of-two cyclotomic field. For a polynomial $a = \sum_j a[j]X^j$, $\|a\|$ denotes its coefficient Euclidean norm. For a self-adjoint polynomial r, we write r_u for the eigenvalues of multiplication by r under the unitary negacyclic Fourier transform, normalized so that Parseval gives $\sum_u |a_u|^2 = \sum_j |a[j]|^2$ and the average eigenvalue is the constant coefficient, $(1/n)\sum_u r_u = r[0]$. The secret key is a basis $B = \left(\begin{smallmatrix} f & F \\ g & G \end{smallmatrix}\right)$ with $fG - gF = 1$. The public key is the Gram matrix

$$Q = B^*B = \begin{pmatrix} q_{00} & q_{01} \\ q_{01}^* & q_{11} \end{pmatrix}, \qquad q_{00}q_{11} - q_{01}q_{01}^* = 1.$$

Only q_{00} (self-adjoint) and q_{01} are encoded; the verifier never materializes q_{11}, which it recovers from $\det Q = 1$. Key generation resamples (f, g) until $\|f\|^2 + \|g\|^2 \geq \ell_{\text{low}}$, and by construction $q_{00}[0] = \|f\|^2 + \|g\|^2$. We write $Q_{\min} = \ell_{\text{low}}$ for the threshold used in KeyNormCheck below.

A signature is $(salt, s_1)$, with s_0 omitted from the wire. The squared norm $\|(w_0, w_1)\|_Q^2$ is the quadratic form induced by the public Gram matrix Q. Verification (Figure 2) recomputes the challenge, rebuilds s_0, and tests whether this norm is below the bound.

$\mathsf{Verify}(pk, m, (salt, s_1))$

decode $pk \to (q_{00}, q_{01})$
$M \leftarrow \mathsf{H}(m \,\|\, \mathsf{H}(pk))$
$(h_0, h_1) \leftarrow \mathsf{H}(M \,\|\, salt) \in \{0,1\}^n \times \{0,1\}^n$
$w_1 \leftarrow h_1 - 2s_1$
if the parity tie-break on w_1 fails (spec Algorithm 18), **return** 0
$s_0 \leftarrow \mathsf{RebuildS0}(q_{00}, q_{01}, h_0, h_1, s_1)$
$w_0 \leftarrow h_0 - 2s_0$
return $\left[\, \|(w_0, w_1)\|_Q^2 \leq 8n\sigma_{\text{verify}}^2 \,\right]$

Fig. 2. HAWK verification at the level we analyze. The HAWK spec implements RebuildS0 and the norm test in fixed-point and two-modular-prime integer arithmetic; here we write the exact real-number form of the same test. Our counterexamples verify under both.

The Cholesky identity. Using $q_{11} = (1 + q_{01}q_{01}^*)/q_{00}$, the public quadratic form admits a coordinate-wise decomposition that all of the HAWK BUFF proofs rest on. Write $q_{00,u}$ for these eigenvalues of multiplication by q_{00}; we call them the *Fourier coordinates* of q_{00}. With $y = w_0 + (q_{01}/q_{00})\, w_1$ for the rebuild residual that Algorithm 18 targets, the first sum below measures error from reconstructing s_0 and the second measures the remaining w_1 component:

$$\|(w_0, w_1)\|_Q^2 \;=\; \underbrace{\sum_u q_{00,u}\, |y_u|^2}_{\text{rebuild term}} \;+\; \underbrace{\sum_u \frac{1}{q_{00,u}}\, |w_{1,u}|^2}_{\text{noise term}}. \tag{1}$$

RebuildS0 chooses s_0 so that the residual coefficients $y[j]$ lie in the rounding cell specified by Algorithm 18 on accepted signatures. The reference verifier computes the integer norm $N = n\,\|(w_0, w_1)\|_Q^2$ via two NTT primes p_1, p_2 (PolyQnorm, HAWK Algorithm 19), and accepts when the two residues agree, N is divisible by n, and $N/n \leq \lfloor 8n\sigma_{\text{verify}}^2 \rfloor$.

In Table 1, we list the parameters used by HAWK for its toy setting, along with NIST Security Level I and V.

