ECC RESEARCH PROGRAM · P5.3

Counting the Designer's Choices: A Game-Based Formalization of Curve-Generation Rigidity

Selection capacity, the min(1, 2^b ε) union bound, provenance non-identifiability, and class-uniform minimal-freedom generators at toy scale

math.st [1] **@math__street** [1]

with GPT 5.6 Sol · Claude Fable 5

[1]Independent research collective, math.st · July 2026

ABSTRACT

Every deployed standard curve was chosen by someone, and the recurring worry — made concrete by the BADA55 manipulation demonstrations — is that the chooser could have screened many candidate parameter sets against a weakness known only to them. We formalize this worry as an accounting problem. A precommitted **accounting contract** fixes a candidate universe Ω, a public safety predicate, and a finite menu of designer-screenable generator executions; the **selection capacity** $b = \log_2 M$ is the logarithm of the number of distinct screenable projected outputs. In the resulting selective-generation game — weak set committed after the contract but before the fresh random tape — we prove that the designer wins with probability at most $\min\bigl(1, 2^b \kappa\varepsilon\bigr)$, where ε bounds the safe weak mass under a reference distribution and κ is a domination factor; no independence between candidates is used, and the bound is attained by translated disjoint hit events. Four quantifier reorderings each void the bound. We then prove a **non-identifiability theorem**: the final published output — even with a seed and a deterministic derivation — cannot determine the historical b, and we give a sufficient provenance certificate that does. An audit of five deployed curves under an explicit profile yields conditional caps: P-256 at most 161 curve-core / 417 package bits, Curve25519 zero core bits given its field, brainpoolP256r1 zero core and one package bit, and $\perp$ (not identifiable from the cited record) for secp256k1 and BLS12-381 — a statement about records, never about intent. Finally we construct generators minimizing b: an ideal canonical-beacon generator with $b = 0$, and two toy instantiations at $p = 127$ validated exhaustively — a forced first-passing sampler uniform over the 4,179 safe coefficient encodings, and a class-uniform kernel over the 67 safe isomorphism classes that corrects the exact $132/13333$ total-variation bias of coefficient sampling and is minimax against single-class weaknesses. The formal results are proved; the sampling statements are conditional on an ideal XOF and an unbiasable beacon; all experiments are deliberately toy-scale.

Keywords. rigidity · standard curves · nothing up my sleeve · selection capacity · provenance · verifiable generation · public beacons

1 Introduction

Elliptic-curve standards ship constants. Some arrive with a derivation — a hashed seed, digits of π, a stated minimality rule — and some arrive bare. The community's informal term for the desirable property is **rigidity**: the published parameters should have left their designer as little room to steer the outcome as possible, because a designer who can screen many candidates can, if a weakness predicate is known to them alone, publish a curve that passes every public test yet lies in the weak set. This is not a hypothetical style of critique: the BADA55 project showed explicitly how "verifiably random" and "verifiably pseudorandom" generation procedures can be manipulated through unaccounted degrees of freedom in hash choice, seed choice, counter conventions, and acceptance criteria [1]. What the literature lacked — and what problem P5.3 demands — is the formal object: a game with explicit quantifiers, a designer-freedom measure b in bits, a proved bound of the shape $\min\bigl(1, 2^b\varepsilon\bigr)$, a reproducible accounting rule, an audit of real standards under it, and a generator that provably minimizes b.

This paper delivers that formalization and reports what it says about five deployed curves. Three findings organize everything. First, the requested bound is a **union bound**, and it is exactly as strong as its quantifier order: the weak set must be committed before the fresh randomness, the menu must be charged in advance, and the candidate marginals must be dominated by the reference distribution — drop any one hypothesis and no bound in terms of the reported b survives. Second, **rigidity of the record and rigidity of the history are different properties**: a published seed plus a deterministic derivation makes replay exact, yet we prove that no function of the final output alone can recover the historical menu size. Auditing therefore returns either a conditional cap or the honest verdict $\perp$ ("not identifiable from the cited source"), never a number conjured from a constant's bit length. Third, the minimal-freedom construction is achievable and cheap to state: an unbiasable public beacon plus forced uniform unranking of a canonically ordered safe set has designer capacity $b = 0$, and at toy scale we exhibit it exactly, down to the isomorphism-class level.

Main results. **(1) Bound.** PROVED In the fixed-menu selective-generation game with selection capacity $b = \log_2 M$, domination factor κ, and safe weak mass at most ε, the designer's success probability is at most $\min\bigl(1, 2^b\kappa\varepsilon\bigr)$; uniform candidate marginals give the requested $\min\bigl(1, 2^b\varepsilon\bigr)$, and the bound is tight. No candidate independence is assumed.

(2) Non-identifiability. PROVED For any published output, admissible histories realize every menu size $1 \le M \le |\Omega|$. Historical b is not a function of the final specification — a seed and deterministic derivation included — and becomes identifiable exactly when a provenance certificate pins the finite menu.

(3) Audit. CITED Under profile A256: P-256 curve-core $b \le 161$ and package $b \le 417$; Curve25519 core $b = 0$ given its field and package $b \le 1$; brainpoolP256r1 core $b = 0$, package $b = 1$; secp256k1 and BLS12-381: $\perp$ on both projections.

(4) Minimal generator. PROVED An ideal canonical-beacon generator attains the minimum $b = 0$. At $p = 127$ (toy scale, exhaustively verified) a forced first-passing sampler is uniform over the 4,179 safe coefficient encodings, and a class-uniform kernel over the 67 safe isomorphism classes is the unique minimax choice against single-class weaknesses, with total-variation gap $132/13333$ between the two projections.

1.1 Contributions and honest scope

We contribute (i) the accounting contract and the selection-capacity measure (§2); (ii) the selective-generation game, the $\min\left(1, 2^b \kappa\varepsilon\right)$ theorem with tightness, and four quantifier failures (§3, Figure 1); (iii) the final-output non-identifiability theorem, a seeded-replay corollary, and a sufficient provenance certificate (§4); (iv) a source-backed audit of P-256, Curve25519, brainpoolP256r1, secp256k1, and BLS12-381 under the explicit profile A256 (§5, Table 2, Figure 2); (v) minimal-freedom generators: the ideal canonical-beacon construction and two exhaustively validated toy kernels at $p = 127$ (§6–7, Figure 3, Figure 4); and (vi) design consequences and tradeoffs (§8).

