Classic McEliece is a safe long-term hedge. Famous last words?

Santosh Pandit, Independent Researcher
ORCID: 0009-0004-5258-8392
contact@santoshpandit.com

Preprint, Version 1, 8 September 2026
DOI: https://doi.org/10.5281/zenodo.22683800

Literature cutoff: 8 September 2026. This is a dated snapshot of a literature that produced seven significant results in four months.

Abstract

Three statements about Classic McEliece are routinely fused, and separating them is the purpose of this paper. Public-key indistinguishability for binary Goppa codes is false at deployed parameters. No stated security claim of the scheme has fallen. The sentence in our title has become a qualified claim rather than a safe one. What follows is a verified snapshot of the literature that produced that change.

Classic McEliece is described as the conservative choice in post-quantum key establishment. The construction dates to 1978, ISO standardised it in June 2026, and Germany's BSI recommends it for long-term confidentiality. Its distinguishing property is the length of its unbroken cryptanalytic record. That reputation rests on two separate propositions, which are often reported as one. The first is that syndrome decoding of a random linear code is hard, a claim with a six-decade attack history and a shallow improvement curve. The second is that the public key of a binary Goppa code is computationally indistinguishable from a random linear code. The second has no reduction to any better-understood problem, and has been known false at high code rates since 2011, though not at deployed rates until much later.

We review the structural cryptanalysis of the second proposition from 2011 to September 2026, verifying each load-bearing claim against the primary source. The position widely reported as current, that structural distinguishers reach only high-rate parameters outside the deployed range, ceased to be accurate in 2023 and was overtaken in August 2026. A distinguisher published that month applies to every Classic McEliece parameter set, at an estimated 2^{114} to 2^{124} bit operations against an information-set-decoding reference of 2^{151} to 2^{287}. That comparison counts bit operations only. The distinguisher also requires 2^{66} to 2^{72} bits of storage, which the reference figures do not price.

Five groups published key-recovery constructions against binary Goppa codes between June and September 2026. Three rest on independent algebraic methods, break published challenge instances in practice at claimed security levels up to 248 bits, and stop well short of Classic McEliece parameters. The estimates that reach the deployed sets all derive from the August distinguisher. An independent conditional tally of its direct route places all five parameter sets

below the NIST classical-gate reference for their claimed category, by 16 bits at Category 1 and roughly 135 bits at Category 5. A separate attempt to convert the same distinguisher into key recovery was refuted in print within weeks.

We report one feature of that cost profile that we have not seen stated elsewhere. The estimates are close to flat across parameter sets while the reference level climbs steeply. Moving from Category 1 to Category 5 therefore recovers about eleven bits of attacker work against a claimed gain of one hundred and twenty-nine, and the margin below the claimed level widens as parameters grow. Those figures are conditional estimates in cost models that exclude memory traffic, and they concern the shape of the published estimates. No attack has been implemented at these sizes, and no target key has been recovered.

We also report our own benchmark of both algorithm families on one host at N = 10,000. It locates the deployment cost of the hedge: nearly free for the sender, self-amortising in bandwidth after roughly a thousand sessions, and expensive only in receiver CPU at session establishment. There, one core sustains about ten decapsulations per second against ML-KEM-1024’s thirty-three thousand. The margin behind the word “conservative” is smaller than it was eighteen months ago, and the assumption is now contested, awaiting responses from the cryptographic community and from the institutions that recommend it.

---

1 Introduction

The case for Classic McEliece has never rested on performance. Public keys run from 261,120 to 1,357,824 bytes. Key generation is the slowest of any candidate NIST considered. NIST declined to standardise the scheme in March 2025 on grounds of key size and limited deployment interest, rather than on security.

The case rests on time. The construction dates to 1978, is roughly contemporary with RSA, and has absorbed six decades of attention from the coding-theory community without a structural break. Competing post-quantum constructions offer small keys and clean reductions. Classic McEliece offers something rarer: a great many people have tried to break it over a long period and have not succeeded.

That property carries the whole commercial and technical argument. It is why BSI recommends the scheme for long-term protection of confidential information. It is why deployed systems use it as the static-key half of hybrid constructions. It is why practitioners reach for it when data must stay confidential for decades rather than for a session. It is also why “**Classic McEliece is a safe long-term hedge**” has become close to a reflex in post-quantum discussion.

This paper asks whether that sentence still holds without qualification.

The answer is not the one we expected. Between 2023 and September 2026 the structural cryptanalysis of binary Goppa codes moved a long way. In 2022 no published structural attack reached the deployed parameter sets. By September 2026 a distinguisher applies to every deployed set. Its estimated bit-operation count falls below the generic-decoding reference, though it needs storage the reference does not. Five research groups have published key-recovery

constructions within four months. The most consequential result is three weeks old at the time of writing.

None of this breaks any security claim that Classic McEliece makes. We hold that distinction throughout. All of it bears on whether “conservative” can still be used without a footnote.

Our contribution is not the discovery of these results. Each is published and public. It is a consolidated and independently verified account of a fast-moving literature. To it we add three observations we have not seen stated together. The first is the inversion of the usual parameter-scaling defence. The second is the partial non-transferability of the defending team’s rebuttal structure. The third is that the 2026 key-recovery literature divides by technique rather than by paper count. Three independent methods stop short of Classic McEliece parameters, and every estimate that reaches them derives from a single technique three weeks old.

2 Background

2.1 The two assumptions

The Classic McEliece key-encapsulation mechanism, in the Niederreiter dual form, rests on two propositions. They are logically independent, and they are often reported as one.

The first is that syndrome decoding for a random linear code is hard. This is the well-founded proposition. General decoding is NP-hard, and the average-case problem has resisted effort since Prange’s algorithm of 1962. The best attacks form the information-set-decoding family, running from Prange through Stern, Dumer, May-Meurer-Thomae, Becker-Joux-May-Meurer, and May-Ozerov. The striking feature of that sequence is how little it moves. Six decades of work have improved the asymptotic exponent by a few percent, with no discontinuity. Quantum information-set decoding delivers roughly a square-root improvement and no more, from Bernstein’s application of Grover search in 2010 through the quantum-walk approaches of Kachigar and Tillich in 2017 and Kirshanova in 2018. Published parameter selection accounts for it.