Table 1. HAWK parameter sets used in this paper. σ_{verify} is the verification standard deviation; the integer norm bound is $\lfloor 8n\sigma_{\text{verify}}^2 \rfloor$; ℓ_{low} is the minimum value of $\|f\|^2 + \|g\|^2 = q_{00}[0]$ that key generation accepts.

set	n	σ_{verify}	$\lfloor 8n\sigma_{\text{verify}}^2 \rfloor$	ℓ_{low}
HAWK-256	256	1.042	2223	556
HAWK-512	512	1.425	8317	2080
HAWK-1024	1024	1.571	20218	7981

3 Weak Keys and Concrete Counterexamples

3.1 The weak-key family

Consider the byte-decodable public key with constant polynomials

$$q_{00} = 1, \qquad q_{01} = a,$$

where $a \in \mathbb{Z}$ lies in the public-key codec range for the parameter set, together with the signature $(salt, s_1)$ with $s_1 = 0$. This key has the algebraic NTRU preimage

$$f = 1, \quad g = 0, \quad F = a, \quad G = 1, \qquad fG - gF = 1,$$

with $q_{00} = ff^* + gg^* = 1$ and $q_{01} = Ff^* + Gg^* = a$. This is a valid secret-key matrix in every algebraic sense. It is not an honest key-generation output: key generation rejects it because $\|f\|^2 + \|g\|^2 = 1$ is far below the minimum ℓ_{low} of Table 1. The verifier imposes no such floor.

3.2 Verifier accepts weak keys

With $s_1 = 0$ we have $w_1 = h_1$, which is binary. For $q_{00} = 1$ the noise term of (1) has every weight $1/q_{00,u} = 1$, and RebuildS0 reconstructs w_0 so that $y = w_0 + a\, w_1$ is coefficientwise tiny. Specializing the Cholesky identity to $q_{00} = 1$, $q_{01} = a$,

$$\|w\|_Q^2 = \|w_0 + a\, w_1\|^2 + \|w_1\|^2 ,$$

where the residual $w_0 + a\, w_1$ is bounded by Algorithm 18's ordinary rounding cell. Since $w_1 = h_1$ is binary, $\|w_1\|^2 \le n$, and the rounding cell gives $\|w_0 + a\, w_1\|^2 \le n$. Thus $\|w\|_Q^2 \le 2n$, which is below $\lfloor 8n\sigma_{\text{verify}}^2 \rfloor$ for all three parameter sets. The rounding step absorbs the cross term $a\, w_1$, so the residual remains at the ordinary rounding scale rather than at the scale of $a\, w_1$. The only remaining verifier condition is the symmetry-breaking check on $w_1 = h_1$. With HAWK's sign convention, a binary nonzero h_1 passes this check because its first nonzero coefficient is positive; the excluded event $h_1 = 0$ has probability 2^{-n} in the random-oracle model.

3.3 The counterexamples

We instantiate the canonical witness with $q_{00} = 1$, $q_{01} = 1$, $salt = 0$, $s_1 = 0$, encoded through the official HAWK codec. Every record below is accepted by both the unmodified C reference verifier (`hawk_verify_finish`) and the unmodified Python reference verifier (`hawkverify`), on all three parameter sets.[1] The witnesses, their digests, and a dual-verifier reproduction harness are available in the companion artifact [3].

MBS. The single encoded pair (pk, sig) above verifies for two distinct messages $m =$ "`HAWK malformed-key MBS counterexample message 0`" and $m' =$ "`... message 1`". For HAWK-512 the per-message values of $\|w\|_Q^2$ are 499 and 513, both below the bound 8317. This directly falsifies MBS over the verifier-accepted public-key domain.

M-S-UEO. For a fixed message and the fixed zero signature, the verifier accepts the four pairwise-distinct public keys $q_{00} = 1$, $q_{01} \in \{0, 1, 2, 16\}$. A single (m, sig) thus verifies under four distinct keys, falsifying M-S-UEO over malicious keys.

wNR. Let $\mathcal{A}_0$ output a high-entropy message sampled independently of the honest key. Let $\mathcal{A}_1$ ignore the honest pk and the honest signature it receives, and instead return the fixed malformed pair (pk', sig') from the MBS witness. By the argument of Section 3.2, verification accepts for every random-oracle challenge with $h_1 \neq 0$, so $\mathcal{A}_1$ wins with probability at least $1 - 2^{-n}$ without learning the hidden message. The artifact also samples this hidden-message experiment with SHAKE-derived messages; every sampled record accepts, 100 out of 100 on each parameter set.