The scope statement is part of the result, not a disclaimer. The audit is an accounting method: a large, small, or unidentifiable b measures selection capacity under a stated contract and carries **no claim about any designer's intent**. Public safety is a separate requirement from rigidity, and neither substitutes for the other. The generator experiments are deliberately toy-scale ($p = 127$, exhaustive point counting); the corresponding sampling statements are conditional on an ideal XOF and an unbiasable beacon, and we state the falsification conditions in §9. Nothing here is a production curve recommendation.

2 The accounting contract and selection capacity

The problem statement posits a generation procedure $\mathsf{Gen} : \{0,1\}^s \to \mathsf{Curves}$ with a public safety predicate Safe and asks for a bound on $\Pr[\mathsf{Gen}(\mathsf{seed}) \in \mathcal{B}]$ in terms of the designer's degrees of freedom b. The first formal task is to say what object b measures. Counting prose choices in a specification is not reproducible; counting encoded bit lengths of constants is wrong (§5.6). The unit that works is the **screenable candidate execution**.

Definition 1 (domain-parameter package). A **domain-parameter package** is a canonical encoding of every public object that may affect security: the base field, curve model and coefficients, subgroup order and cofactor, base point, and any representation data. The finite set of packages in the target profile is Ω, and the public safety predicate is $\mathsf{Safe} : \Omega \to \{0,1\}$.

Definition 2 (accounting contract). An **accounting contract** $\mathcal{A}$ fixes, before curve generation:

1. the target profile and canonical encoding of Ω;
2. Safe, including its version and all thresholds;
3. a finite index set I of designer-screenable candidate executions;
4. a joint random experiment $(X_i)_{i \in I}$ with each $X_i \in \Omega$;
5. a reference distribution ν on Ω; and
6. the projection used by the hidden weakness predicate — for example the curve core, or the entire domain-parameter package.

An incrementing counter inside one execution does **not** add an index when the contract forces publication of the first passing candidate. Skipping a passing candidate is a new screenable choice and must add an index.

Definition 3 (selection capacity). Let $M = |I|$ after identifying executions whose projected outputs are equal for every random tape. The **selection capacity** of $\mathcal{A}$ is

$$b(\mathcal{A}) = \log_2 M. \tag{1}$$

This is a support-size (max-entropy) measure, not Shannon entropy; it can be non-integral when M is not a power of two.

Definition 4 (category accounting). A category accounting is a decision tree whose branches are choices such as security profile, field, curve model, hash/XOF, seed source, encoding, counter and stopping rule, safety/cofactor policy, and base-point rule. The authoritative b is the logarithm of the number of distinct screenable leaves. If category j has at most m_j branches after every history, the convenient compositional cap is

$$b \le \sum_j \log_2 m_j, \tag{2}$$

with equality exactly when the branches form a full Cartesian product of distinct projected outputs.

Two design decisions in these definitions do real work. Charging by **projected outputs** (item 6 of the contract) makes b sensitive to what the weakness can depend on: a base-point choice is free capacity against a weakness of the curve core, but not against a weakness of the full package — the audit in §5 reports both projections separately for exactly this reason. And the counter rule separates **forced iteration** (a first-passing loop the designer cannot steer, which costs nothing) from **selection** (any ability to skip a passing candidate, which must be charged); this is the formal line between Brainpool-style seed increments and an "arbitrary reseed on failure" step.

3 The selective-generation game and the main bound

Definition 5 (game $\mathsf{RigSel}_{\mathcal{A},\varepsilon}$). The order of play is:

1. The auditor fixes $\mathcal{A}$.
2. A weakness source commits to $\mathcal{B} \subseteq \Omega$. The set may depend on $\mathcal{A}$, and the designer knows it, but it may not depend on the fresh random tape used in Step 3.
3. The challenger samples the joint experiment $(X_i)_{i \in I}$.
4. The designer sees or computes the candidates, tests membership in its hidden $\mathcal{B}$, and publishes one index i^*, or fails.
5. The designer wins exactly when $X_{i^*} \in \mathcal{B}$ and $\mathsf{Safe}(X_{i^*}) = 1$.

The accounting contract is the **meta-specification**. A designer may choose a residual final specification after learning $\mathcal{B}$, but every such choice must already be represented in I. This models a designer who knows a weakness before screening parameters, without allowing them to invent an uncharged generator tailored to that weakness. Figure 1 shows the order of play and where the capacity enters.

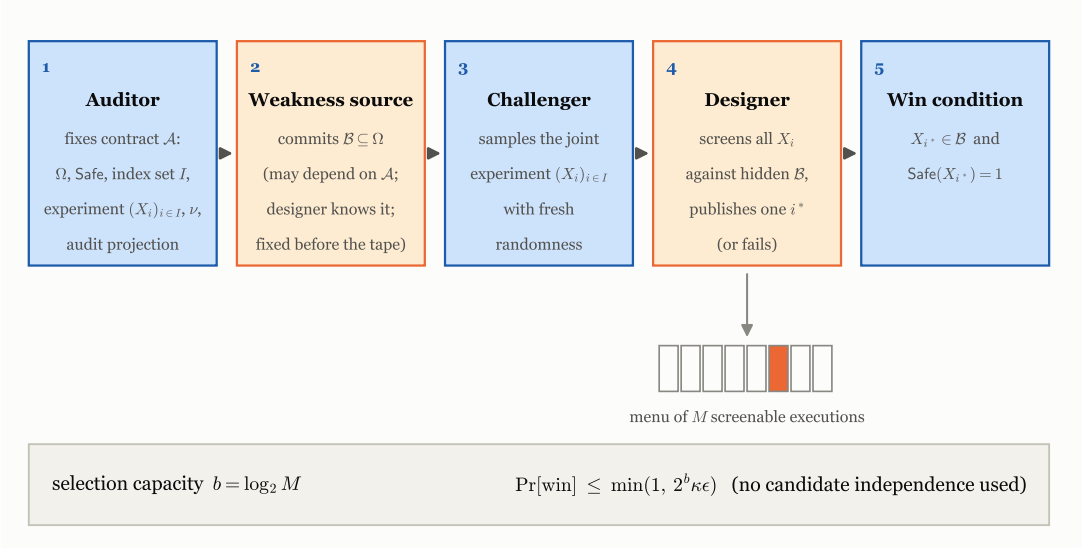

Figure 1: The selective-generation game $\mathsf{RigSel}_{\mathcal{A},\varepsilon}$ (Definition 5). The weak set is committed after the contract but before the fresh tape; the designer screens a precommitted menu of M executions and publishes one. The only quantity the bound charges is the menu size, through $b=\log_2 M$.