The second is that the public key of a binary Goppa code is computationally indistinguishable from a random linear code of the same length and dimension. This proposition is weaker, in a way that is under-reported. It has no reduction to any better-understood problem. The security of the scheme is not proved to follow from it, and it is not proved to follow from anything else. It is an empirical claim about the difficulty of recognising algebraic structure, sustained by the observation that people have looked and not found.

2.2 What the scheme claims

The distinction matters because the second proposition is not the security claim.

The Classic McEliece submission grounds security on one-wayness under chosen-plaintext attack of the underlying decoding function. A tight Fujisaki-Okamoto-style transform with implicit rejection converts this to IND-CCA2 security. The submission team has consistently declined to claim public-key pseudorandomness as a theorem. They treat structural distinguishers as a minimum bar an attack must clear, rather than as an assumption the scheme depends on.

Their June 2026 response to a key-recovery claim states the position directly, writing of the targeted property that it is

> "a property that Classic McEliece explicitly avoids relying upon; not the McEliece one-wayness property that Classic McEliece explicitly relies upon; and not Classic McEliece's explicit security goal, namely IND-CCA2."

This is a documented design position held since the submission. A distinguisher, however cheap, does not by itself break a scheme that never claimed indistinguishability. Section 5.5 returns to how far this position extends to a key-recovery attack, which targets the property the scheme does rely on.

2.3 Why the key is large

The size of a Classic McEliece public key is the security argument in physical form, rather than an implementation inconvenience that a better design would have avoided.

Nearly every attempt to shrink a code-based public key has worked by adding algebraic structure. Nearly every such attempt has been broken by an attack exploiting the structure added. Generalised Reed-Solomon constructions fell to Sidelnikov and Shestakov in 1992. Reed-Muller variants, quasi-cyclic and quasi-dyadic compact-key variants, algebraic-geometry constructions, and wild Goppa codes over quadratic extensions have each been broken since, in most cases by distinguisher-then-filtration techniques that recover the hidden algebraic object. Classic McEliece declines the trade. It carries a megabyte-scale public key because it refuses the compactness that killed its relatives.

We believe the decision to keep a large key is what kept Classic McEliece alive while its compact relatives fell. It is not a defence against the 2026 results, which operate on the full-length code.

3 Related Work: the position as previously reported

The received account of Goppa distinguishing was accurate when written, and it continues to circulate.

Faugère, Gauthier-Umaña, Otmani, Perret and Tillich published a distinguisher for high-rate McEliece cryptosystems in 2011, as ePrint 2010/331, with a journal version in the IEEE Transactions on Information Theory in 2013. The distinguisher observes that the componentwise square of the dual of an alternant or Goppa code has a dimension defect relative to a random code of the same parameters. It detects that defect by linear algebra. It works when the dual dimension is small enough for the structured square to stay deficient, which means code rates close to one.

Classic McEliece does not operate at rates close to one. Its five parameter sets have rates between 0.729 and 0.797, with Goppa degree t between 64 and 128. The submission's own design rationale states the gap in its own terms. The distinguisher stops working at 8 errors for n = 1024, where the original 1978 parameters used 50, and at 20 errors for n = 8192, where the selected parameters use between 96 and 128. At the length that matters the deployed parameters therefore sit roughly five to six times beyond its reach. This is a difference in algebraic structure

rather than a narrow numeric margin. It is the basis for the standard statement that the only published structural result against binary Goppa codes comes nowhere near the parameters anyone deploys.

The same document records what the team judged the worst case to be:

> "The worst-case possibility is that this distinguisher somehow allows an inversion attack faster than attacks for random codes."

The 2026 literature is an attempt to realise that possibility. The rationale also considers choosing parameters to minimise ciphertext size. That would mean smaller t, "possibly in the range where [the distinguisher] can distinguish public keys from uniform random matrices", and the option is declined on the ground that it would complicate the security analysis. Staying clear of the distinguishable regime was a deliberate parameter-selection criterion rather than an accident of the design.

The 2026 survey by Mojaveri and Khosravi, independent of the submission team, states the accompanying caution well. **Breaking an obsolete parameter set is not the same as recovering the hidden Goppa structure, and distinguishing a public code does not necessarily lead to practical key recovery.**

Every element of that account is correct. What our review adds is not the discovery that it has been overtaken, since each result below is published. It is a consolidated statement of how far it has been overtaken, and a verification pass establishing which of the circulating characterisations of the newer work are accurate.

4 Scope and method

This review comes out of our own practice. We implement cryptographic agility and international algorithm interoperability at BeatQuantum, which means holding a working view of which algorithms are safe to carry, for which purpose, and for how long. We also track BSI's recommendations, which have carried Classic McEliece independently of the NIST outcome. Classic McEliece occupies a specific position in that architecture, which we set out separately in our work on the Hierarchy of Cryptographic Needs. It is the non-lattice key-establishment option, selected on the strength of its cryptanalytic record rather than its performance. A review of that record is therefore an operational requirement before it is an academic exercise.

A literature review carries a different methodological risk from a measurement study. The risk here is inaccurate secondary characterisation of primary results, and we treated it as the central problem.

We assembled candidate results from a broad initial sweep, then discarded the sweep and worked only from primary sources, retrieving each paper from the IACR Cryptology ePrint Archive or the publisher of record. For ePrint 2026/1630 we extracted the concrete cost tables from the paper itself rather than from any reported summary, and we quote its own figures. We read the defending team's most recent response document in full. For institutional positions we read the source documents, retrieving BSI TR-02102-1 version 2026-01 and TR-02102-2 version 2026-01 directly.

Claims we could not verify against a primary source are marked as unverified and excluded from the argument. Section 8 lists the three excluded claims.

Section 6 departs from this and reports our own measurements. The deployment cost of Classic McEliece is widely quoted as a set of ratios, and we wanted figures we had produced rather than figures we had read. That section states its own method.

The protocol earned its keep. During the review we found three cases where circulating secondary accounts misstated a primary source in ways that changed its meaning. Two reported the defending team’s rebuttals of weak claims as concessions of catastrophic weakness. One reported an attack cost against a lattice scheme as an attack cost against Classic McEliece. In this literature, at this speed, secondary characterisation is not a safe basis for a security judgement.

5 The escalation, 2023 to 2026

We separate distinguishers from key recovery. The two carry different weight and are routinely conflated.

5.1 Distinguishers reach the deployed parameters

Three results moved the distinguisher line from the high-rate regime into the deployed one.

