Cofactor-torsion attacks on hinted scalar multiplications in SNARK circuits

Youssef El Housni

Consensys, Linea
youssef.elhousni@consensys.net

Abstract. Proving an elliptic curve scalar multiplication $[k]P = Q \in E(\mathbb{F}_p)$ inside a SNARK is much cheaper when the output Q is *hinted* by the prover and only *verified* in-circuit, rather than recomputed. The recent scalar multiplication techniques of Eagen, El Housni, Masson and Piellard (Latincrypt 2025) certify a hinted Q with a short lattice reduction (fraction) decomposition of the scalar and a single group identity, and yield the fastest known in-circuit scalar multiplications. We observe that the soundness of this whole family rests on an *implicit prime-order hypothesis*: the certifying identity is checked over the full group $E(\mathbb{F}_p)$, and only when $E(\mathbb{F}_p)$ has trivial cofactor does it force Q into the prime-order subgroup. On *cofactor* curves the certification is unsound. We give two concrete forgery classes against these hinted gadgets: an *any-scalar* attack that, for a target scalar fixed by the statement, adapts the decomposition so a small rational torsion point cancels from the identity; and a *chosen-scalar* attack that vanishes the output-side coefficients modulo a small cofactor prime and solves for the scalar afterwards. Both make the gadget accept $Q' = [k]P + T$ for a torsion point $T \neq 0_E$ as if it were $[k]P$. We quantify the reachable torsion in terms of the sub-scalar range bound and validate the attacks on widely deployed curves such as BLS12-381, BN254 and BW6-761. The straightforward fix is to check that Q lies in the subgroup but is expensive. We propose a cheaper one that binds the hinted output through a hinted preimage, with the minimal such constant that suffices against each attack model.

1 Introduction

Scalar multiplication over an elliptic curve is one of the most expensive computations proved inside SNARK circuits, and a central target for optimization [8,3,4]. A key insight is that, in a SNARK, a computation can be *verified* rather than *recomputed*: the prover supplies the result $Q = [k]P$ as a non-deterministic advice value (a *hint*), and the circuit only checks a relation that pins Q down. The state-of-the-art techniques of Eagen, El Housni, Masson and Piellard [4] (hereafter the *hinted-output* methods) push this idea furthest. Given the hinted output $Q = [k]P$, they use a lattice-reduction (fraction) decomposition of the scalar to replace the N-bit scalar multiplication by a *shorter* multi-scalar-multiplication that must vanish. Without an efficient endomorphism, proving $[k]P = Q$ reduces to an $N/2$-bit double-scalar-multiplication $[x]P - [z]Q = 0_E$; and on curves

carrying an efficient endomorphism ψ of small complex-multiplication discriminant (e.g. j-invariant 0), it reduces to an $N/4$-bit quadruple-scalar-multiplication $[x]P + [y]\psi(P) - [z]Q - [t]\psi(Q) = 0_E$. Both cases extend to a simultaneous decomposition of several scalars (multi-scalar multiplication). Across these variants the methods give, according to [4], between 22% and 53% fewer constraints than the previous state of the art. These gadgets are widely deployed — for instance in the `gnark` library [2], where they underlie in-circuit group operations on the elliptic curves BLS12-381, BW6-761 and BN254.

The implicit hypothesis. The certifying identity of every hinted-output method has the shape

$$\sum_{i=1}^{n} \big([x_i]P_i + [y_i]\psi(P_i)\big) - [z]Q - [t]\psi(Q) = 0_E$$

where $n = 1$ for the single-scalar methods and $n = 2$ for the joint ones. For an honest $Q = \sum_{i=1}^{n}[k_i]P_i$, with each P_i of prime order r and $\psi(P_i) = [\lambda]P_i$, it collapses to $\sum_{i=1}^{n}[x_i + \lambda y_i - k_i(z + \lambda t)]P_i = 0_E$, so a correct short decomposition forces $Q = \sum_{i=1}^{n}[k_i]P_i$. Crucially, *this argument requires the hinted output Q — not only the inputs P_i — to lie in the order-r subgroup* $\mathbb{G}$: the identity is read over the whole group $E(\mathbb{F}_p)$, so if $Q \notin \mathbb{G}$ a nonzero contribution from points outside $\mathbb{G}$ survives. The method of [4] is phrased over $\mathbb{G}$ and its implementation checks that the *inputs* P_i lie in $\mathbb{G}$; but the hinted *output* Q is pinned only by the identity above and is never subjected to a subgroup check. On a prime-order curve this omission is invisible, since $E(\mathbb{F}_p) = \mathbb{G}$ makes $Q \in \mathbb{G}$ automatic; but it is stated over $\mathbb{G}$ in general and benchmarked on *cofactor* curves — notably BLS12-381 and BW6-761 [4] — where nothing constrains the hinted Q to $\mathbb{G}$.