Definition 6 (source-bounded weak mass and domination). Write μ_i for the marginal distribution of X_i. The **source-bounded weak mass** is

$$\varepsilon_{\mathcal{A}}(\mathcal{B}) = \max_{i\in I} \mu_i(\mathcal{B}\cap \mathsf{Safe}^{-1}(1)). \tag{3}$$

When ν is uniform and all $\mu_i=\nu$, this is the usual safe weak-set density. More generally the **domination factor** is

$$\kappa = \max_{i\in I,\ x:\nu(x)>0} \frac{\mu_i(x)}{\nu(x)}, \tag{4}$$

with $\kappa=\infty$ if some μ_i charges a point outside the support of ν.

Theorem 7 (selective-generation union bound). PROVED If $\nu(\mathcal{B}\cap\mathsf{Safe}^{-1}(1))\le\varepsilon$ and $\mu_i(x)\le\kappa\nu(x)$ for every i,x, then

$$\Pr\left[\mathsf{RigSel}_{\mathcal{A},\varepsilon}=1\right] \le \min(1, M\kappa\varepsilon) = \min\left(1, 2^b\kappa\varepsilon\right). \tag{5}$$

In particular, for uniform candidate marginals $\kappa=1$ and the requested bound $\min\left(1,2^b\varepsilon\right)$ holds.

Proposition 8 (distribution-free form). PROVED With the source-bounded mass of Definition 7, the same argument gives

$$\Pr[\mathsf{RigSel}=1] \le \min\left(1, 2^b\varepsilon_{\mathcal{A}}(\mathcal{B})\right). \tag{6}$$

This is the correct statement when "fraction of curves" is not tied to a uniform or dominated sampling kernel.

Proof. For each i let $E_i=\{X_i\in\mathcal{B}\cap\mathsf{Safe}^{-1}(1)\}$. A win implies $\bigcup_i E_i$, **regardless of how the designer selects** i^* — this is where the adversary's whole strategy is absorbed. Domination gives

$$\Pr[E_i] = \mu_i(\mathcal{B} \cap \mathsf{Safe}^{-1}(1)) \le \kappa\nu(\mathcal{B} \cap \mathsf{Safe}^{-1}(1)) \le \kappa\varepsilon, \tag{7}$$

and the union bound gives $\Pr\left[\bigcup_i E_i\right] \le M\kappa\varepsilon$; every probability is at most one. No independence between candidates is used. For the distribution-free form, bound each $\Pr[E_i]$ directly by $\varepsilon_{\mathcal{A}}(\mathcal{B})$. □

Proposition 9 (tightness). PROVED Let $\Omega = \mathbb{Z}/N\mathbb{Z}$, let R be uniform, let $\mathcal{B}$ be a block of k elements, and set $X_i = R + t_i$ for translations whose inverse images of $\mathcal{B}$ are disjoint. When $Mk \le N$, every X_i is uniform and the win probability is **exactly** $Mk/N = M\varepsilon$. When the translated blocks cover Ω, the saturation value 1 is attained. For independent uniform candidates the exact probability is $1 - (1-\varepsilon)^M$, so the union bound is asymptotically tight when $M\varepsilon$ is small.

The tightness construction is deliberately **correlated** — all candidates are shifts of one uniform draw — which is why the theorem must not, and does not, assume independence. Real generator menus (several hash functions applied to one seed source, several counters of one stream) are correlated in exactly this way.

3.1 Quantifier failures

Each hypothesis of the game is necessary; reordering any quantifier destroys the bound. The four failure modes below are the formal content of the phrase "nothing up my sleeve" and mark the boundary between what the theorem promises and what folklore sometimes claims for it.

- **Post-output weakness.** PROVED If $\mathcal{B}$ may be chosen after a candidate is observed, it can be the singleton containing that candidate. The designer then wins with probability one even when $\varepsilon = 1/|\Omega|$ and $b = 0$. (Rigidity bounds are meaningless against weaknesses defined by the output itself.)
- **Uncharged meta-specification.** PROVED If the generator itself may be chosen after $\mathcal{B}$ without that choice appearing in I, a constant generator can target an element of $\mathcal{B}$; no bound in terms of the **reported** b follows. (This is the BADA55 scenario [1]: the manipulated freedom lives above the documented procedure.)
- **Biased source.** PROVED Cardinal density alone is insufficient when a candidate distribution concentrates on $\mathcal{B}$. The factor κ, or the direct marginal bound $\varepsilon_{\mathcal{A}}(\mathcal{B})$, is necessary.
- **No fresh experiment.** PROVED A deterministic fixed curve admits no nontrivial probability statement against a fixed $\mathcal{B}$: it is either weak or not. Reproducibility can make $b = 0$, but probabilistic assurance additionally needs randomness independent of $\mathcal{B}$, or a distributional model for $\mathcal{B}$.

4 Provenance is not identifiable from the final output

The audit question is historical: **how much capacity did the designer of a published curve actually have?** The clean negative result of this section is that the published record, restricted to the final output, cannot answer it.

Definition 10 (record map and identifiability). Let ℓ map a complete design history H to the surviving public record $O = \ell(H)$. Historical selection capacity is **identifiable from O** on a class of admissible histories when $b(H)$ is constant on every fiber $\ell^{-1}(O)$.

Theorem 11 (final-output non-identifiability). PROVED Fix a published projected output $x \in \Omega$. If the public record contains x but places no further restriction on the pre-publication history, then for every integer M with $1 \le M \le |\Omega|$ there is a history with that same record and selection capacity $\log_2 M$. Consequently, no function of the final output alone can recover the historical b on this class.

Proof. For $M = 1$, use the forced singleton menu $\{x\}$. For general M, choose $M - 1$ other distinct projected packages and make those packages together with x the designer-screenable menu; the designer publishes x. Both histories have the same surviving output, while their capacities are 0 and $\log_2 M$. Hence b is not constant on the observation fiber. □

Corollary 12 (seeded replay). PROVED A published seed and a deterministic seed-to-curve derivation do not by themselves identify provenance capacity. If the origin of the seed is absent, the same record is compatible both with an externally forced singleton seed and with designer screening over any finite collection of seeds containing it, up to the number of distinct projected outputs of the derivation.

This corollary is the formal reason "verifiably random" NIST-style seeds do not close the provenance question: the derivation constrains what follows the seed, not what preceded it. The positive counterpart says exactly which additional evidence closes the fiber.