4 Unsoundness of the ADMSW24 Section 5.1 Proofs

In this section, we reconcile our concrete counterexamples with the security proofs of ADMSW24 by pinpointing the exact gaps that permit our attacks. Crucially, our counterexamples do not rely on keys outside the domain of the proofs. The preimage $(f, g, F, G) = (1, 0, 1, 1)$ is a valid HAWK basis satisfying $B^*B = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 2 \end{smallmatrix}\right)$ and $fG - gF = 1$. Instead, our witnesses reveal that specific steps in the proofs simply fail for this valid basis. These failures stem from four distinct gaps that affect all four proofs in Section 5.1 of ADMSW24 (S-CEO, S-DEO, MBS, and wNR). The gaps fall into two categories. The first gap is a missing case analysis. The remaining three gaps silently assume that the key has the magnitude, distribution, or preimage of an honestly generated key, none of which the verifier actually enforces.

[1] The companion artifact's default reproduction runs the C verifier on all records and the Python verifier on all MBS and M-S-UEO records, and samples the wNR records; a full Python cross-check of all wNR records runs under an opt-in flag.

Setup, restated. For two messages that both accept under the same (pk, sig), ADMSW24 bounds the distance $\|w - w'\|_Q = \|B(w - w')\|$ by expanding it as

$$\left\| \begin{pmatrix} (h_1 - h_1')(F - fq_{01}/q_{00}) - 2f(\varepsilon - \varepsilon'), \\ (h_1 - h_1')(G - gq_{01}/q_{00}) - 2g(\varepsilon - \varepsilon') \end{pmatrix} \right\|, \tag{2}$$

where $\varepsilon, \varepsilon' \in [-\frac{1}{2}, \frac{1}{2})$ denote the rounding errors in $s_0 = \lceil h_0/2 + (q_{01}/q_{00})(h_1/2 - s_1) \rfloor$. The proof argues that this distance falls below 2θ (twice the verification radius) only with negligible probability, because the factors $F - fq_{01}/q_{00}$ and $G - gq_{01}/q_{00}$ are fixed and amplify the random differences $h_1 - h_1'$ and $\varepsilon - \varepsilon'$.

Flaw 1: a case the proof never handles. For the preimage $(1, 0, 1, 1)$, in every Fourier coordinate,

$$F - fq_{01}/q_{00} = 1 - 1 \cdot 1 = 0, \tag{3}$$
$$G - gq_{01}/q_{00} = 1 - 0 \cdot 1 = 1, \tag{4}$$
$$f = 1, \quad g = 0. \tag{5}$$

The first coordinate collapses to $-2(\varepsilon - \varepsilon')$, whose magnitude is strictly bounded by 2 regardless of $h_1 - h_1'$. The second coordinate collapses to $h_1 - h_1'$, leaving no amplification factor in front of it. The proof needs the $h_1 - h_1'$ contribution to dominate, but here the relevant coefficient is exactly zero. There is no case analysis in ADMSW24 for $F - fq_{01}/q_{00} = 0$, even though this holds for any preimage satisfying $fq_{01} = Fq_{00}$, a subspace and not a measure-zero point.

Flaw 2: an assumed honest key norm. Where the factors are nonzero, the negligible-probability conclusion silently needs $F - fq_{01}/q_{00}$, $G - gq_{01}/q_{00}$, f, and g to have magnitudes of order $\sigma_{\text{kg}}\sqrt{n}$, where σ_{kg} is the key-generation standard deviation. Honest key generation enforces this through its minimum-size requirement $\|f\|^2 + \|g\|^2 \geq \ell_{\text{low}}$, but the verifier accepts the key $(1, 1)$, for which $\|f\|^2 + \|g\|^2 = 1$, three orders of magnitude below the honest threshold. At that scale, the unamplified random term contributes at most $\|h_1 - h_1'\|^2 \approx n/2$, which falls far below the verification radius.

Flaw 3: an assumed honest key distribution. The S-CEO analysis estimates that a θ-ball holds about $2^{31 \cdot 3}$ integer points out of $2^{31 \cdot n}$, giving an acceptance probability around $2^{-31(n-3)}$. This count is computed specifically for honest-distributed (q_{00}, q_{01}). For our malformed key, with $q_{00} = 1$ rather than the honest $q_{00}[0] = \|f\|^2 + \|g\|^2 \geq \ell_{\text{low}}$, the same ball encompasses almost the entire signature space. The wNR analysis reuses this count verbatim, and the M-S-UEO bound relies on the same volume-counting argument, so all three inherit the gap. The 100 out of 100 wNR acceptance, using messages from a PRF unrelated to the malformed key, is the direct experimental witness for this flaw.