Our contribution. We show that on *cofactor* curves — where $\#E(\mathbb{F}_p) = h \cdot r$ with $h > 1$ — the hinted-output certification is *unsound*, because nothing binds the hinted output Q into the prime-order subgroup. Concretely:

- We describe two forgery classes (Section 3). The *any-scalar* attack (Section 3.2) works for a target scalar k fixed by the statement, using a rational torsion point T whose order divides a small cofactor prime; the attacker adapts the decomposition so T cancels from the identity. The *chosen-scalar* attack (Section 3.3) vanishes the output-side coefficients modulo a cofactor prime and solves for k. In its endomorphism-agnostic form it reaches every cofactor prime below $r^{1/4}$; with a rational eigenvector it reaches every prime below $r^{1/2}$ — a subtlety that also fixes the minimal clearing constant needed by the countermeasure.
- We quantify which torsion orders each attack can reach, as a lattice-index function of the range check on the sub-scalar bit-sizes, and validate the attacks on the BLS12-381 $\mathbb{G}_1/\mathbb{G}_2$, BN254 $\mathbb{G}_2$ and BW6-761 $\mathbb{G}_1/\mathbb{G}_2$ (Section 5.1), where $\mathbb{G}_1$ and $\mathbb{G}_2$ are respectively the first and second pairing groups.

– We give a (in most cases) better fix than subgroup membership applied to the hinted point (Section 4): bind the hinted output into the subgroup via a hinted preimage S with $[h']S = Q$ and S on-curve, together with the minimal clearing constant h' that suffices against each attack model.

2 Preliminaries

2.1 Elliptic curves and cofactors

Let E be an ordinary elliptic curve over $\mathbb{F}_p$ with $\#E(\mathbb{F}_p) = p + 1 - t = h \cdot r$, where r is a large prime, h is the *cofactor*, and $\mathbb{G} = E(\mathbb{F}_p)[r]$ is the prime-order subgroup. We write $N = \lceil \log_2 r \rceil$ for the bit-size of r and 0_E for the point at infinity. When a small prime ℓ divides h, the curve has a nonzero rational point T of order ℓ *outside* $\mathbb{G}$; we call such a T a *torsion point*. More generally, suppose that E admits an efficiently computable endomorphism ψ that preserves $\mathbb{G}$. Since $\mathbb{G}$ is cyclic of prime order, the restriction of ψ to $\mathbb{G}$ is scalar multiplication by some $\lambda \in \mathbb{F}_r$, i.e. $\psi(P) = [\lambda]P$ for every $P \in \mathbb{G}$. The endomorphism-based methods below use only this property (together with a low-degree relation for λ), rather than any particular formula for ψ.

A standard GLV example [7] is a curve with j-invariant 0, $E : y^2 = x^3 + b$. If the ground field contains a primitive cube root of unity ω, then $\psi : (x, y) \mapsto (\omega x, y)$ is efficient and its eigenvalue λ on $\mathbb{G}$ is a primitive cube root of unity in $\mathbb{F}_r$. Moreover, $T_0 = (0, \sqrt{b})$, when rational, has order 3 and is *fixed* by ψ, since $\psi(T_0) = (\omega \cdot 0, \sqrt{b}) = T_0$.

A second source is the GLS construction [6] for curves over extension fields. If a curve over $\mathbb{F}_{p^d}$ is obtained from a curve over $\mathbb{F}_p$ by twisting, one transports the p-power Frobenius through the twisting isomorphism to obtain an efficiently computable endomorphism ψ; in twist coordinates it consists essentially of field Frobenius maps and multiplications by constants. On an invariant prime-order subgroup it again acts as $[\lambda]$. This is the standard situation for the $\mathbb{G}_2$ groups of BN and BLS12 curves, which are represented on twists over $\mathbb{F}_{p^2}$.

2.2 Hinted-output scalar multiplication

We recall the four in-circuit methods of [4]. In each, the prover hints the output Q, and the circuit checks a fraction decomposition of the scalar together with one group identity. For the endomorphism variants (dim. 4, 6) that our attack targets, each sub-scalar of nominal size $r^{1/4}$ is range-checked to $\approx N/4 + 2$ bits, i.e. below $2^{N/4+2}$.

Integer fraction decomposition (dim. 2). Write $k = x/z \bmod r$ with x, z of size $\approx r^{1/2}$. Then $[k]P = Q \iff [x]P - [z]Q = 0_E$.

Simultaneous fraction decomposition (dim. 3). For a two-scalar $[k_1]P_1 + [k_2]P_2$, write $k_1 = x_1/z$, $k_2 = x_2/z \bmod r$. Then $[k_1]P_1 + [k_2]P_2 = Q \iff [x_1]P_1 + [x_2]P_2 - [z]Q = 0_E$.

Fraction decomposition in a quadratic extension (dim. 4). With the endomorphism, write $k = (x + \lambda y)/(z + \lambda t) \bmod r$ — a fraction in $(\mathbb{Z}/r\mathbb{Z})[\lambda]$ — with x, y, z, t of size $\approx r^{1/4}$. Then

$$[k]P = Q \iff [x]P + [y]\psi(P) - [z]Q - [t]\psi(Q) = 0_E. \tag{1}$$

Simultaneous fraction decomposition in a quadratic extension (dim. 6). The two-scalar analogue, with $k_1, k_2 = (x_i + \lambda y_i)/(z + \lambda t)$ and identity $[x_1]P_1 + [y_1]\psi(P_1) + [x_2]P_2 + [y_2]\psi(P_2) - [z]Q - [t]\psi(Q) = 0_E$.