Proposition 13 (certificate criterion). PROVED A record identifies b once it fixes a finite accounting contract and supplies evidence that every admissible history consistent with the record has the same quotient menu size. One sufficient certificate contains:

1. a digest and externally dated commitment to the meta-specification, including the safety predicate and audit projection;
2. the complete finite candidate domain and canonical equivalence relation;
3. the exact generator, dependency versions, enumeration order, tie breakers, and forced stopping rule;
4. the origin and transcript of any fresh randomness, with restart, suppression, and substitution rules;
5. a replay transcript or succinct proof covering rejected candidates and the published candidate; and
6. a statement of every residual designer-selectable branch.

These items determine the screenable index set I, its quotient by projected-output equality, and therefore $b = \log_2 |I|$. Omitting an item need not make identification impossible, but then uniqueness of the quotient menu size requires a separate proof.

Remark 14 (the meaning of $\perp$). PROVED A literal-only record can establish exact replay while leaving provenance non-identifiable. The symbol $\perp$ in the audit below means precisely that the cited record does not determine a common value of b across its admissible history fiber. It asserts neither zero nor infinity, and it carries no claim of malicious selection.

5 An audit of five deployed curves under profile A256

Definition 15 (audit profile A256). Profile A256 fixes the named approximately 128-bit security level, a prime subgroup of at most 256 bits, and the published generator document. It uses

two target projections: **core** (field plus curve equation, modulo only conversions explicitly fixed by the source) and **package** (the core plus subgroup, cofactor, base point, and representation parameters). Its category rules are:

1. a value obtained by a fully specified derivation and forced first-passing rule costs zero bits;
2. an unexplained L-bit seed receives the **full-seed sensitivity cap** $b \leq L$ — all 2^L inputs are conservatively treated as screenable; this is a cap, not a historical claim that every seed was tried;
3. an explicit random choice, a "choose one solution" step, or an unresolved finite solution set with m distinguishable outputs costs $\log_2 m$ bits;
4. a published literal with no finite source menu or exhaustive selection rule is marked $\perp$; its encoding length is **not** substituted for menu size;
5. a parameter designated as a profile input is reported conditionally (e.g. "given p"), with unresolved upstream freedom left visible as boundary debt; and
6. the curve-core count quotients out base-point-only choices; the package count does not.

The audited value is a selection-capacity **upper bound**; it is exact only when the cited source gives the complete menu and distinctness is established.

The audit uses published primary specifications only [2], [3], [4], [5], [6], [7], [8], [9], [10]. Table 1 is the category vector — the accounting that prevents apparently default choices (hash, encoding, endianness, counter rule, base point) from silently disappearing. Table 2 and Figure 2 summarize the results.

Table 1: A256 category vector (bits per category; "0" = fully forced by the cited source, "$\perp$" = no finite source menu). The profile/size category is zero for every row and omitted. "Reseed free" marks FIPS 186-4's return to an arbitrary new seed on failure; "stated, no domain" marks SEC 2 v1's repeated-selection sentence without a candidate domain or order.

Curve	field	equation	hash / enc.	seed	counter / stop	safety / cof.	base point
P-256	$\perp$ up; 0 given p	≤ 1 (b-roots)	0	≤ 160	forced; reseed free	0	≤ 256 pkg-only
Curve25519	$\perp$ up; 0 given p	0	0	0	0	0	≤ 1 sign; 0 u-only
brainpoolP256r1	0 (π)	0	0	0 (e)	0	0	1 pkg-only
secp256k1	$\perp$	$\perp$	0	none	stated, no domain	0	$\perp$
BLS12-381	$u : \perp$	$\perp$	0	none	incomplete	criteria only	0 given rule

Table 2: A256 audit results. Core and package are conditional selection-capacity caps under the source boundary in the first column; $\perp$ means "not identifiable from the cited public record" (Remark 14) — neither zero nor infinity, and no claim about motive.

Curve / source boundary	core b	package b	**Boundary debt**
NIST P-256 — FIPS 186-4, fixed p, $a = -3$, SHA-1 rule	≤ 161	≤ 417	selection of p, model, seed provenance
Curve25519 — RFC 7748, fixed p	0	≤ 1 affine; 0 u-only	exhaustive field-selection rule
brainpoolP256r1 — RFC 5639	0	1	target-size and policy choice only
secp256k1 — SEC 2 v1 and v2	$\perp$	$\perp$	finite menus/order for p, a, b, G
BLS12-381 — construction note + earliest surviving commit	$\perp$	$\perp$	finite menu/order for u; pre-publication transcript

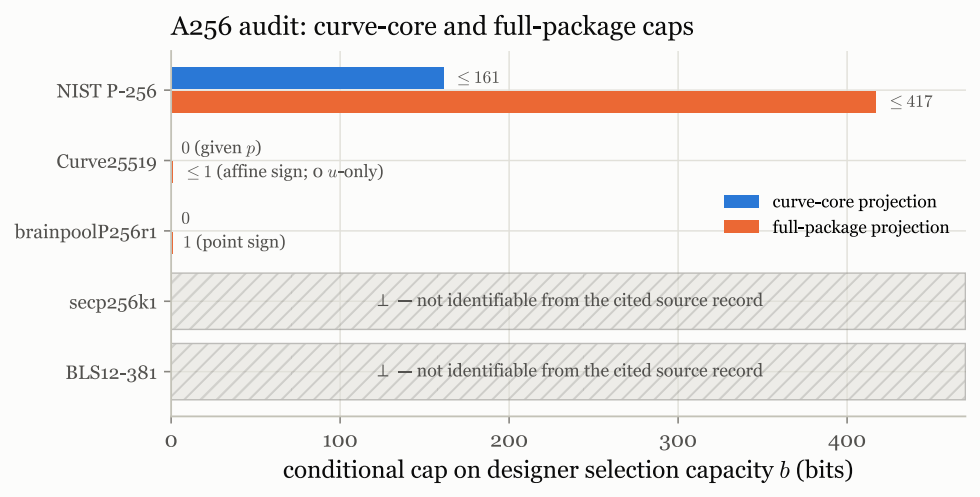

Figure 2: The audit of Table 2 as a chart. Numeric bars are conditional caps; hatched rows are $\perp$ verdicts, which are **statements about the surviving record**, not large or small numbers. The core/package split is the audit projection of Definition 2: base-point freedom exists only in the package column.