Flaw 4: an assumed honest preimage. The argument is parametrized by some basis B with $B^*B = Q$ and $fG - gF = 1$, of which several exist for a given (q_{00}, q_{01}). For honest keys, the minimum-size requirement pins all preimages to an honest scale, so the choice is harmless. For the malformed family, the verifier accepts preimages of arbitrarily small norm, including the degenerate $(1, 0, a, 1)$ basis of Flaw 1. The proof never quantifies over preimages and silently substitutes the honest one.

Consequence for the Düzlü–Struck narrative. Düzlü and Struck show that if a scheme Σ has MBS, then its PS-3 transform inherits S-UEO and a form of non-resignability, and they argue conceptually that MBS is the right formalization of "no effective weak keys" [7]. HAWK is structurally a PS-3-style scheme, with target $M = \mathsf{H}(m \| \mathsf{H}(pk))$, and a reader applying [7] would take ADMSW24's HAWK MBS check mark as the input hypothesis. Our MBS counterexample removes that hypothesis for HAWK. There is no contradiction inside [7], whose theorem is conditional on MBS; what disappears is HAWK as the canonical lattice example, since the malformed (pk, sig) is exactly the kind of effective weak key the narrative says MBS rules out.

Which games actually break. The four gaps above invalidate all four ADMSW24 Section 5.1 proofs on the verifier-accepted domain, but an unsound proof does not by itself imply a broken game. Section 3 exhibits counterexamples for three of the four games: MBS, M-S-UEO, and wNR. For S-CEO and S-DEO, no counterexample is known, and Section 6 explains why those games appear to survive the malformed family even though their proofs do not.

5 What Key Generation Enforces

Our counterexamples verify on the unmodified reference implementations, so the gap is between what key generation enforces and what verification accepts. This section records predicates that honest key generation already satisfies and that are cheap to test after decode. Section 5.1 defines KeyNormCheck, which matches key generation's norm floor, and KeyIdentityCheck, the finite algebraic condition under which the reference verifier's modular norm equals the intended quadratic form. We prove that every honestly generated key passes the norm floor exactly (Section 5.2). If enforced at verification time, this floor defeats the entire constant-q_{00} attack family from Section 3 for the intended form (Section 5.3). Section 5.5 reports the measurements behind these statements. Theorem 5.6 shows what a much stronger per-coordinate floor would yield for arbitrary keys in the same proof framework, but honest keys fail that condition. These predicates serve only to state formal guarantees, not as a specification proposal, and Section 6 details the gaps that remain.

5.1 The predicate

Definition 5.1 (KeyNormCheck). *For each parameter set fix the constant* $Q_{\min}$ *equal to* ℓ_{low}*, the smallest value of* $\|f\|^2 + \|g\|^2$ *that HAWK key generation*

*will output (*556/2080/7981 *for HAWK-*256/512/1024*). KeyNormCheck accepts a decoded public key iff* $q_{00}[0] \geq Q_{\min}$.

The check is a single signed comparison after public-key decode. It uses no transform, no extra allocation, and no per-coordinate loop, and it does not change the encoded public-key byte format.

Definition 5.2 (KeyIdentityCheck). *KeyIdentityCheck accepts a decoded public key iff* q_{00} *divides* $1 + q_{01}q_{01}^*$ *in* $\mathbb{Z}[X]/(X^n + 1)$ *and the quotient* $q_{11} = (1 + q_{01}q_{01}^*)/q_{00}$ *is an integral self-adjoint polynomial in the parameter-set coefficient range.*

This is a finite algebraic condition on the decoded coefficients. We separate it from KeyNormCheck because the norm floor is the bound key generation already applies, while KeyIdentityCheck is the condition under which PolyQnorm computes the quadratic form our theorems reason about.

Why a per-coordinate floor is a different bound. A natural first guess is to bound every Fourier coordinate from below, $\min_u q_{00,u} \geq \mu_{\min}$. This does not work. A constant key $q_{00} = K$, $q_{01} = a$ has all of its Fourier coordinates equal to K, so its smallest coordinate equals its average; a constant K chosen between the attack scale and the honest range therefore clears any per-coordinate bound calibrated to honest keys while still breaking MBS. On HAWK-256, for instance, $K = 8$ clears such a bound and still verifies two messages. The quantity that actually separates honest keys from the constant-key attacks is the *average* Fourier coordinate, which equals the constant term $q_{00}[0] = \|f\|^2 + \|g\|^2$, not the minimum.