In every case the circuit additionally binds the decomposition to the scalar (e.g. $k \cdot z = x \bmod r$ for the simple case) and guards the output-side denominator against zero (e.g. $z \neq 0$). The hinted output Q, however, is pinned only by the identity above and never checked for subgroup membership — the gap we exploit in Section 3.

3 The attacks

We present two families of forgeries against the hinted-output methods of [4]. Both exploit the same gap — the hinted output Q is never bound to $\mathbb{G}$ — to make the certifying identity accept a *torsion-shifted* output $Q' = Q + T$ with $T \neq 0_E$. They differ in the adversary's power over the scalar: the *any-scalar* family works for a target fixed by the statement (the attacker only adapts the decomposition), whereas the *chosen-scalar* family lets the attacker also pick the scalar. Both apply, in principle, to all four methods of [4]; what differs is their *reachability* — which cofactor primes ℓ each can exploit on a given curve — which we quantify and tabulate per family below.

3.1 Core observation

Write the output as a multi-scalar sum $Q = \sum_{i=1}^{n} [k_i]P_i$, with $n = 1$ for the single-scalar methods and $n = 2$ for the joint ones. All four methods of [4] hint Q and verify one identity of the common shape

$$\sum_{i=1}^{n} \big([x_i]P_i + [y_i]\psi(P_i)\big) \;=\; [z]\,Q + [t]\,\psi(Q), \tag{2}$$

where the endomorphism terms appear only for the small-discriminant variants (dim. 4 and 6; otherwise $y_i = t = 0$), and (x_i, y_i, z, t) is the short fraction decomposition, separately bound to the k_i. With every $P_i \in \mathbb{G}$ and $Q = \sum_i [k_i]P_i \in \mathbb{G}$, identity (2) holds and pins the output.

Now fix a small prime $\ell \mid h$ and a rational torsion point $T \in E(\mathbb{F}_p)[\ell]$, and let the prover hint $Q' = Q + T \neq Q$. As (2) is read over the whole group $E(\mathbb{F}_p)$, substituting Q' leaves the input side and the honest part of the output side unchanged and produces the single residual

$$[z]\,T + [t]\,\psi(T) \;\in\; E[\ell]. \tag{3}$$

The forgery is accepted *iff the attacker makes the output-side coefficients* (z, t) *vanish this residual.* The two families are the two ways of doing so.

Relation to small-subgroup attacks. This is a proof-forgery analogue of the classical small-subgroup attack [10], specialized to the composite-order ambient groups arising on pairing-friendly curves [1]. In the classical attack, an adversary supplies a point T of order ℓ, and observing $[a]T$ reveals information about a secret exponent a mod ℓ. Here the unchecked element is instead the prover-hinted output $Q' = Q + T$, and the verifier applies the public operator $[z] + [t]\psi$. By arranging $[z]T + [t]\psi(T) = 0_E$, the prover places T in the kernel of the certifying identity, causing acceptance rather than secret leakage. Output-subgroup binding therefore plays the role of classical public-key validation.

Running example. We illustrate throughout on $E/\mathbb{F}_{103} : y^2 = x^3 + 4$, a cofactor sibling of the prime-order example of [4, Ex. 3]: here $\#E = 111 = 3 \cdot 37$, so $h = 3$ and $r = 37$; the endomorphism has $\lambda = 26$ and $\omega = 46$; $T_0 = (0, 2)$ is a 3-torsion point with $\psi(T_0) = T_0$; and $P = (76, 32)$ generates $\mathbb{G} = E(\mathbb{F}_{103})[37]$. All forgeries below were verified exhaustively over $E(\mathbb{F}_{103})$.

3.2 Any-scalar forgery

Here the scalar is fixed, so the attacker keeps the honest scalar relation and only steers the decomposition into the sublattice on which (3) vanishes.

Mechanism. The decompositions come from a rank-m lattice, where $m \in \{2, 3, 4, 6\}$ is the method dimension, with honest vectors of size $\approx 1.25\, r^{1/d}$ ($d = 2$ for the integer methods, $d = 4$ for the endomorphism ones). Requiring the residual (3) to vanish confines the vector to a sublattice: if c independent congruences modulo ℓ are imposed, this sublattice has index ℓ^c and its shortest vector is $\approx \ell^{\,c/m}$ times longer than the honest one. The range check leaves a fixed slack ρ over the honest length. This slack is not 1 because the circuit does not check against the typical geometric length $\approx 1.25\, r^{1/4}$ itself: it uses the integer bit bound $2^{\lceil N/4 \rceil + 2}$. Thus

$$\rho := \frac{2^{\lceil N/4 \rceil + 2}}{1.25\, r^{1/4}} > 1.$$

Even before rounding, the two extra range-check bits contribute a factor 4, giving $4/1.25 \approx 3.2$; rounding $N/4$ up to a whole bit contributes the additional curve-dependent factor $2^{\lceil N/4 \rceil - N/4}$. A forgery therefore fits precisely when

$$\ell^{\,c/m} \lesssim \rho \qquad \Longleftrightarrow \qquad \ell \lesssim \rho^{\,m/c}. \tag{4}$$

The short vector in either sublattice below is found by the Lenstra–Lenstra–Lovász algorithm (LLL) [9] on a rank-4–6 lattice. The residual is vanished by one of two routes. On the *eigen route* ($c = 1$), if T is a rational ψ-eigenvector $\psi(T) = [\mu]T$, the residual is $[z + \mu t]T$ and the single condition $z + \mu t \equiv 0 \pmod{\ell}$ suffices; this needs the endomorphism, so it applies to the dim. 4/6 methods and reaches $\ell \lesssim \rho^4$ and $\ell \lesssim \rho^6$, respectively. On the *both-zero route*, with no eigenstructure one forces every output coefficient to vanish — $z \equiv t \equiv 0 \pmod{\ell}$ for dim. 4/6 ($c = 2$, $\ell \lesssim \rho^2$), or $z \equiv 0 \pmod{\ell}$ for the integer dim.