5.1 Source trace and judgement calls

NIST P-256. CITED FIPS 186-4 states that the recommended curves were generated using SHA-1 by the ANSI X9.62 / IEEE 1363-2000 method; it lists P-256's 160-bit seed, derived coefficient, order, and base point, describes the supplied base point as a **sample** while allowing any point of order n, and its prime-case procedure chooses an **arbitrary** 160-bit seed, expands it deterministically, returns to a fresh arbitrary seed when a candidate is unsuitable, and then chooses a, b satisfying $cb^2 \equiv a^3 \pmod{p}$ [2]. PROVED Judgement: the seed is charged under the full-seed sensitivity cap (≤ 160); fixing $a = -3$ leaves at most two coefficient roots, since every nonzero square relation $cb^2 \equiv -27 \pmod{p}$ has at most two roots, for one more bit. The core cap is therefore 161; adding fewer than 2^{256} nonidentity base-point choices gives the conservative package cap 417. This is not an exact joint count — seeds or roots may fail safety or collide — which is exactly why rule 2 calls it a cap.

Curve25519. CITED Given p, RFC 7748 selects the minimal positive Montgomery coefficient A with the required curve/twist cofactors and prime quotients, and the base point with minimal positive u; its base-point code takes a square root after fixing u, but the prose minimizes only u, not the sign of v; and the RFC states that the precise field prime depends on implementation concerns it does not fully articulate [3]. The original paper records six field-prime candidates chosen down to $2^{255} - 19$ by the smallest reduction constant, and the successive acceptable A values with the reasons the first two were rejected [4]. PROVED Judgement: zero curve-core capacity after p is fixed; at most one affine sign bit in the package, which the X25519 u-only package quotients out. The documented field rationale is valuable evidence but not an exhaustive precommitted universe of field shapes, so the **unconditional** provenance count is not inferred to be zero — the freedom is held at the boundary, visible in Table 2.

brainpoolP256r1. CITED RFC 5639 fixes big-endian conversion, SHA-1, seed increments, prime search, curve tests, and first-passing rules; its field seeds are consecutive 160-bit substrings derived from π and its curve seeds from e; and step 12 of the base-point construction explicitly chooses **at random** between Q and $-Q$ [5]. PROVED Judgement: zero core bits — forced increments add no capacity because the source does not permit skipping a passing candidate — and exactly one package bit for the documented sign choice.

secp256k1. CITED SEC 2 v1 states that the prime-field Koblitz parameters were obtained by **repeatedly selecting** parameters admitting an efficiently computable endomorphism until a prime-order curve was found, and publishes the full tuple without a candidate domain, enumeration or sampling distribution, rejected candidates, or base-point derivation [6]. SEC 2 v2 preserves the tuple ($a = 0$, $b = 7$), classifies the curve as Koblitz, and adds no seed, menu, or stopping rule [7]. PROVED Judgement: the v1 sentence establishes a broad stopping condition but leaves multiple histories with different menu sizes compatible with the published tuple. Fixed literals make document replay deterministic, but **replay is not provenance**; A256 reports $\perp$, not zero and not the sum of the literals' bit lengths.

BLS12-381. CITED The construction note gives $u = -$`0xd201000000010000`, the fields and curves, and goals including a 255-bit scalar field, a large power-of-two root of unity, efficient towers, and low Hamming weight; it defers a fuller account of selection to future work [8]. The earliest surviving commit of the linked repository (zero-parent, July 8, 2017) adds explicit bounds $q < 2^{383}$, $r < 2^{255}$, the residue condition $u \bmod 72 \in \{16, 64\}$, the claim that the chosen u gives the largest q and smallest Hamming weight meeting the requirements, and a canonical G1/G2 generator rule (lexicographically smallest valid coordinates, then cofactor scaling) [9]. The crates.io and docs.rs histories begin at `pairing` 0.9.0 on the same date; no earlier registry snapshot exists [10]. PROVED Judgement: the surviving repository strengthens replay and reduces **generator-only** residual freedom to zero once its conventions are fixed. It still does not state a finite domain for signed u, a numeric threshold for "large" 2^n, a priority between the two optimization objectives, or a rejected-candidate transcript — and its root commit postdates the March announcement. Both unconditional counts remain $\perp$.

5.2 Reproducibility versus provenance

Three facts justify the audit's refusal to output numbers it cannot defend. PROVED Treating every literal in a published document as forced gives a replay count of zero for almost any named curve, making the metric unable to distinguish a documented first-passing construction from an

unexplained constant; A256 avoids this collapse by reporting $\perp$ rather than silently assigning zero. PROVED Conversely, charging each literal by its encoded bit length is not a valid reconstruction of designer freedom: a 256-bit constant may be the forced output of a zero-choice algorithm or one item from a much larger cross-category search. PROVED And by Theorem 10 this is not merely missing arithmetic: singleton and multi-candidate histories produce the same record, so further computation on the tuple cannot select the historical b. A dated menu commitment is **additional evidence**, not a derivable quantity.

6 Minimal-designer-freedom generation

The audit's counterpart is constructive: what does a generator with **provably minimal** capacity look like?

Definition 16 (ideal canonical-beacon generator). Fix a security profile λ, a finite canonically ordered safe package set Ω_λ, and an exact uniform-unranking routine. After the accounting contract and $\mathcal{B}$ are fixed, the challenger supplies an infinite unbiased public bit stream R that the designer cannot predict, select, restart, or suppress. For $k = \lceil \log_2 |\Omega_\lambda| \rceil$, read successive k -bit blocks; publish the package with that index for the first block smaller than $|\Omega_\lambda|$, rejecting larger blocks by a forced rule.

Proposition 17 (minimality). PROVED The ideal canonical-beacon generator has designer selection capacity $b = 0$, outputs the uniform distribution on Ω_λ, and is minimal because $b = \log_2 M \geq 0$ for every nonempty menu. Theorem 8 therefore gives win probability at most ε.

Proof. The forced stopping rule exposes exactly one publishable candidate, so $M = 1$. Rejection of indices outside the final incomplete power-of-two range leaves every accepted index with the same probability. The lower bound $b \geq 0$ follows from $M \geq 1$. □

Remark 18 (concrete refinement). CONDITIONAL: ideal XOF; unbiasable public beacon A practical specification can replace R by domain-separated XOF blocks from a **future** public beacon, use exact rejection sampling for field elements, and force the first curve passing a versioned safety predicate and the first canonically encoded subgroup generator. The resulting reference distribution is the one induced by that fixed rejection sampler; it need not be cardinal-uniform over a differently defined curve universe. The condition would be falsified by a demonstrable beacon-biasing strategy or a distinguishable bias in the XOF-derived candidate stream.

Remark 19 (implementation limitation). PROVED Uniform unranking of the entire safe set is a definitional construction, not an efficient production algorithm. A deployable generator must replace it with a specified sampling kernel and then state weak-set density **relative to that kernel** — the subject of the next two sections.