5.2 Honest acceptance is exact

Theorem 5.3 (Honest KeyNormCheck acceptance). *For any of the three HAWK parameter sets and* $(sk, pk) \leftarrow \mathsf{KGen}$, $\Pr[pk \text{ passes } \mathsf{KeyNormCheck}] = 1$.

Proof. HAWK key generation rejection-samples (f, g) until $\|f\|^2 + \|g\|^2 \geq \ell_{\text{low}}$, with $\ell_{\text{low}} = 556/2080/7981$. By the identity $q_{00}[0] = \|f\|^2 + \|g\|^2$, every output satisfies $q_{00}[0] \geq \ell_{\text{low}}$. Since $Q_{\min} = \ell_{\text{low}}$ by Definition 5.1, the predicate holds unconditionally. □

Because $Q_{\min}$ equals the exact bound that key generation enforces, honest acceptance is exact rather than probabilistic. We confirmed this empirically over 16384 honest keys per parameter set: the observed minima were 556, 2080, and 7982, the last one above the HAWK-1024 bound 7981.

5.3 Constant-key MBS under the norm floor

The attacks of Section 3 all use a constant q_{00}, one whose Fourier coordinates are all equal. For this entire class of constant-q_{00} MBS attacks, enforcing KeyNormCheck at verification would be provably sufficient once the norm is evaluated as the intended quadratic form over the reals.

Lemma 5.4 (Parity anti-concentration). *For real z, $\frac{1}{2}\big(\mathrm{dist}_2(z)^2 + \mathrm{dist}_2(z+1)^2\big) \geq \frac{1}{4}$, where $\mathrm{dist}_2(x) = \min_{k\in\mathbb{Z}} |x - 2k|$. Hence for a uniform bit $h \in \{0,1\}$, $\mathbb{E}_h[\mathrm{dist}_2(z+h)^2] \geq \frac{1}{4}$.*

Proof. By reducing z modulo 2, we may assume $z \in [0,1]$, where the left side evaluates to $\frac{1}{2}(z^2 + (1-z)^2) = \frac{1}{2}(2z^2 - 2z + 1) \geq \frac{1}{4}$. □

Theorem 5.5 (Constant-key MBS under KeyNormCheck). *Let the adversary output (pk, sig, m, m') in the MBS game with constant $q_{00} = K$ and arbitrary q_{01}, s_1. Under KeyNormCheck we have $K \geq Q_{\min}$. If $K > 32\sigma^2_{\mathrm{verify}}$, then for a fresh random-oracle challenge the verifier accepts with probability at most*

$$p_K = \exp\big(-2n(\tfrac{1}{4} - 8\sigma^2_{\mathrm{verify}}/K)^2\big),$$

and so for two distinct messages with independent challenges and at most Q random-oracle queries, $\mathsf{Adv}^{\mathrm{MBS}}(A) \leq O(Q^2 p_K^2)$.

Proof. For constant $q_{00} = K$, the Cholesky identity (1) gives the rebuild summand $K\,\|w_0 + (q_{01}/K)w_1\|^2$. Condition on h_1, so $z := (q_{01}/K)(h_1 - 2s_1)$ is fixed. Algorithm 18 may pick any integer s_0, but $w_0[j] = h_0[j] - 2s_0[j]$, so $(w_0[j] + z_j)^2 \geq \mathrm{dist}_2(z_j + h_0[j])^2$. By Lemma 5.4 the independent variables $X_j = \mathrm{dist}_2(z_j + h_0[j])^2 \in [0,1]$ have conditional mean at least $1/4$, and the rebuild term of (1) is at least $K\sum_j X_j$. Acceptance requires $K\sum_j X_j \leq 8n\sigma^2_{\mathrm{verify}}$, that is $\sum_j X_j \leq 8n\sigma^2_{\mathrm{verify}}/K$. Hoeffding's lower-tail bound gives p_K whenever $8\sigma^2_{\mathrm{verify}}/K < 1/4$, that is $K > 32\sigma^2_{\mathrm{verify}}$. Two distinct messages give independent challenges, and the union bound over random-oracle queries gives the advantage. □

The thresholds satisfy $Q_{\min} > 32\sigma^2_{\mathrm{verify}}$ with wide margin. For HAWK-256/512/1024, $32\sigma^2_{\mathrm{verify}} \approx 34.7/65.0/79.0$, against $Q_{\min} = 556/2080/7981$. Theorem 5.5 closes the entire constant-q_{00} MBS attack class, which includes the family of Section 3 at $K = 1$. The same single-signature estimate is the main ingredient one would need for constant-key wNR and M-S-UEO as well, though we do not write out those full game reductions here; they additionally depend on the non-constant and arithmetic obligations isolated in Section 6.