2/3 methods ($c = 1$, $\ell \lesssim \rho^2$ resp. ρ^3) — and any rational ℓ-torsion works. Simply scaling an honest vector by ℓ is the crudest way to land in the both-zero sublattice (penalty ℓ against the LLL-reduced $\ell^{1/2}$), so on its own it reaches only the smallest primes. Since $z \equiv t \equiv 0$ implies $z + \mu t \equiv 0$, the both-zero sublattice lies *inside* the eigen one whenever μ exists, and a scaled vector is then simultaneously both. The eigen route is therefore stronger than the both-zero route whenever such an eigenvector exists.

Conditions. The any-scalar forgery succeeds for a cofactor prime $\ell \mid h$ when *(i)* $E(\mathbb{F}_p)$ has a rational ℓ-torsion point (automatic for $\ell \mid h$); *(ii)* on the eigen route, that point is a ψ-eigenvector; and *(iii)* ℓ meets the reachability bound (4). Whether condition (ii) holds depends on the action of ψ on the rational ℓ-torsion and is instantiated in Section 5.1.

Attack 1 (any-scalar, running example) *Take $k = 11$ with honest output $Q = [11]P$ and honest decomposition $(x, y, z, t) = (-1, -1, 0, -1)$, for which (3) evaluates to $[0]T_0 + [-1]\psi(T_0) = -T_0 \neq 0_E$, so $Q' = Q + T_0$ is caught. Scaling to $(-3, -3, 0, -3)$ preserves the scalar relation ($\gcd(3, r) = 1$) and gives residual $[-3]T_0 = 0_E$: the gadget now accepts $Q' = [11]P + T_0$ as $[11]P$. Here the two routes coincide — $\mu = 1$ and the scaled $(z, t) = (0, -3)$ has $z \equiv t \equiv 0 \pmod 3$ — so this is at once an eigen and a both-zero forgery.*

3.3 Chosen-scalar forgery

Here the attacker controls (or witnesses) the scalar, so it fixes the output-side coefficients *first* and solves for a compatible k afterwards.

Mechanism. Fix the output-side coefficients *first* and solve for k. The simplest choice puts one coefficient equal to ℓ: $(z, t) = (0, \ell)$ for the endomorphism methods, $z = \ell$ for the integer ones, so the residual (3) is $[\ell]\psi(T) = 0_E$ (resp. $[\ell]T = 0_E$) automatically, for *any* rational ℓ-torsion T. The scalar relation then fixes k: with a minimal input side ($[1]P_1$) the identity forces $k = (\lambda\ell)^{-1} \bmod r$ (endomorphism) or $k = \ell^{-1} \bmod r$ (integer). This is endomorphism-agnostic and, since ℓ need only fit one coefficient, reaches every cofactor prime $\ell < 2^{N/4+2}$.

Crucially, once k is *derived* rather than given, the *input*-side coefficients are unconstrained — any (x, y) yields a compatible k — so only the output pair (z, t) bounds ℓ, and one may use the eigen route there. If $\psi(T) = [\mu]T$, the residual vanishes under $z + \mu t \equiv 0 \pmod \ell$ — a rank-2 lattice of determinant ℓ whose shortest vector is $\approx \sqrt{\ell}$. Both z, t then fit the range as soon as $\sqrt{\ell} < 2^{N/4+2}$, i.e.

$$\ell \;<\; 2^{\,N/2+4} \;\approx\; r^{1/2}, \tag{5}$$

quadratically larger than the endomorphism-agnostic $2^{N/4+2} \approx r^{1/4}$. (Contrast the any-scalar case, where k is fixed: the input side is then pinned to a coset, the full rank-4 lattice must be balanced, and ℓ is capped at $\approx \rho^4$.) This distinction matters for the fix (Section 4): a clearing constant sized to $2^{N/4+2}$ leaves eigen-reachable torsion in $[2^{N/4+2}, 2^{N/2+4})$ unbound.

Conditions. The chosen-scalar forgery succeeds when *(i)* $\ell \mid h$ with a rational ℓ-torsion point; *(ii)* $\ell < 2^{N/4+2}$, or $\ell < 2^{N/2+4}$ when that point is a ψ-eigenvector; and *(iii)* the attacker can realize the forced scalar k in the statement. Condition (iii) is discrete-log-hard in most protocols, so this class is chiefly of theoretical concern; but a library primitive must be sound for *all* scalars, so the conservative reading defends against every reachable ℓ.