Couvreur, Mora and Tillich, at Asiacrypt 2023, replaced the square-dimension observable with a matrix space of quadratic relations, attacked by MinRank and Gröbner-basis methods. Their distinguisher works, in their own words, “for code rates above 2/3”, and so covers every Classic McEliece parameter set. On concrete cost they are equally direct. For the NIST submission parameters the complexity “is way too complex to threaten the cryptosystem, but is smaller than known key recovery attacks for most of the parameters of the submission”. Both halves of that sentence deserve attention. The first has not been disputed since. The second records that a structural method had already undercut known key recovery in 2023, three years before the results of Section 5.2.

Randriambololona’s syzygy distinguisher, ePrint 2024/1193 and Eurocrypt 2025, changed the invariant again, computing graded Betti numbers of a shortening of the dual code. The 2023 result had already reached the Classic McEliece parameter sets, so the advance here is cost rather than reach. The authors state a complexity “subexponential in the error-correcting capability, hence better than that of generic decoding algorithms”. They describe the work as the first analysis in the chosen-plaintext model, since the system’s introduction in 1978, to break the exponential barrier. Its concrete cost at the smallest parameter set was reported at approximately 2^{529} operations, a figure the submission team addressed in a note dated 17 April 2025. The result depends on a heuristic prediction for the Betti numbers of random codes.

Hemmert and Wiemers, ePrint 2025/1661 and CRYPTO 2026, introduced a higher-order vanishing distinguisher for alternant and Goppa codes, including the Classic McEliece parameters, at a better concrete cost. The independently reported leading-term figure for the smallest parameter set is approximately 2^{298}. Both authors are employed by the German Federal Office for Information Security, which bears on Section 7.3.

At the end of 2025 the position was that structural distinguishers reached the deployed parameter sets, but at costs between 2^{298} and 2^{529}. Those figures sit far above generic decoding and far above exhaustive search over the 256-bit seed space. This weakened the 2011 position without being alarming, and the submission team's response, that a distinguisher which does not decode is not a break, was proportionate.

5.2 Key recovery arrives

Five groups published key-recovery constructions against binary Goppa codes between June and September 2026. The distinguisher-focused framing of the earlier literature does not prepare a reader for this, so we treat it separately. Four are described here. The fifth accompanies the August 2026 distinguisher and is treated with it in Section 5.3.

Briaud, Lemoine, Randriambololona and Tillich, ePrint 2026/1232, dated 10 June 2026, locate rank-2 matrices in the code of quadratic relations of a binary Goppa code and use them to recover an equivalent secret key. They report practical key recovery against published McEliece challenge instances issued by the Technology Innovation Institute, at claimed security levels of 83, 89, 119, 166, 210 and 248 bits. They also recover a CFS key with parameters r = 9 and m = 16 at 74.9-bit security, in fourteen hours using 24 GB of memory. They conjecture subexponential complexity. They state that the attack does not break Classic McEliece parameters. The author affiliations include ANSSI, the French national cybersecurity agency.

Hemmert, ePrint 2026/1339, dated 2 July 2026, achieves key recovery for binary Goppa codes across a wide parameter range using the higher-order vanishing machinery, breaking previously unbroken challenge instances. The author conjectures that the approach also applies to Classic McEliece parameter sets, while stating that the conjectured complexity remains significantly higher than the targeted security level. The author is employed by BSI.

Saarinen, ePrint 2026/1778, dated 22 August 2026, recovers keys for five Technology Innovation Institute challenges including TII-252. The analysis does not indicate a threat to Classic McEliece parameters in its present form.

Vedenev, ePrint 2026/1747, last revised 6 September 2026, derives two key-recovery attacks from the August 2026 distinguisher described below. For binary Goppa codes the author reports that derivative flag recovery remains out of computational reach. For Classic McEliece parameters the author states that the minor-code variant costs essentially several runs of the distinguisher, placing it well below the claimed security levels. The attacks are verified end-to-end over F_4 rather than over binary Goppa codes.

That claim did not stand for long. Apon, ePrint 2026/1810, dated 26 August 2026, gives a targeted refutation. Vedenev's held-position count assumes that the resulting linear equations are independent across positions. Apon reports finite-field experiments on binary Goppa instances that contradict the assumption, showing a waterfall in which additional held positions abruptly yield a single independent equation rather than many. He then proves an algebraic-geometric obstruction. For a binary Goppa polynomial of degree t, an explicit construction gives a 2t + 3 dimensional family modulo the true solution, so the number of held positions required satisfies c >= 2t + 3. At mceliece8192128 that is c >= 259, which places Vedenev's algorithm "well beyond 2^{1500} bit-operations" at Category 5.

Two features of the refutation matter as much as the result. Apon's experiments run on proper binary Goppa instances, which is the object Vedenev verified over F_4 rather than over binary Goppa. And his scope is carefully bounded. He separates the GIJS distinguisher, which he calls "seemingly valid", from Vedenev's reconstruction step, and states that he does not claim every possible use of the hold-out kernels fails to yield a key-recovery algorithm. The obstruction is conditional on an explicit nondegeneracy condition, which he states rather than folds into a generic-rank assumption.

Counting papers is the wrong way to read this. The right division is by technique, and it separates cleanly.

Three of the four rest on independent algebraic methods, and all three stop short of Classic McEliece. Briaud and colleagues state that the attack does not break its parameters. Hemmert places the conjectured complexity significantly above the targeted security level. Saarinen states that HOVER does not indicate a threat to them in its present form. These are the papers that break instances in practice, and they break instances well below Classic McEliece sizes.

The fourth, Vedenev, is not an independent method. It derives from the August 2026 distinguisher of Section 5.3, and it is that single technique, rather than the key-recovery literature as a whole, which produces estimates reaching Classic McEliece parameters. Its particular claim has since been refuted, though the technique it builds on has not.

The concentration matters for how the risk should be read. Three mature and independently developed attack families halt above the target levels and say so. The estimates that reach the deployed sets come from one technique a few weeks old, and the first published attempt to turn that technique into key recovery was answered within days by a lower bound. A reader who aggregates these papers into a single trend will overstate the breadth of the threat and understate both its novelty and its instability.

5.3 The August 2026 distinguisher

Ghoshal, Ishai, Jain and Sun published ePrint 2026/1630, "Quasipolynomial Cryptanalysis of the McEliece Cryptosystem (or: PIR Meets McEliece)", received 7 August 2026 and revised 27 August 2026. The authors are at IIT Madras, Technion and AWS, and Carnegie Mellon University. By their own account the construction originates in a failed attempt to build doubly efficient private information retrieval from algebraic locally decodable codes.