6.1 Toy instantiation: forced first-passing at $p = 127$

The script `sample_rigid_curve.py` (SG-08) instantiates the refinement over the largest prime below 2^7 congruent to 3 mod 4, i.e. $p = 127$. Candidate coefficients (a, b) are derived from domain-separated SHAKE256 blocks by **exact rejection** (values above the largest multiple of p in the byte range are rejected, so no modular-reduction bias exists), and the first candidate passing a fixed public safety profile is published. The profile requires curve and twist prime-subgroup bit lengths at least 5, curve and twist cofactors at most 8, embedding degree at least 4, absolute Frobenius

discriminant at least 16, and excludes traces 0 and 1. The base point follows a deterministic shared-library rule. The script is deliberately limited to ≤ 16-bit fields because it uses exact exhaustive point counting; it is an auditable reference object, not a production generator.

Lemma 20 (exact-uniform field sampling). CONDITIONAL: ideal-XOF model Let L be the largest multiple of p below the byte-string range. Conditional on an XOF integer being below L, each residue mod p has exactly L/p preimages. Repeating with fresh domain-separated blocks therefore returns a uniform field element.

Lemma 21 (first-passing distribution). CONDITIONAL: ideal-XOF model Let $S \subseteq \mathbb{F}_p^2$ be the coefficient encodings passing the fixed safety predicate and $\alpha = |S|/p^2 > 0$. For any $x \in S$, the probability that x is the first passing candidate is

$$\sum_{j \geq 0} (1-\alpha)^j \cdot \frac{1}{p^2} = \frac{1}{p^2 \alpha} = \frac{1}{|S|}. \tag{8}$$

The declared reference distribution is therefore uniform over safe coefficient encodings — not over isomorphism classes.

Proposition 22 (toy selection capacity). PROVED If the public beacon is unique and cannot be selected, restarted, or suppressed by the designer, the forced first-passing rule has one publishable execution and hence $b = 0$. The batch sample index in the measurement runs exists for measurement only; allowing a designer to select among batch rows would add the corresponding menu capacity. The cryptographic sampling statement remains conditional on the ideal-XOF model.

EMPIRICAL: bits=7, p=127, 8 beacon labels The smoke run of 2026-06-25 (Table 3, Figure 3) found a passing curve for every one of eight beacon labels, with the largest selected counter equal to 13; runtime was below one second. Three regression tests independently recomputed the selected curve order by exhaustive (x, y) enumeration, verified every recorded safety field, verified the subgroup generator, verified that every earlier counter fails the profile, and confirmed deterministic replay.

Table 3: EMPIRICAL: bits=7, p=127, 8 beacon labels The eight published toy curves from `sample_rigid_curve_b7_n8_20260625.csv`: forced counter, coefficients, curve order, prime subgroup order r, cofactor h, trace, and raw Frobenius discriminant. Repeated orders across labels reflect the small toy universe (67 safe classes, §7), not a sampler defect.

label	counter	(a, b)	$\#E(\mathbb{F}_{127})$	r	h	trace t	$t^2 - 4p$
0	13	$(113, 115)$	133	19	7	-5	-483
1	0	$(115, 12)$	123	41	3	5	-483
2	11	$(80, 26)$	107	107	1	21	-67
3	1	$(92, 90)$	115	23	5	13	-339
4	2	$(46, 28)$	111	37	3	17	-219
5	1	$(110, 37)$	115	23	5	13	-339
6	0	$(61, 118)$	123	41	3	5	-483
7	11	$(2, 1)$	123	41	3	5	-483

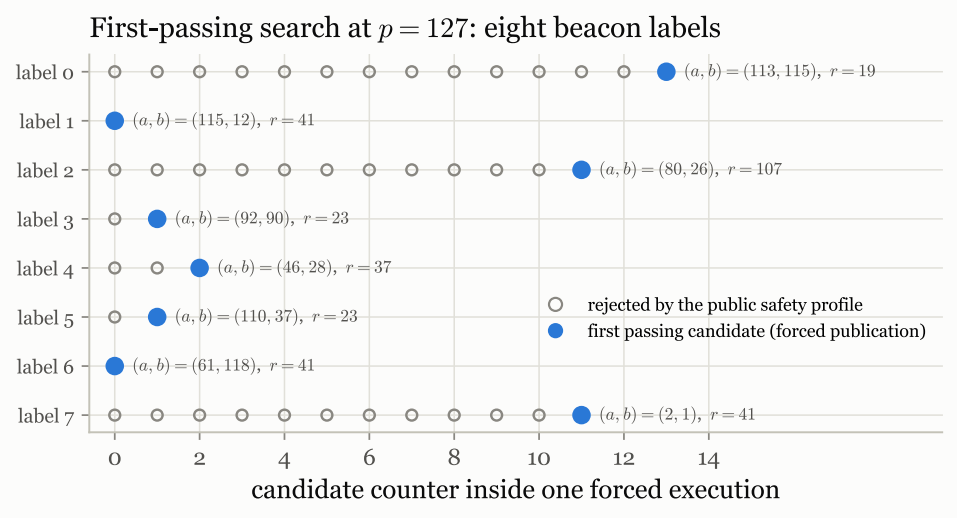

Figure 3: EMPIRICAL: bits=7, p=127, 8 beacon labels The forced first-passing searches behind Table 3. Gray circles are candidates rejected by the public safety profile (each rejection was re-verified by the regression suite); the blue marker is the forced publication. The designer steers none of it: the counter is not a menu.

7 Class-uniform refinement: the exact census at $p = 127$

The first-passing kernel is uniform over safe coefficient **encodings**. Whether that is the right reference distribution depends on the audit projection: a hidden weakness that respects isomorphism sees **classes**, not encodings, and uniformity does not survive the quotient when orbit sizes differ. At toy scale this can be settled exactly, and attempt A004 does so.

Lemma 23 (short-Weierstrass class criterion). PROVED Over a field of characteristic greater than three, two short-Weierstrass models $y^2 = x^3 + ax + b$ and $y^2 = x^3 + a'x + b'$ are isomorphic over the field, preserving the point at infinity, exactly when there is $u \neq 0$ with $(a', b') = (u^4 a, u^6 b)$.