Attack 2 (chosen-scalar, running example) *Take* $(x, y, z, t) = (1, 0, 0, 3)$*, so the output side is* $[0]Q' + [3]\psi(Q')$ *and* $[3]\psi(T_0) = 0_E$*. The scalar relation then forces* $k = (3\lambda)^{-1} = 28 \bmod 37$*, and* $Q' = [28]P + T_0$ *is accepted.*

4 The fix

The sound repair is to re-establish, on cofactor curves, the property prime order gave for free: *the hinted output lies in* $\mathbb{G}$. Checking membership directly is expensive. Instead we bind Q into the subgroup through a *hinted preimage*, which replaces the output hint rather than adding to it. Let h' be a constant invertible modulo r (chosen below). The prover additionally hints $S = [k \cdot h'^{-1} \bmod r]\, P$, and the circuit checks

$$S \text{ on-curve}, \qquad [h']\, S = Q. \tag{6}$$

A torsion-tainted $Q' = [k]P + T$ with $\mathrm{ord}(T) \mid h'$ has *no* on-curve preimage under $[h']$, so no witness S exists and the forgery is rejected; an honest $Q \in \mathbb{G}$ has the preimage $S = [kh'^{-1}]P$. The on-curve constraint on S is essential: without it the doublings in $[h']S$ are unconstrained.

The constant h' must clear every *reachable* torsion. Against the *any-scalar* attack it need only cover the prime powers satisfying (4). The *chosen-scalar* attack is more demanding: by (5) it reaches every cofactor prime power below $2^{N/4+2}$, and below $2^{N/2+4} \approx r^{1/2}$ for those carrying a ψ-eigenvector. Being sound for all scalars, one takes

$$h' = \prod_{\substack{\ell^a \,\|\, h \\ \ell^a \text{ reachable}}} \ell^a,$$

with reachability at the $2^{N/2+4}$ bound for eigen-primes. Thus h' can be smaller than the full cofactor when the reachable part is small and separable; when every cofactor component is reachable, one must take $h' = h$. In the latter case, a method that computes rather than hints the output may be cheaper than adding the binding (6). Section 5.1 instantiates these choices on the deployed curves. On the toy curve of Section 3.1, $h' = 3$ is the whole cofactor: the honest $Q = [11]P$ has preimage $S = (77, 91)$ with $[3]S = Q$, whereas neither forged output $[11]P + T_0$ nor $[28]P + T_0$ admits an on-curve preimage under [3], so both are rejected.

5 Implementation

The methods of [4] are deployed in `gnark` [2] over many curves including the pairing-friendly curves BLS12-381, BW6-761 and BN254. The five groups with a nontrivial cofactor — BLS12-381 $\mathbb{G}_1/\mathbb{G}_2$, BW6-761 $\mathbb{G}_1/\mathbb{G}_2$ and BN254 $\mathbb{G}_2$, all using the dim. 4/6 methods — are exposed. We first quantify which torsion each attack reaches on these groups (Section 5.1), then describe how our proof of concept realizes the forgeries (Section 5.2) and the fix, with its constraint cost (Section 5.3).

5.1 Reachability and impact on deployed curves

Table 1 instantiates the two reachability criteria on the affected groups. The exact slack from Section 3.2 is $\rho \approx 3.8$ for BLS12-381/BN254 and $\rho \approx 5$ for BW6-761, rather than the uniform asymptotic estimate $\rho \approx 3.2$, because the range width is rounded to a whole number of bits. We therefore confirm every boundary case in the table by an explicit LLL computation.

Group	cofactor small part	any-scalar ℓ	chosen-scalar ℓ
BLS12-381 $\mathbb{G}_1$	$3 \cdot 11^2 \cdot 10177^2 \cdot 859267^2 \cdot 52437899^2$	$3, 11$	all
BW6-761 $\mathbb{G}_1$	$2^2 \cdot 127$	$2, 127$	all
BLS12-381 $\mathbb{G}_2$	$13^2 \cdot 23^2 \cdot 2713 \cdot 11953$	$13, 23$	$13, 23, 2713, 11953$
BW6-761 $\mathbb{G}_2$	$3 \cdot 13$	$3, 13$	all
BN254 $\mathbb{G}_2$	10069	—	10069

Table 1. Reachability on the affected `gnark` groups. "Cofactor small part" lists the prime powers below $2^{N/4+2}$. Any-scalar entries were confirmed by explicit LLL computations; chosen-scalar entries follow (5). "All" means that every cofactor component is reachable, and "—" that none is reachable.

Concrete instances. We instantiate every affected group; the range width is 66 bits for the BLS12-381/BN254 groups ($N = 255$, honest sub-scalars ≈ 64 bits) and 97 bits for BW6-761 ($N = 377$, ≈ 94 bits). The sub-scalar sizes reported below are the shortest vectors found by LLL.