The algorithm is simple to state. Choose a parameter s of order log n. Hold out one column of the public matrix. By linear algebra, compute the space of multilinear homogeneous degree-(s+1) polynomials that vanish on the union of Hamming balls of radius less than s around every other column. Sample a random nonzero polynomial from that space and evaluate it at the held-out column. For a Goppa-McEliece public key the vanishing constraints force the evaluation to zero. For a uniformly random matrix it is nonzero with constant probability. The result runs in time $n^{O(\log n)}$ and distinguishes with advantage 1 - o(1) on all Classic McEliece parameter sets NIST considered.

The concrete estimates below are the paper's own, in the binary-operations metric, reproduced from its Table 1. T_{new} is the new distinguisher, T_{ISD} the information-set-decoding reference for

the same parameter set, and T_{HW} the leading-term cost of the previous best structural distinguisher.

Parameter set	m	n	t	k	$\log_2 T_{new}$	$\log_2 T_{ISD}$	$\log_2 T_{HW}$
mceliece348864	12	3488	64	2720	114	151	298
mceliece460896	13	4608	96	3360	120	191	595
mceliece6688128	13	6688	128	5024	124	257	691
mceliece6960119	13	6960	119	5413	123	257	565
mceliece8192128	13	8192	128	6528	124	287	551

Storage runs to 2^{66} and 2^{72} bits, on the order of eight to five hundred exabytes. That is the main reason the distinguisher is not executable. The paper's caption notes that optimisations in its Remark 5.1 are not reflected in the table and would cut the T_{new} figures by about twenty bits.

The T_{HW} column is not monotone in parameter size, and the reproduction above is faithful rather than mistaken. The prior distinguisher is costed at 2^{691} for mceliece6688128 but 2^{551} for the larger mceliece8192128, and at 2^{565} for mceliece6960119. We verified the alignment of every row against the source. The pattern is consistent with a distinguisher whose cost depends on the relation between length, dimension and Goppa degree rather than on parameter size alone. Note that mceliece6960119 carries a smaller t, and that mceliece8192128 offers more evaluation points at the same t and the same redundancy.

Table 3 of the same paper gives baseline and explicitly unoptimised estimates for a heuristic key-recovery attack: $2^{130.38}$ for mceliece348864, $2^{148.95}$ for mceliece460896, $2^{141.28}$ for mceliece6688128, $2^{140.45}$ for mceliece6960119, and $2^{141.60}$ for mceliece8192128.

Three qualifications apply before any inference. The distinguisher is presented as provable, while the key-recovery and decryption attacks are characterised as heuristic. The authors describe the decryption attack as not concretely efficient. They note separately that the key-recovery attack is much closer to the distinguisher, and may be relevant to NIST security levels. That distinction is theirs rather than ours. The cost model is new and the community has not audited it. The submission team maintains a systematically computer-checked framework, CryptAttackTester, for this purpose, and argues on its public comparison page that verifiability of attack cost estimates is a substantive property many published estimates lack.

One further point bears on how the result is received rather than on whether it is correct. The distinguishing core uses Hasse derivatives, Hermite interpolation, and Reed-Muller minimum-distance bounds. It does not depend on the heuristic predictions about random-code invariants that the syzygy and higher-order-vanishing distinguishers require. On what the distinguisher assumes, this result stands on firmer ground than its predecessors.

The first independent cost analysis appeared within three weeks, and it is the most decision-relevant document in this literature. Saarinen published ePrint 2026/1786 on 24 August 2026, last revised 7 September 2026, an itemised conditional tally of the direct holdout key-recovery route for each Classic McEliece parameter set. It compares against the NIST classical-gate reference level for each claimed category. That is the right comparison, since the reference is

what the parameter set is claimed to achieve rather than a modelled cost for one competing attack.

Parameter set	Category and NIST reference	Work	State	Margin below reference
mceliece348864	1 / 2^{143}	$2^{126.77}$	$2^{51.33}$	16.23 bits
mceliece460896	3 / 2^{207}	$2^{145.22}$	$2^{59.10}$	61.78 bits
mceliece6688128	5 / 2^{272}	$2^{137.48}$	$2^{55.18}$	134.52 bits
mceliece6960119	5 / 2^{272}	$2^{136.65}$	$2^{54.86}$	135.35 bits
mceliece8192128	5 / 2^{272}	$2^{137.48}$	$2^{55.18}$	134.52 bits

The author's summary is that "all five arithmetic estimates lie below the NIST classical-gate reference level for their claimed category". The identical figures for mceliece6688128 and mceliece8192128 are not a transcription artefact. The two sets share the same derived attack parameters in this route, since both carry t = 128, m = 13 and n - k = 1664. The tally does not otherwise depend on length.

The caveats are the author's own and they are extensive. The estimates are conditional on exact finite-instance structural conditions, and the paper does not prove that the required gates, public rank inequalities, or the coordinatewise selector succeed on the Classic McEliece cells displayed. Nonselectivity has been observed on small generated instances outside those cells. The model "excludes address generation and memory traffic and retains explicit budget allowances for some stages", which concedes directly the objection that a bit-operation count is not a full cost. The figures are "arithmetic and state estimates, not elapsed-time claims". And, stated without qualification, "no Classic McEliece target key has been recovered".

The refutation described in Section 5.2 does not reach this tally. It is directed at Vedenev's reconstruction step, and its author explicitly declines to claim that every use of the hold-out kernels fails. The direct-route figures therefore stand at the time of writing, unrefuted and unaudited.

Two things nevertheless follow. The state requirement of 2^{51} to 2^{59} bits is four to five orders of magnitude below the storage the distinguisher needs. The memory objection that weighs heavily against the distinguisher therefore weighs much less against this route. And the margin below the reference level grows with parameter size, from 16 bits at Category 1 to roughly 135 bits at Category 5.

5.4 The scaling inversion

The most consequential feature of the August 2026 cost profile has not, so far as we have seen, been stated in the secondary discussion of that paper.

What matters most is not the absolute cost of any of these attacks. It is how that cost behaves as the parameter set grows.