5.4 A stronger condition: a floor under every Fourier coordinate

KeyNormCheck constrains only the average Fourier coordinate $q_{00}[0]$. If verification instead required a floor under *every* Fourier coordinate, the single-signature bound becomes immediate for arbitrary keys, constant or not, when the norm is evaluated exactly over the reals.

Theorem 5.6 (Floor under every Fourier coordinate). *Evaluate the verifier's norm exactly over the reals, and suppose a decoded public key has every Fourier coordinate at least μ, that is $q_{00,u} \geq \mu$ for all u, with $\mu > 32\sigma^2_{\mathrm{verify}}$. Fix any q_{01} and any signature body $(salt, s_1)$. Over a fresh random-oracle challenge (h_0, h_1), this exact norm test accepts with probability at most*

$$\exp\big(-2n(\tfrac{1}{4} - 8\sigma^2_{\mathrm{verify}}/\mu)^2\big).$$

Table 2. The KeyNormCheck threshold sits an order of magnitude above the constant-key MBS attack boundary and exactly at key generation's minimum.

set	$Q_{\min} = \ell_{\text{low}}$	constant-key MBS boundary	$32\sigma^2_{\text{verify}}$
HAWK-256	556	$K \leq 23$	34.7
HAWK-512	2080	$K \leq 13$	65.0
HAWK-1024	7981	$K \leq 13$	79.0

Proof. Condition on h_1, so $w_1 = h_1 - 2s_1$ and $z = (q_{01}/q_{00})w_1$ are fixed. For any integer s_0 chosen by Algorithm 18, $w_0[j] = h_0[j] - 2s_0[j]$, hence

$$(w_0[j] + z_j)^2 \geq \text{dist}_2(z_j + h_0[j])^2.$$

By Lemma 5.4, the independent variables $X_j = \text{dist}_2(z_j + h_0[j])^2$ lie in $[0, 1]$ and have conditional mean at least $1/4$. By the unitary normalization stated in Section 2.2, Parseval identifies $\sum_u |y_u|^2$ with the coefficient norm $\sum_j |y[j]|^2$. Since every Fourier coordinate is at least μ, the rebuild term of (1) is at least $\mu \sum_j X_j$, so acceptance implies $\sum_j X_j \leq 8n\sigma^2_{\text{verify}}/\mu$. Hoeffding's lower-tail inequality gives the stated bound. □

Theorem 5.6 is a complete conditional theorem for the norm evaluated exactly over the reals. A floor under every Fourier coordinate rejects many honest keys, as our calibration over honest keys confirms. It also shows that a much stronger spectral condition would reach arbitrary keys in the same proof framework, but honest key generation outputs keys that fail that stronger bound.

5.5 Calibration

Table 2 records the separation between key generation's minimum and the attack boundaries. Adversarial sweeps covered the constant family, sparse non-constant perturbations, and 75600 dense non-constant q_{00} shapes; they found zero MBS or wNR acceptances once $q_{00}[0] \geq Q_{\min}$, on all three parameter sets.

6 Proof Status and Open Problems

Our positive results assume the verifier enforces the bounds from Section 5. Theorem 5.3 is exact and unconditional on honest key generation: every output passes KeyNormCheck. Theorem 5.5 proves MBS security under KeyNormCheck for all keys with constant q_{00}, with the norm evaluated over the reals. Theorem 5.6 shows that a per-coordinate Fourier floor would reach arbitrary keys under the same framework, but honest keys fail that stronger requirement. The rest of this section records what remains open even if the verifier applies the norm floor.