- **BLS12-381** $\mathbb{G}_1$ ($3{\cdot}11^2{\cdot}10177^2{\cdot}859267^2{\cdot}52437899^2$). Any-scalar reaches $\ell = 3$ and $\ell = 11$. For $\ell = 3$ the ψ-fixed 3-torsion gives $\mu = 1$; the index-3 sublattice $z + t \equiv 0 \pmod 3$ has a vector only $\approx 1.3\times$ honest, so *any* statement scalar is forgeable. For $\ell = 11$ ($11 \equiv 2 \pmod 3$, no eigenvalue) the both-zero route $z \equiv t \equiv 0 \pmod{11}$ still fits, at ≈ 65 bits. The larger $\ell = 10177$ overflows even the eigen route ($10177^{1/4} \approx 10$), so it — like the remaining factors $859267, 52437899$ of h_1 — is chosen-scalar only: $(x, y, z, t) = (1, 0, 0, 10177)$, $k = (10177\,\lambda)^{-1}$. All factors of h_1 lie below $2^{N/4+2}$, so chosen-scalar reaches *all*.

- **BLS12-381** $\mathbb{G}_2$ ($13^2 \cdot 23^2 \cdot 2713 \cdot 11953$). Any-scalar reaches $\ell = 13$ (eigen, $\mu = 3$; the 2-dimensional rational torsion forces the eigenvector to be projected out via $(\psi - [9])$, ≈ 65 bits) and $\ell = 23$ (both-zero, landing right at the 66-bit bound). The larger 2713 and 11953 overflow by either route and are chosen-scalar only.
- **BN254** $\mathbb{G}_2$ (10069). The sole small prime is beyond any-scalar reach ($10069^{1/4} \approx 10 \gg \rho$), yet chosen-scalar hits it directly: $t = 10069 < 2^{66}$, $k = (10069\,\lambda)^{-1}$. (BN254 $\mathbb{G}_1$ is prime-order, hence immune.)
- **BW6-761** $\mathbb{G}_1$ ($2^2 \cdot 127$). The wider 97-bit range lets any-scalar reach *both* primes: $\ell = 2$ by the both-zero route (≈ 94 bits) and $\ell = 127$ by the eigen route ($127 \equiv 1 \pmod 3$, $\mu = 107$ on the 1-dimensional 127-torsion, ≈ 95 bits). The latter sits above the naive asymptotic cutoff ($\rho \approx 3.2$, $\ell \lesssim 100$) but well inside the exact range ($\rho \approx 5$, $\ell \lesssim \rho^4 \approx 625$). Chosen-scalar reaches further: with k derived the input side is free, so the eigen route on (z, t) alone (shortest $\approx \sqrt{\ell}$) lifts the bound to $\approx r^{1/2}$ (2^{194}, twice the 97-bit range) and covers the prime factors of the ≈ 330-bit cofactor remainder — *every* cofactor prime is chosen-reachable, which is why the sound fix here must clear the full cofactor.
- **BW6-761** $\mathbb{G}_2$ ($3 \cdot 13$, over $\mathbb{F}_p$). Any-scalar reaches $\ell = 3$ (scaling) and $\ell = 13$ (eigen, $\mu = 3$ on the 1-dimensional 13-torsion), both at ≈ 95 bits; as on $\mathbb{G}_1$, chosen-scalar reaches the whole ≈ 330-bit cofactor via the $2^{N/2+4}$ eigen bound.

Resulting clearing constants. Applying the product rule of Section 4, the chosen-scalar model allows BLS12-381 $\mathbb{G}_2$ to omit its unreachable 448-bit cofactor prime and use $h' = 13^2 \cdot 23^2 \cdot 2713 \cdot 11953$; BN254 $\mathbb{G}_2$ similarly uses $h' = 10069$ and omits its unreachable 178-bit prime. Every cofactor component is chosen-reachable for BLS12-381 $\mathbb{G}_1$ and for BW6-761 $\mathbb{G}_1/\mathbb{G}_2$, so these groups require the full cofactor. In the latter cases, classical GLV [7], which computes rather than hints the output, can be cheaper than adding the binding (6); the measured comparison appears in Section 5.3.

Across all groups the forgeries are unconditional at the gadget level; only condition (iii) — realizing the forced k inside a meaningful statement, a discrete-log problem — separates the theoretical chosen-scalar class from the practical any-scalar one.

5.2 Exploiting the gadget

Both the attacks and the fix were implemented against the `gnark` reference implementation [2] of [4], in Go; the findings were disclosed to the maintainers and patched. A minimal, self-contained proof of concept is available in [5]:

`https://github.com/yelhousni/scalarmul-in-snark-attack`

The hinted output is produced by a solver *hint* — an out-of-circuit routine that returns Q and its fraction decomposition as non-deterministic advice. A

malicious prover need not touch the circuit at all: it suffices to replace two hints, which in `gnark` is a one-line override each (`solver.OverrideHint`). The decomposition hint (`rationalReconstructExt` / `...G2`) is made to return the forged decomposition (x, y, z, t) (named (u_1, u_2, v_1, v_2) in the code), and the output hint (`scalarMulHint` / `scalarMulG2Hint`) the torsion-shifted point $Q' = Q + T$. Compiling the gadget and calling `ccs.IsSolved` on this witness then *succeeds*: the constraint system accepts $Q' \neq Q$ as the scalar-multiplication output. The artifact drives the public `ScalarMul` entry point on every affected group and instantiates both forgery families for every reachable cofactor prime listed in Table 1. For any-scalar forgeries it selects the scaling, eigen, or both-zero route as applicable and reuses `gnark`'s decomposition reducer for the required LLL computation. The $\mathbb{G}_2$ tests apply the same hint overrides to the tower-field gadget and generate rational torsion points on the fly. In every case, the unpatched constraint system accepts the torsion-shifted output.