Read down the tally in Section 5.3. The NIST reference level rises from 2^{143} at Category 1 to 2^{272} at Category 5, a gain of 129 bits. That gain is what a user buys by moving to a larger parameter

set. The tallied work rises from $2^{126.77}$ to $2^{137.48}$ over the same range, a gain of about 11 bits. The distinguisher figures behave the same way, rising from 2^{114} to 2^{124} while the generic-decoding reference climbs from 2^{151} to 2^{287}.

The margin below the claimed level therefore widens as parameters grow, rather than closing: 16 bits at mceliece348864, 62 at mceliece460896, and 134 to 135 across the three Category 5 sets.

Moving from Category 1 to Category 5	Change
Claimed NIST classical-gate reference	2^{143} to 2^{272}, a gain of 129 bits
Tallied attack work	$2^{126.77}$ to $2^{137.48}$, a gain of 11 bits
Margin below the claimed level	widens from 16 bits to 135 bits

This inverts the conventional response to cryptanalytic pressure. When an attack erodes the margin of a parameter set, the standard remedy is to move to a larger one, and against generic decoding that remedy works well. Against these estimates it is close to counterproductive. Moving from Category 1 to Category 5 raises the attacker's tallied work by about 11 bits while raising the claimed level by 129. The larger sets therefore sit further below their claimed level rather than nearer it. If the estimates hold, no Classic McEliece parameter set restores a comfortable margin, and the usual mitigation is unavailable. In absolute terms the cheapest set to attack remains mceliece348864 at $2^{126.77}$, which is the set NIST rated contentious and ISO declined to standardise.

This is what quasipolynomial rather than exponential dependence means once expressed in the deployed range. It differs in kind from an exponential attack with a better constant.

Two limits on this observation should be stated where it is made rather than deferred.

It describes the shape of a set of published estimates rather than a demonstrated attack. Every figure in it is conditional, none of the underlying attacks has been implemented at these sizes, and no key at any Classic McEliece parameter set has been recovered.

It also rests on cost models that do not price memory. The distinguisher figures assume 2^{66} to 2^{72} bits of storage the generic-decoding reference does not need, and Saarinen states that his tally "excludes address generation and memory traffic". The submission team argues, we think correctly, that memory access is a real cost many published estimates omit, and maintains a computer-checked framework for attack costing partly for that reason. A comparison that prices bit operations and ignores storage is not a complete security comparison. The shape of the curve survives that objection, because the flatness holds within each cost model and the memory requirement is itself roughly flat across parameter sets. The inference that any of these attacks is cheaper in practice than generic decoding does not survive it, and we do not make that inference.

5.5 The defender's position

The submission team's response record is detailed, and on the evidence of three previous episodes it is effective. Their published notes address the syzygy distinguisher on 16 August 2024, and a claimed 2^{529} distinguisher cost on 17 April 2025. A note of 20 November 2025 addresses a paper on decapsulation, and one of 23 June 2026 a claimed 2^{610} key-recovery attack.

We read the June 2026 note in full, because it is the most recent and shows the structure of the defence. The argument has two prongs.

The first is cost. The team grants the paper's conjectures for the sake of argument and dismisses the result on the number. The attacks have cost far above 2^{256}. The largest reported demonstration attacks a toy size of (n, t) = (482, 7), which 2008 information-set-decoding software breaks in a fraction of a second on one core. The paper's asymptotic framing addresses "a strawman parameterization", because Classic McEliece uses size parameters and "explicitly overrides asymptotics with concrete analysis".

The second is the property argument quoted in Section 2.2. The targeted distinguishing property is one the scheme avoids relying on.

Both prongs hold against the paper they address. They transfer only partly to the August 2026 result.

The cost prong is unavailable. Estimates of 2^{114} for distinguishing and 2^{130} for key recovery are not far above 2^{256}. They fall below 2^{256} and below the generic-decoding reference for the same parameter sets. Whatever the right response to those figures is, it cannot be the June 2026 response.

The property prong reaches the distinguisher and stops there. A key-recovery attack targets the one-wayness property the scheme states it relies on, and the observation that indistinguishability was never claimed does not address it.

What remains is narrower than what has sufficed on every previous occasion: the key-recovery component is heuristic, it is unimplemented, and its cost model has not been independently audited. That is a substantial and legitimate defence, and a different one from the defence used before. It should not be reported as though the previous rebuttals had already disposed of this case.

The submission team has not published a response at the time of writing. The relevant page carries version 2026.06.23, which precedes the August result, and the most recent posting listed is dated 23 June 2026. The team has answered every prior claim in detail, and a careful cost-model analysis takes time. An interval of one month across a northern-hemisphere summer is weak evidence of anything, and we record the absence as a fact rather than as a signal.

6 The cost of the hedge

Structural security is one half of a deployment decision. The other is what the mechanism costs to run, and for Classic McEliece that cost is unusual enough to determine which architectures can carry it at all. We measured it rather than citing it.

6.1 Method

Both algorithm families were benchmarked on a single host using Botan 3.13.0, with 10,000 iterations of each operation, on 8 September 2026. The host is an AMD Ryzen 9 7950X with four vCPUs allocated, one thread per core, and 8 GB of memory. The benchmark is single-threaded. We report medians and 95th percentiles. The same harness and iteration count were run on 20 August 2026, which gives a reproducibility check rather than a single snapshot.

Measurements were taken on a live production server rather than an idle dedicated machine. Section 8 returns to what that costs in confidence.

Public key sizes are reported raw. The harness measures the DER SubjectPublicKeyInfo encoding, which adds a constant 22 bytes for ML-KEM and 26 bytes for Classic McEliece. Once that wrapper is subtracted, every measured size matched the specification, which is our check that the harness is sound.

The two runs agree closely. Across all fifteen measurements the medians differ by less than 6 percent, and eleven of the fifteen by less than 2 percent. The largest deviations fall on encapsulation, the only operation with a sub-millisecond median, where timer granularity and background load matter most. The September run has tighter 95th percentiles throughout, consistent with a quieter host.

6.2 Results

Scheme	Category	Public key	Ciphertext	Shared secret
ML-KEM-768	3	1,184 B	1,088 B	32 B
ML-KEM-1024	5	1,568 B	1,568 B	32 B
mceliece460896f	3	524,160 B	156 B	32 B
mceliece6688128f	5	1,044,992 B	208 B	32 B
mceliece8192128f	5	1,357,824 B	208 B	32 B

Timings in milliseconds, median with 95th percentile in parentheses, N = 10,000.