Proof. Substituting the general Weierstrass change of variables into two short models forces its translation and shear terms to vanish because 2 and 3 are invertible. The remaining scaling $(x, y) = (u^2 x', u^3 y')$ gives the displayed coefficient action, and every such scaling is an isomorphism. □

Proposition 24 (kernel projection formula). PROVED Let $\mathcal{C}$ be the safe coefficient encodings, partitioned into scaling orbits O. Conditional on the coefficient kernel of §6.1 passing safety, its induced class probability is

$$\Pr[O] = |O|/|\mathcal{C}|. \tag{9}$$

Coefficient-uniform sampling is class-uniform exactly when all safe orbits have equal size. (The safety profile is constant on orbits, since it depends only on isomorphism-invariant curve and twist orders and derived quantities.)

EMPIRICAL: exhaustive p=127 census Exhaustive enumeration of all $0 \leq a, b < 127$ and all scalings $1 \leq u < 127$ gives: 16002 nonsingular encodings forming 258 isomorphism classes with orbit histogram $\{21 : 6,\ 63 : 252\}$; the fixed safety profile accepts 4179 encodings in 67 classes with safe

histogram $\{21 : 1,\ 63 : 66\}$. The single exceptional safe class is the $a = 0$ class with key $(0, 13)$, whose extra automorphisms shrink its orbit.

Proposition 25 (exact class masses and bias). PROVED Given the census counts, the exceptional safe class has coefficient-kernel mass $21/4179 = 1/199$ and every other safe class has mass $63/4179 = 3/199$, versus $1/67$ under class-uniform sampling. The total-variation distance is

$$d_{\mathrm{TV}} = \frac{1}{2}\left(\left|\frac{1}{199} - \frac{1}{67}\right| + 66 \cdot \left|\frac{3}{199} - \frac{1}{67}\right|\right) = \frac{132}{13333} \approx 0.0099. \tag{10}$$

Remark 26 (class-uniform kernel). CONDITIONAL: SHAKE256 blocks independent uniform The kernel `class_uniform_kernel.py` (SG-10) lexicographically orders the 67 canonical safe representatives and uses exact rejection to unrank one future-beacon-derived index, so it outputs every safe $\mathbb{F}_{127}$-isomorphism class with probability exactly $1/67$ (Figure 4).

Proposition 27 (toy capacity and minimax optimality). PROVED When the profile, class list, sample label, and future beacon are externally fixed and restart/suppression is forbidden, the class-uniform generator exposes one publishable execution and has $b = 0$. Moreover, among distributions with full support on a fixed set of N classes, the uniform distribution uniquely minimizes the largest singleton probability: from

$$1 = \sum_C \mu(C) \le N \max_C \mu(C) \tag{11}$$

we get $\max_C \mu(C) \ge 1/N$, with equality only when all masses are equal. Hence the toy class kernel is minimax against a hidden weakness containing one isomorphism class, under this fixed universe and projection, attaining $1/67$.

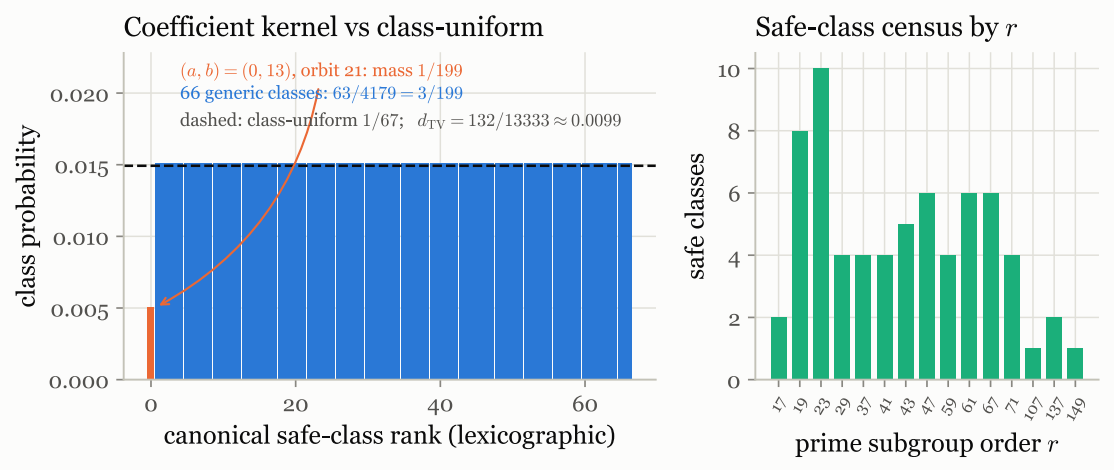

Figure 4: EMPIRICAL: exhaustive p=127 census Left: class mass of the coefficient-uniform kernel over the 67 canonical safe classes (from `class_kernel_b7_20260708.csv`), against the class-uniform value $1/67$. The single orbit-21 class $(0, 13)$ receives $1/199$; the 66 generic orbit-63 classes receive $3/199$; the exact total-variation gap is $132/13333$. Right: the census of safe classes by prime subgroup order r — the finite universe that the class-uniform kernel unranks.

The two kernels package the same lesson at different projections: "uniform" is not a property of a generator but of a generator **and** an equivalence convention, which is why the accounting

contract (Definition 2) carries the projection as explicit data. Fixing it in advance is what makes "the fraction of weak curves" a well-defined ε in Theorem 8.

8 Design consequences and comparison with the audit

Three consequences follow from the formalism and are worth stating as design rules.

- PROVED **A low b never compensates for a weak Safe.** Safety and designer selection capacity are separate coordinates: the bound controls only the probability of landing in a $\mathcal{B}$ that is rare among **safe** outputs.
- PROVED **Performance-motivated fields or models do not inherently add freedom** when the architecture, cost function, tie-breaking rule, and stopping rule are fixed before $\mathcal{B}$. An undocumented shortlist, by contrast, adds unmeasured freedom.
- PROVED **Algorithm agility is compatible with low b** only when the choice or migration rule is externally fixed. Allowing a designer to try several hashes, fields, encodings, or safety thresholds multiplies the screenable menu.

Against the audit, the constructions of §6–7 sit as follows. CITED RFC 5639 comes closest among the audited sources to an end-to-end forced construction — both seed families derive from named constants (π, e) with fixed increment rules — and its one explicit random point-sign choice is exactly its one package bit [5]. CITED RFC 7748 deterministically derives coefficient and base point but leaves the full field-selection considerations outside the derivation [3], [4]. CITED FIPS 186-4 makes the coefficient relation verifiable from a published seed but leaves the arbitrary seed and coefficient root outside any canonical provenance rule [2]. CITED The secp256k1 and BLS12-381 records do not publish finite selection menus sufficient for a source-only provenance number [6], [7], [8], [9], [10]. The ideal canonical-beacon generator dominates all five rows on the rigidity coordinate ($b = 0$ with fresh randomness independent of $\mathcal{B}$) — at the cost of requiring a beacon infrastructure and an explicitly enumerable (or exactly samplable) safe set, which is the real engineering tradeoff the audit surfaces.