The central open problem: uneven keys under the norm floor. Whether KeyNormCheck alone implies MBS for keys whose Fourier coordinates are not all equal is open. The constant case worked because $\sum_u q_{00,u} |y_u|^2 \geq q_{00}[0] \|y\|^2$, but this inequality holds only when the coordinates are equal: when they are uneven, a few small coordinates can absorb the quadratic form while the average stays above $Q_{\min}$, so Theorem 5.5 does not carry over. After conditioning on the random oracle, an accepted signature gives the event $e^\top Q_{00} e \leq 8n\sigma^2_{\text{verify}}$, where $e = a + \varepsilon/2$ is a fixed vector perturbed by independent $\pm 1/2$ signs. The sharp question is a counting problem: over all sign patterns $\varepsilon \in \{\pm 1\}^n$, how many push the form below the threshold,

$$\sup_a \left|\{\varepsilon \in \{\pm 1\}^n : \textstyle\sum_j \lambda_j |(U\varepsilon)_j - z_j|^2 \leq 4T\}\right|,$$

where the λ_j are the Fourier coordinates and U is the transform that diagonalizes Q_{00}. These operators are negacyclic, inherited from the ring $X^n + 1$, and that structure appears essential. We can settle the exact-zero case, that at most one sign pattern lands the form exactly at zero, so the probability of hitting it is at most 2^{-n}, but the positive-radius version that the verifier threshold actually needs is open. Extended adversarial search found no MBS or wNR counterexample once $q_{00}[0] \geq Q_{\min}$, including dense non-constant q_{00} shapes, but that empirical evidence does not close the proof gap.

S-CEO and S-DEO. We did not find S-CEO or S-DEO counterexamples for this malformed family, and extended searches returned none, even with message grinding and scaled constant keys. Unlike MBS or wNR, where the adversary fabricates both the key and the signature, the S-CEO and S-DEO games force the adversary to reuse an honest signature. Consequently, s_1 is a discrete-Gaussian output with expected norm $\|s_1\|^2 = \Theta(n)$, meaning the adversary can no longer arbitrarily set $s_1 = 0$. By splitting on the largest Fourier coordinate $\max_u q_{00,u}$, a two-branch argument recovers a heuristic exponential bound even for the unmodified verifier. When that coordinate is large, the rebuild term blows up; when it is small, the noise term blows up, driven by the honest $\|w_1\|^2 = \Theta(n)$ contribution. While this argument remains a sketch, and its large-coordinate branch relies on an unproven heuristic regarding the random oracle's image, it suggests that the S-CEO and S-DEO *games* themselves likely survive the constant malformed family. Nevertheless, as demonstrated by Flaws 3 and 4, the ADMSW24 *proofs* for these games remain fundamentally unsound over the verifier-accepted domain.

Relating the theorems to the reference verifier's modular norm. Our conditional theorems are stated for the norm evaluated exactly over the reals. Relating them to the integer N returned by PolyQnorm on the reference verifier is a separate finite question.

The modular norm must equal the intended quadratic form. Every positive result in this paper, including the constant-key theorem, assumes that the integer norm

N the verifier computes equals the quadratic form the proofs reason about. This is not automatic on the verifier-accepted domain. PolyQnorm computes N modulo two NTT primes and reassembles it by the Chinese remainder theorem. This matches the intended norm only when the decoded key passes KeyIdentityCheck and the integer result stays in range, $0 \leq N < p_1 p_2$. The range bound holds for the exact rebuild, and the key identity is a single polynomial check on the decoded coefficients. This identity really is needed: a decoded key such as $q_{00} = 2$, $q_{01} = 0$ has no integer $q_{11} = 1/2$, yet the unmodified verifier still accepts it. Our theorems are therefore phrased in terms of the intended norm on keys satisfying KeyIdentityCheck, not every byte string the current verifier happens to accept. We do not claim that enforcing KeyIdentityCheck at verification, with or without KeyNormCheck, restores BUFF security.

7 Discussion

Protocol implications. The weak-key surface matters wherever a system accepts user-supplied HAWK public keys and treats verification as evidence of key ownership. A proof-of-possession check that accepts "this public key verifies a challenge signature" is not a key-validity check: the family $q_{00} = 1$, $q_{01} = a$, $s_1 = 0$ passes it without any honest key-generation secret. A registration that demands an actual key-generation witness, or that enforces the same norm floor key generation already applies, rejects it. The same concern applies to signature aggregation protocols that admit arbitrary public keys, where a malformed key with a message-independent signature undermines per-signature binding.