Scheme	Key generation	Encapsulation	Decapsulation
ML-KEM-768	0.061 (0.072)	0.020 (0.023)	0.023 (0.024)
ML-KEM-1024	0.092 (0.100)	0.025 (0.029)	0.030 (0.032)
mceliece460896f	82.52 (90.23)	0.114 (0.183)	42.32 (44.47)
mceliece6688128f	204.47 (221.89)	0.201 (0.315)	80.78 (86.91)
mceliece8192128f	237.86 (263.52)	0.217 (0.311)	99.84 (108.15)

At Category 5, relative to ML-KEM-1024:

Quantity	mceliece6688128f	mceliece8192128f
Public key	666× larger	866× larger
Ciphertext	7.5× smaller	7.5× smaller
Key generation	2,227× slower	2,591× slower
Encapsulation	8.2× slower	8.9× slower
Decapsulation	2,666× slower	3,295× slower

6.3 Where the cost actually falls

Read as ratios, Classic McEliece looks unusable. Read as absolute costs, three of those four penalties largely disappear.

Key generation costs 0.20 to 0.24 seconds, once, for a key intended to last years. In the static-key pattern this is not an operational cost.

Encapsulation costs about 0.2 ms against ML-KEM's 0.025 ms. The sender pays a fifth of a millisecond. That is the right place for a cost to fall, since the sender is often the constrained party.

The public key is large but sent once. Using the measured sizes, and assuming the key is cached after first transmission, total bytes on the wire favour Classic McEliece after 767 encapsulations for mceliece6688128f and 997 for mceliece8192128f against ML-KEM-1024. The Category 3 pair crosses over at 561. A static key reused for more than about a thousand sessions is cheaper in bandwidth than ML-KEM rather than more expensive.

Decapsulation is the exception, and the whole cost of the hedge lands there. At 81 to 100 ms per operation against 0.03 ms, one core sustains roughly 10 to 12 Classic McEliece decapsulations per second against roughly 33,000 for ML-KEM-1024. That is a capacity constraint rather than a latency annoyance, and it falls on the receiver, which in most deployments is the server.

The practical summary is short. The hedge is nearly free for the sender, nearly free in key management, and self-amortising in bandwidth after about a thousand sessions. It is expensive in one place only: receiver CPU at session establishment. An architecture terminating many short-lived sessions will feel it immediately. An architecture establishing few long-lived tunnels, which is the pattern the deployed hybrid VPN constructions use, will not notice it. This is the shape of the constraint that bounds where the hedge can be deployed, and it bears directly on Section 7.2.

7 Discussion

7.1 Is the historical confidence shaken?

It is qualified rather than refuted, and the qualification is larger than the current public framing of Classic McEliece admits.

Three statements have to be kept apart. Conflating them is the main analytical hazard in this area.

Binary Goppa public keys are indistinguishable from random linear codes. **This is false**, and now false at the deployed parameter sets, at an estimated cost below the generic-decoding reference. It was not false at deployed parameters in 2022.

A stated security claim of Classic McEliece has fallen. **This is unsupported**. The claim is one-wayness under chosen-plaintext attack with a tight transform to IND-CCA2. No proven attack touches it, and no implemented non-heuristic key recovery against binary irreducible Goppa codes at cryptographic sizes exists.

Classic McEliece is a safe long-term hedge. This is the statement in our title, and it is the **one that has become uncertain**. It has not been refuted. It has stopped being a statement that can be made without citing the state of the literature, which differs from the position that held when the scheme's conservative reputation formed.

The component of the historical confidence that weakened can be identified. Decoding hardness is untouched. The graveyard argument of Section 2.3 is intact, and we regard it as the most durable part of the case. What weakened is the empirical proposition that many people have looked for structure in binary Goppa public keys and found nothing usable. Between 2023 and 2026 several groups looked and found a sequence of progressively cheaper invariants: quadratic relations, graded Betti numbers, higher-order vanishing, and hold-out multiplicities. The structural improvement curve for binary Goppa codes was flat for twelve years. It moved by roughly 184 bits in a single year at the smallest parameter set. Whether or not any of this becomes an executable attack, the stability of structural analysis is no longer available as evidence, because it is no longer a fact.

One asymmetry cuts against complacency. Per Section 2.1, the security of Classic McEliece rests on one proposition with a shallow six-decade improvement curve, and one with no reduction and a twelve-year-old counterexample. Public discussion has consistently attributed the confidence generated by the first to the construction as a whole. That was defensible while nothing threatened the second. It is less defensible now.

7.2 The last-survivor problem

The common architectural use of Classic McEliece is as a diversity hedge. It is a key-establishment mechanism resting on decoding random linear codes rather than on lattice problems, deployed alongside or beneath a lattice mechanism so that a break of one does not compromise the other. That reasoning is sound, and nothing here undermines it. The events of 2026 are independent of anything that might happen to lattice assumptions, and independence is the point of the construction.

The review undermines a different and more common belief: that the hedge is also the safer of the two. On that view, an organisation forced to pick one mechanism to protect data for decades has an obvious conservative choice. That belief rested on the stability of the structural record. The structural record is now the more active research frontier of the two, not the less.

This changes how cryptographic agility should be argued for. The usual case is that nobody can know which assumption will fail, so changing mechanism should be cheap. The 2026 literature strengthens that case in a specific way. The family nominated as the fallback, chosen because its assumptions were believed the best understood, came under rapid pressure from four independent groups with no advance warning visible to a deploying organisation. An organisation that selected Classic McEliece in early 2025 on the then-current literature, and has read nothing since, holds a different asset in September 2026 from the one it believed it acquired. Nothing about its own deployment changed.

We resist the stronger framing, that the question is now which family will be the last survivor. That presumes a contest one family loses, and we have no evidence for it. The likeliest outcome remains that both lattice and code-based assumptions hold, and that current results mature into a better understanding of the boundary rather than into attacks. The code-based family is also not reducible to Classic McEliece, since NIST selected HQC in March 2025. HQC is a much younger construction, and carries a quasi-cyclic structure of the kind Section 2.3 identifies as historically hazardous. The defensible formulation is narrower and more useful. No currently deployable key-establishment mechanism rests on an assumption whose stability can be asserted

from its track record alone. Any architecture depending on such an assertion for one algorithm carries a risk it has probably not priced.