5.3 The fix and its cost

The fix in code. The repair is a single point-level routine, `assertPointInSubgroup(R)`, applied once to the hinted output — or, in a multi-scalar multiplication, once to the aggregate sum. This suffices because the certifying identity has only one output point: forcing that aggregate into $\mathbb{G}$ removes its entire torsion component, so the identity is again evaluated inside the prime-order subgroup. It requests a preimage $S = [h'^{-1} \bmod r]\, R$ from a new hint, asserts that S is on the curve, and enforces $[h']\, S = R$ through a windowed constant multiplication (`mulByConstant`) by the per-curve clearing constant h' of Section 4, stored as `CofactorClearing` in the curve parameters; the $R = 0_E$ case is handled by a data-independent select. The artifact demonstrates this binding on *every* affected group: it supplies $S = [h'^{-1} \bmod r]R$ for the group's clearing constant h' (the product of the prime powers reachable at the $2^{N/2+4}$ chosen-scalar bound of Section 4), asserts S on-curve, and enforces $[h']S = R$ by a sound double-and-add built from the complete addition, whereupon every forgery (chosen- and any-scalar) is rejected while the honest witness still solves. For BLS12-381 $\mathbb{G}_2$ and BN254 $\mathbb{G}_2$, h' is a small constant (the large cofactor prime is unreachable). For BLS12-381 $\mathbb{G}_1$ and BW6-761 $\mathbb{G}_1/\mathbb{G}_2$, however, essentially the whole cofactor is reachable, so the binding must clear a large ($\approx$126- to 330-bit) constant; there the sound hinted variant is dearer than a classical GLV scalar multiplication, so the public entry points route to classical GLV, which needs no binding since it *computes* rather than hints the output.

Constraint cost. Table 2 reports R1CS (Groth16) and SCS (PLONK) counts for one emulated scalar multiplication. For every group deployed with classical GLV — BLS12-381 $\mathbb{G}_1$ and BW6-761 $\mathbb{G}_1/\mathbb{G}_2$ — the sound classical-GLV gadget is *smaller* than the original unsound hinted one (371k vs 589k R1CS on BW6-761 $\mathbb{G}_1$; 136k vs 237k on BLS12-381 $\mathbb{G}_1$), so the fix removes the hinted off-subgroup output *and* shrinks the circuit. The hinted-preimage binding goes the other way: on BW6-761 $\mathbb{G}_2$ the previous method is 896k R1CS (2.4× classical

GLV) — and was itself under-sized (Section 3.3), a sound binding over the full ≈ 330-bit cofactor being larger still. Binding stays worthwhile only where the reachable cofactor part is a single tiny factor, as on BLS12-381 $\mathbb{G}_2$ and BN254 $\mathbb{G}_2$ (Section 5.3).

Group	variant	R1CS	SCS
BW6-761 $\mathbb{G}_1$	hinted, unbound (unsound)	588 770	2 096 549
	classical GLV (deployed)	371 120	1 328 872
BW6-761 $\mathbb{G}_2$	hinted + binding (unsound)	895 498	3 135 285
	classical GLV (deployed)	371 120	1 328 872
BLS12-381 $\mathbb{G}_1$	hinted, unbound (unsound)	237 040	874 028
	hinted + binding	400 306	1 419 163
	classical GLV (deployed)	136 423	508 040

Table 2. Constraint counts for one emulated scalar multiplication over the BN254 scalar field. R1CS denotes the Groth16 arithmetization and SCS the PLONK arithmetization. "Deployed" marks the gadget currently shipped by `gnark`; unsound variants are included as comparison baselines.

6 Conclusion

The hinted-output scalar-multiplication methods are sound over prime-order groups but not over cofactor groups: even when you check the input point to the prime-subgroup, the certifying identity is read on the full curve, and nothing constrains the hinted output to the prime-order subgroup. We gave two forgery classes — an any-scalar attack that steers the decomposition into a torsion-cancelling sublattice, and a chosen-scalar attack that fixes the output-side coefficients and solves for the scalar — characterized the torsion orders each can reach as a function of the sub-scalar range bound, and showed the impact on production instantiations. The clean fix is a hinted preimage that binds the output into the subgroup, with a minimal clearing constant tuned to the threat model. More broadly, the episode illustrates a recurring pitfall in SNARK gadget design: a soundness argument that is valid under an unstated hypothesis (here, prime order), silently violated when the gadget is re-instantiated on a group that does not satisfy it.

Acknowledgements. A special case of this bug — the BLS12-381 $\mathbb{G}_1$ attack obtained by scaling the decomposition by 3 — was first discovered by the Kimi K3 model during an automated audit; Appendix A reproduces that finding. Prompted by this observation, we showed that it belongs to a much larger family of attacks, identified the distinct eigen and both-zero routes for realizing any-scalar forgeries, discovered the chosen-scalar family, exhaustively investigated

the affected deployed curves, and developed the fix analyzed here. We thank Gautam Botrel, Thomas Piellard, and Simon Masson for carefully reading earlier versions of this paper.