What would restore BUFF security? Our results refute the stronger reading of ADMSW24 Remark 7, that PS-3 alone suffices for HAWK BUFF security. The generic BUFF transform of [2] provably yields the properties, at the cost of a larger signature. Whether enforcing key-generation bounds such as KeyNormCheck could substitute for that transform is not settled here. Theorem 5.6 shows that a much stronger per-coordinate floor would suffice for arbitrary keys in our proof framework, but honest keys fail that condition. Section 6 records the open proof gap for uneven keys even under the norm floor.

Conclusion. HAWK does not satisfy MBS, M-S-UEO, or wNR over the public-key domain that its verifier accepts, and the ADMSW24 proofs that claimed otherwise leave one case unanalyzed and, in three further places, silently assume the honest key norm and distribution that only key generation enforces. Honest key generation already enforces a norm floor that our attacks violate; enforcing that bound at verification would pass honest keys exactly and defeat the entire constant-key attack class for the intended quadratic form. We make no claim that doing so restores BUFF security. Reducing the remaining case of uneven keys to a concrete counting problem about negacyclic operators gives a self-contained target for follow-up work.

Acknowledgements

I thank LayerZero Labs for providing compute resources used for AI-assisted experimentation that helped discover and refine the attacks, and for iteration on the analysis of verifier acceptance. I also thank Giuseppe Vitto for proofreading and helpful feedback on this paper.

AI Disclosure

AI models, including GPT 5.2, 5.4 & 5.5, Opus 4.6, 4.7, & 4.8, Gemini 3.1 Pro, and Composer 2.5, were used to find the attacks, analyze verifier acceptance, and heavily edit the paper based on the author's guidance.

References

1. Aulbach, T., Düzlü, S., Meyer, M., Struck, P., Weishäupl, M.: Hash your keys before signing: BUFF security of the additional NIST PQC signatures. PQCrypto 2024. IACR ePrint 2024/591, https://eprint.iacr.org/2024/591 (2024)
2. Cremers, C., Düzlü, S., Fiedler, R., Fischlin, M., Janson, C.: BUFFing signature schemes beyond unforgeability and the case of post-quantum signatures. In: IEEE Symposium on Security and Privacy (S&P) (2021), iACR ePrint 2020/1525, https://eprint.iacr.org/2020/1525
3. Dao, Q.: HAWK weak-key counterexamples: Reproducibility artifact. https://github.com/quangvdao/hawk-weak-keys (2025)
4. Don, J., Fehr, S., Huang, Y.H., Liao, J.J., Struck, P.: Hide-and-seek and the non-resignability of the BUFF transform. IACR ePrint 2024/793, https://eprint.iacr.org/2024/793 (2024)
5. Ducas, L., Postlethwaite, E.W., Pulles, L.N., van Woerden, W.: Hawk: Module LIP makes lattice signatures fast, compact and simple. In: ASIACRYPT (2022), iACR ePrint 2022/1155, https://eprint.iacr.org/2022/1155
6. Düzlü, S., Fiedler, R., Fischlin, M.: BUFFing FALCON without increasing the signature size. IACR ePrint 2024/710, https://eprint.iacr.org/2024/710 (2024)
7. Düzlü, S., Struck, P.: The role of message-bound signatures for the beyond Un-Forgeability features and weak keys. ISC 2024. IACR ePrint 2024/1669, https://eprint.iacr.org/2024/1669 (2024)
8. Jackson, D., Cremers, C., Cohn-Gordon, K., Sasse, R.: Seems legit: Automated analysis of subtle attacks on protocols that use signatures. In: ACM CCS (2019), iACR ePrint 2019/779
9. National Institute of Standards and Technology: Status report on the second round of the additional digital signature schemes for the NIST post-quantum cryptography standardization process. NIST IR 8610, NIST (May 2026), https://csrc.nist.gov/pubs/ir/8610/final
10. Pornin, T., Stern, J.P.: Digital signatures do not guarantee exclusive ownership. In: Applied Cryptography and Network Security (ACNS) (2005)
11. Ristenpart, T., Yilek, S.: The power of proofs-of-possession: Securing multiparty signatures against rogue-key attacks. In: EUROCRYPT (2007)

12. Team, T.H.: HAWK: Module-LIP based signature scheme. NIST additional signatures round 2 specification, v1.1. https://csrc.nist.gov/csrc/media/Projects/pqc-dig-sig/documents/round-2/spec-files/hawk-spec-round2-web.pdf (2025)