7.3 Institutional position and commercial consequence

The commercial standing of Classic McEliece rests unusually heavily on institutional endorsement rather than on market adoption, and specifically on BSI TR-02102-1, which recommends the scheme for long-term protection of confidential information. Section 2.4.2 of version 2026-01 recommends mceliece460896, mceliece6688128 and mceliece8192128 with their f variants. It grounds the recommendation in the algorithm's provenance rather than in any deployment property, describing the underlying mechanism as

> "about as old as the RSA mechanism and is therefore considered conservative and very thoroughly analysed."

Two features of this matter. The stated rationale is the proposition this paper examines, and the word carrying it is "conservative". The document is dated 23 January 2026, so it predates every result in Sections 5.2 and 5.3. The same holds for the ISO/IEC 18033-2 amendment published in June 2026, which standardised sixteen parameter sets from mceliece460896 upward and excluded mceliece348864. It holds for NIST's March 2025 status report, which declined standardisation on grounds unrelated to structural security, while noting NIST may consider an ISO-based standard once the ISO process completes.

The leading cryptanalysts in this line are employed by the two European agencies whose guidance most directly underwrites the scheme's commercial position. Hemmert and Wiemers are at BSI. Randriambololona is at ANSSI. This is not a conflict and should not be presented as one. National cryptographic agencies employing strong cryptanalysts who publish against widely deployed constructions is the system working as intended, and it argues for confidence in those agencies. It does mean the institutions best placed to judge what these results imply are the institutions whose published positions have not yet been updated.

Whether those positions change is the single external variable that will most affect the practical availability of Classic McEliece in commercial work, more than any further paper. A deployment justified by a BSI recommendation is only as durable as the recommendation. This is a forward-looking judgement about institutional behaviour rather than a finding of this review, and the cryptanalytic literature gives no basis for predicting the direction. BSI's own cryptanalyst published a key-recovery result that he places above the target security levels, which is at least as consistent with a maintained recommendation as with a withdrawn one.

Institutional position is not the only bound on commercial reach, and it may not be the binding one. Section 6 identifies a second constraint that no agency decision affects. Receiver-side cost confines Classic McEliece to architectures establishing few long-lived sessions. On the most favourable reading of the cryptanalysis, the set of deployments that can carry the algorithm remains narrower than the set that would benefit from a non-lattice hedge.

7.4 What follows for a deployer

This paper refuses the question "is it broken", so it owes the reader the question it will answer. Five things follow for someone already running Classic McEliece.

Keep hybrid constructions. Nothing here bears on the lattice half, and the case for combining a lattice mechanism with a structurally distinct one is unchanged. A deployment relying on Classic McEliece alone always carried more risk than one that does not.

Do not treat larger parameter sets as a repair. If the estimates in Section 5.4 hold, moving up a category buys about eleven bits against the attacker while the claimed level rises by one hundred and twenty-nine. That is the opposite of the usual remedy, and it is the most directly actionable finding here.

Do not remove it on the strength of a distinguisher. No stated security claim has fallen, no attack has been implemented at these sizes, and no key has been recovered. Section 5.2 also shows how quickly a claimed key recovery can be answered by a lower bound.

Check the deployment shape against Section 6. Receiver CPU at session establishment is the binding constraint. Few long-lived tunnels are comfortable. Many short-lived sessions are not.

Watch three places rather than the literature as a whole. The submission team’s response page had not moved past 23 June 2026 at our cutoff. BSI TR-02102-1 carries a rationale that predates every result in Sections 5.2 and 5.3. And the cost models of Section 5.3 have yet to be independently audited.

8 Limitations

This review is a snapshot taken on 8 September 2026 of a literature that produced seven significant results in four months. The most consequential is three weeks old, and one paper we cite was revised the day before our cutoff. During drafting we twice found that our own sweep had missed a relevant paper published within the review window, each located only after external review. The second was a refutation of a result we had already written up. A reader should assume the same is true of this version.

We verified the primary source for each result reported. We did not reproduce, audit, or independently check any cost estimate. The Table 1 and Table 3 figures in Section 5.3 come from ePrint 2026/1630 itself, and the tally figures from ePrint 2026/1786. Those cost models are new, and this dispute already contains two occasions where a claimed cost estimate against Classic McEliece did not survive scrutiny. A reader who concludes from this paper that Classic McEliece is broken has drawn a stronger conclusion than the evidence supports, and a stronger conclusion than the authors of the cited work draw.

Three claims appearing in secondary discussion are excluded because we could not verify them. The first is a multi-instance security degradation result for mceliece348864, where secondary characterisations of the session threshold differed by six orders of magnitude. The second is a reported optimised cost for the syzygy distinguisher, differing from the figure the submission team addressed. The third is an award attribution for one of the cited papers. Each may be accurate. None is relied on here.

Our comparison of attack costs against generic-decoding costs uses the T_{ISD} column of the cited paper as the reference. Information-set-decoding estimates are themselves model-dependent, and the correct accounting for memory access is a live dispute that has previously moved the effective security of the smallest parameter set by tens of bits. Statements of the form “below the

ISD reference” describe one paper’s internally consistent accounting rather than settled facts about relative cost. The point applies with most force to the distinguisher, whose storage requirement exceeds the reference attack’s by many orders of magnitude. It applies with less force to the key-recovery tally of Section 5.3, whose state requirement is smaller by four to five orders of magnitude. Esser, May and Zweydinger reported in 2021 that the round-3 parameter sets overestimated security by about twenty bits at the 192-bit level, and ten bits at the 256-bit level. The reference column has moved historically.

The benchmark of Section 6 was taken on a virtualised production host rather than a quiet dedicated machine, and absolute timings on other hardware will differ. Ratios travel better than absolute figures, and both algorithm families were measured on the same host under the same conditions, which is what the comparison requires. Agreement between the August and September runs, within 6 percent on every median and within 2 percent on eleven of fifteen, is our evidence that background load did not distort the result materially. One caveat runs the other way. The Ryzen 9 7950X is a current, high-performing core, so the decapsulation throughput in Section 6.3 should be read as favourable rather than typical. Older or more heavily shared server hardware will sustain fewer than ten Classic McEliece decapsulations per second per core, which strengthens rather than weakens the deployment constraint we draw from it.

Section 7.3 makes a forward-looking judgement about institutional behaviour with no foundation in the cryptanalytic literature. It is an argued opinion rather than a finding, and marked as such where it appears.