9 Limitations and open questions

Toy scale. All positive sampling results run at $p = 127$ (with a $p = 31$ smoke profile) where the safe universe can be enumerated exhaustively. The $1/67$, $1/199$, $3/199$, and $132/13333$ values are exact for this universe and are claims about nothing larger. A production-scale profile would require replacing exhaustive unranking by a specified sampling kernel and proving its induced distribution — explicitly a new, separately authorized problem in the research plan.

Conditional hypotheses. The uniformity of both toy kernels is conditional on SHAKE256's domain-separated outputs behaving as independent uniform byte strings, and the $b = 0$ claims are conditional on a beacon the designer cannot select, restart, or suppress. A demonstrable beacon-biasing strategy or a distinguishable XOF bias falsifies the corresponding statements; the information-theoretic Proposition 16 is unaffected.

Caps, not exact counts. The numeric audit entries are conditional upper bounds under stated source boundaries (e.g. P-256's ≤ 161 assumes the field, model, and hash rule as given). Exactness would require the complete menu and distinctness of its projected outputs, which no audited source supplies.

The archival question (Q014). The $\perp$ verdicts are source-bounded: our archival search — both SEC 2 editions, the BLS12-381 announcement, every reachable commit of the linked repository, and the package registries — found stronger partial provenance than the announcements alone (a stopping criterion for secp256k1; bounds, a residue condition $u \bmod 72 \in \{16, 64\}$, and a canonical generator rule for BLS12-381) but no complete finite menu [6], [9], [10]. One Web Archive CDX request timed out — a failed retrieval channel, not evidence of absence. The question reopens only for a dated **pre-publication** artifact fixing the finite domain, selection order or objective priority, and transcript; later narrative alone cannot change the verdict, by Theorem 10.

What is not claimed. No statement here concerns any designer's motive; $\perp$ is a property of records. Rigidity does not replace curve, twist, subgroup, implementation, or protocol security analysis. And the fixed-menu game does not model weaknesses chosen after publication (§3.1): against $\mathcal{B}$ defined by the output itself, no generation procedure helps.

10 Conclusion

P5.3 asked for a definition, a bound, an accounting rule, an audit, and a construction, and each deliverable now exists in a form a referee can check. Rigidity is **selection capacity over a precommitted menu**: $b = \log_2 M$ distinct screenable projected outputs, charged under an explicit contract. The designer's advantage is bounded by $\min\bigl(1, 2^b \kappa\varepsilon\bigr)$ — a union bound, tight, independence-free, and exactly as strong as its quantifier order (Theorem 8). Historical b is not recoverable from a published curve, however deterministic its derivation looks (Theorem 10); it becomes recoverable precisely when a dated certificate pins the menu (Proposition 12). Under that discipline, the five-curve audit separates cleanly into conditional numeric caps (P-256, Curve25519, Brainpool) and honest non-identifiability (secp256k1, BLS12-381). The constructive answer — beacon plus forced canonical unranking — attains the minimum $b = 0$, and its toy instantiations at $p = 127$ show the full pipeline working end to end, including the subtlety the formalism was built to catch: uniform over encodings and uniform over isomorphism classes differ by an exact, computable 132/13333, and only the accounting contract's projection says which one "uniform" should mean.

Reproducibility

All experiments are deterministic and toy-scale. The forced first-passing sampler is `code/sample_rigid_curve.py` (restricted to $\leq$ 16-bit fields; validated against $\#E(\mathbb{F}_5) = 9$ for $y^2 = x^3 + x + 1$ and independent point enumeration); its smoke run produced `data/sample_rigid_curve_b7_n8_20260625.csv` in under one second. The class-uniform kernel and census generator is `code/class_uniform_kernel.py` (restricted to ≤ 8-bit fields); deterministic replay of its outputs yields SHA-256 `1BA019A7DA47C2FB64764B3D9A79680C7CB2904D6AD9062899069689AEB03F15` for `class_kernel_b7_20260708.json` and `EDC4C7875E2CE7A0AB0F44529BD65A99D35ADF5A57C8E485070B983DBFD382A9` for `class_kernel_b7_20260708.csv`. The combined shared and P5.3 test suite passed 79 tests under Python 3.13.4, including six dedicated kernel tests (exact counts and rational masses, every explicit scaling orbit, rank boundaries, subgroup points, deterministic domain separation, and the fixed $p = 31$ smoke CLI). Figures in this paper are generated from the checked-in CSV/JSON data by `figures/P5.3/make.py`. Every mathematical claim above carries one of the epistemic tags PROVED, CITED, CONDITIONAL: model, EMPIRICAL: range as used in the research log; untagged sentences are exposition, not claims.

References

[1] D. J. Bernstein *et al.*, "How to manipulate curve standards: a white paper for the black hat," in *Security Standardisation Research — SSR 2015*, in LNCS, vol. 9497. 2015, pp. 109–139.

[2] National Institute of Standards and Technology, "Digital Signature Standard (DSS)," technical report FIPS PUB 186–4, 2013. doi: 10.6028/NIST.FIPS.186-4.

[3] A. Langley, M. Hamburg, and S. Turner, "Elliptic curves for security." 2016. doi: 10.17487/RFC7748.

[4] D. J. Bernstein, "Curve25519: New Diffie–Hellman speed records," in *Public Key Cryptography — PKC 2006*, in LNCS, vol. 3958. 2006, pp. 207–228.

[5] M. Lochter and J. Merkle, "Elliptic curve cryptography (ECC) Brainpool standard curves and curve generation." 2010. doi: 10.17487/RFC5639.

[6] Standards for Efficient Cryptography Group, "SEC 2: Recommended elliptic curve domain parameters, version 1.0," technical report, 2000.

[7] Standards for Efficient Cryptography Group, "SEC 2: Recommended elliptic curve domain parameters, version 2.0," technical report, 2010.

[8] S. Bowe, "BLS12-381: New zk-SNARK elliptic curve construction." 2017.

[9] S. Bowe, "pairing library, zero-parent commit a06216f: initial BLS12-381 README and canonical generator rule." 2017.

[10] crates.io and docs.rs, "pairing crate version history, earliest version 0.9.0." 2017.