A The original Kimi K3 finding

The following reproduces the finding from the Kimi K3 audit report, with only notation and typesetting adapted to this paper and the evident divisibility typo $3 \nmid V$ corrected to $3 \mid V$.

BLS12-381 `ScalarMul` *admits a 3-torsion offset. Location:* `std/algebra/emulated/sw_emulated/point.go`, lines 1692–1736 and 1901–1906. The finding was confirmed by an R1CS-level proof of concept on eight out of eight random scalars.

The routine `scalarMulGLVAndFakeGLV` verifies the hinted output Q by computing sub-scalars $u_1, u_2, v_1, v_2 < 2^{66}$ satisfying

$$s(v_1 + \lambda v_2) \equiv -(u_1 + \lambda u_2) \pmod{r}, \qquad (v_1, v_2) \neq (0, 0),$$

and checking

$$[u_1]P + [u_2]\Phi(P) + [v_1]Q + [v_2]\Phi(Q) = 0_E.$$

Equivalently, on the prime-order subgroup this has the form $[V](Q - [s]P) = 0_E$, with $V \not\equiv 0 \pmod{r}$. The identity, however, is checked on the full curve, whose order is hr and whose BLS12-381 $\mathbb{G}_1$ cofactor is divisible by 3^2. Nonvanishing of V modulo r therefore does not rule out a nonzero cofactor-torsion component: when $3 \mid V$, a point of order 3 is annihilated.

The bound $2^{66} \approx 4.76\, r^{1/4}$ leaves enough headroom over an honest decomposition of size about $1.25\, r^{1/4}$ to multiply every coefficient by 3. Hence the prover can arrange $3 \mid V$ and return

$$Q' = [s]P + T_0, \qquad T_0 = (0, 2), \qquad [3]T_0 = 0_E, \qquad \Phi(T_0) = T_0,$$

while satisfying every constraint. The proof of concept forged the output hint as $[s]P + T_0$, and the circuit accepted it. The issue reached the EIP-2537 `BLS12_G1MSM` gadget because that gadget checked $P \in \mathbb{G}_1$, whereas the hinted output was bound only by the identity above.

The initial report suggested either tightening the sub-scalar bound below $2r^{1/4}$, which blocks this particular scaling-by-3 witness, or checking the hinted output in the prime-order subgroup.

The broader attacks developed in this paper show why blocking that one scaling witness is insufficient in general; Section 4 gives the resulting repair.

References

1. Barreto, P.S.L.M., Costello, C., Misoczki, R., Naehrig, M., Pereira, G.C.C.F., Zanon, G.: Subgroup security in pairing-based cryptography. In: Lauter, K.E., Rodríguez-Henríquez, F. (eds.) LATINCRYPT 2015. LNCS, vol. 9230, pp. 245–265. Springer, Cham (Aug 2015). https://doi.org/10.1007/978-3-319-22174-8_14
2. Botrel, G., Piellard, T., El Housni, Y., Kubjas, I., Tabaie, A.: gnark: a fast zk-SNARK library in Go. https://github.com/consensys/gnark (2020)
3. Bowe, S., Grigg, J., Hopwood, D.: Halo: Recursive proof composition without a trusted setup. Cryptology ePrint Archive, Report 2019/1021 (2019), https://eprint.iacr.org/2019/1021
4. Eagen, L., El Housni, Y., Masson, S., Piellard, T.: Fast elliptic curve scalar multiplications in SN(T)ARK circuits. In: Aly, A., Tibouchi, M. (eds.) LATINCRYPT 2023. LNCS, vol. 14168, pp. 109–129. Springer, Cham (Oct 2023). https://doi.org/10.1007/978-3-032-06754-8_4
5. El Housni, Y.: scalarmul-in-snark-attack: a proof of concept for the cofactor-torsion soundness attack on hinted scalar multiplication in gnark. https://github.com/yelhousni/scalarmul-in-snark-attack (2026)
6. Galbraith, S.D., Lin, X., Scott, M.: Endomorphisms for faster elliptic curve cryptography on a large class of curves. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 518–535. Springer, Berlin, Heidelberg (Apr 2009). https://doi.org/10.1007/978-3-642-01001-9_30
7. Gallant, R.P., Lambert, R.J., Vanstone, S.A.: Faster point multiplication on elliptic curves with efficient endomorphisms. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 190–200. Springer, Berlin, Heidelberg (Aug 2001). https://doi.org/10.1007/3-540-44647-8_11
8. Kosba, A., Zhao, Z., Miller, A., Qian, Y., Chan, H., Papamanthou, C., Pass, R., shelat, a., Shi, E.: C∅c∅: A framework for building composable zero-knowledge proofs. Cryptology ePrint Archive, Report 2015/1093 (2015), https://eprint.iacr.org/2015/1093
9. Lenstra, H.j., Lenstra, A., Lovász, L.: Factoring polynomials with rational coefficients. Mathematische Annalen **261**, 515–534 (1982), http://eudml.org/doc/182903
10. Lim, C.H., Lee, P.J.: A key recovery attack on discrete log-based schemes using a prime order subgroup. In: Kaliski, Jr., B.S. (ed.) CRYPTO'97. LNCS, vol. 1294, pp. 249–263. Springer, Berlin, Heidelberg (Aug 1997). https://doi.org/10.1007/BFb0052240