9 Conclusion

We reviewed the structural cryptanalysis of the Goppa-code indistinguishability assumption from 2011 to September 2026, verifying each load-bearing claim against its primary source. The widely reported position, that the only structural distinguisher against binary Goppa codes applies at rates far above the deployed range, held until 2023 and no longer holds. Distinguishers reached the deployed parameter sets in 2023 at prohibitive cost, and improved through 2024 and 2025. In August 2026 they reached an estimated 2^{114} to 2^{124} bit operations across all five parameter sets, against a generic-decoding reference of 2^{151} to 2^{287}. That comparison prices bit operations and not the 2^{66} to 2^{72} bits of storage the distinguisher requires, so it does not establish that the attack is cheaper in any security-relevant sense. Five groups published key-recovery constructions against binary Goppa codes between June and September 2026. Three rest on independent algebraic methods and stop short of Classic McEliece parameters, though all three break published challenge instances in practice at claimed security levels up to 248 bits. The rest derive from the August distinguisher. An independent conditional tally of that route places every parameter set below the NIST classical-gate reference for its claimed category, by 16 bits at Category 1 and about 135 bits at Category 5. A separate attempt to convert the same distinguisher into key recovery was refuted in print within days of publication.

No stated security claim of Classic McEliece has fallen. The scheme claims one-wayness under chosen-plaintext attack rather than public-key pseudorandomness. No proven attack reaches that property. Every key-recovery estimate at deployed sizes is heuristic and unimplemented, and no target key at any Classic McEliece parameter set has been recovered. The graveyard argument for the large public key is untouched, and remains the most durable part of the conservative case.

What changed is the evidentiary basis for the word “conservative”. That word was earned by the absence of structural progress rather than by any theorem, and structural progress at the smallest parameter set moved by roughly 184 bits in twelve months. Our principal analytical observation concerns the shape of the new cost profile. It is close to flat across parameter sets, while the claimed security level climbs steeply. Moving from Category 1 to Category 5 raises the tallied attack work by about eleven bits against a claimed gain of one hundred and twenty-nine. The margin below the claimed level therefore widens with parameter size rather than closing. If the published estimates hold, the usual defensive move of selecting larger parameters is unavailable. Our secondary observation concerns the defence. The submission team’s rebuttal structure answered every previous claim successfully, but transfers only partly to the August 2026 result. The cost prong does not apply to figures below 2^{256}, and the property prong does not reach a key-recovery attack.

The question in our title takes a qualified answer. “Classic McEliece is a safe long-term hedge” has not been falsified, and may never be. It has stopped being a sentence that can responsibly stand alone. With its qualifications it remains defensible, and we would still deploy on it. The qualifications are these. Classic McEliece is ISO-standardised and agency-recommended. It is structurally distinct from the lattice family. Its central structural assumption is under active and fast-moving attack, with no proven break to date. Stated without those qualifications, as a claim that a long unbroken record makes it the safe choice, it is the kind of sentence this literature has a habit of overtaking.

Our own implementation of Classic McEliece at BeatQuantum is motivated by cryptoagility needs as well as ongoing research into its business applications. We do not think of Classic McEliece as the ultimate insurance policy for when everything else is broken.

Data Availability

All primary sources cited are publicly available at the identifiers in the References. The concrete cost figures in Section 5.3 are extracted from Tables 1 and 3 of IACR ePrint 2026/1630 and from Table 4 of IACR ePrint 2026/1786, and can be checked against those documents.

The benchmark data underlying Section 6 was generated by the author and is included in this deposit. It comprises the two run outputs of 20 August 2026 and 8 September 2026, both produced by the same harness at N = 10,000, together with the recorded key and ciphertext sizes.

Suggested Citation

S. Pandit, “Classic McEliece is a safe long-term hedge. Famous last words?,” Preprint, Version 1, Zenodo, September 2026. DOI: https://doi.org/10.5281/zenodo.22683800.

Declarations

The author received no external funding for this work and declares no competing interests. This research was conducted in the author’s personal capacity, and the views expressed are the author’s own. The author has a commercial interest in post-quantum cryptography consulting and has deployed Classic McEliece in a research capacity. The conclusions of this paper are less

favourable to that position than the position would suggest, and are stated as the evidence requires.

Acknowledgments

Initial literature identification used several AI research tools, whose outputs were treated as candidate leads rather than findings. Every result reported here was verified by the author against the primary source. Several claims in those outputs were excluded as unverifiable, or found on inspection to misstate their sources. The verification protocol in Section 4 was adopted in response to that experience.

References

[1] J.-C. Faugère, V. Gauthier-Umaña, A. Otmani, L. Perret, and J.-P. Tillich, “A Distinguisher for High-Rate McEliece Cryptosystems,” IEEE Transactions on Information Theory, vol. 59, no. 10, pp. 6830-6844, 2013. IACR ePrint 2010/331.

[2] A. Couvreur, R. Mora, and J.-P. Tillich, “A New Approach Based on Quadratic Forms to Attack the McEliece Cryptosystem,” ASIACRYPT 2023. IACR ePrint 2023/950.

[3] M. Bardet, R. Mora, and J.-P. Tillich, “Polynomial Time Key-Recovery Attack on High Rate Random Alternant Codes,” IEEE Transactions on Information Theory, vol. 70, no. 6, pp. 4492-4511, 2024. arXiv:2304.14757.

[4] H. Randriambololona, “The Syzygy Distinguisher,” EUROCRYPT 2025. IACR ePrint 2024/1193.

[5] A. Lemoine, R. Mora, and J.-P. Tillich, “Understanding the New Distinguisher of Alternant Codes at Degree 2,” Designs, Codes and Cryptography, vol. 93, 2025. IACR ePrint 2025/531.

[6] T. Hemmert and A. Wiemers, “Distinguishing Goppa Codes Using Higher-Order Vanishing,” CRYPTO 2026. IACR ePrint 2025/1661.

[7] P. Briaud, A. Lemoine, H. Randriambololona, and J.-P. Tillich, “A Heuristic Subexponential Attack on the McEliece Cryptosystem,” IACR ePrint 2026/1232, June 2026.

[8] T. Hemmert, “Key Recovery for the McEliece Cryptosystem Using Higher-Order Vanishing,” IACR ePrint 2026/1339, July 2026.

[9] A. Ghoshal, Y. Ishai, A. Jain, and N. Sun, “Quasipolynomial Cryptanalysis of the McEliece Cryptosystem (or: PIR Meets McEliece),” IACR ePrint 2026/1630, August 2026.

[10] K. Vedenev, “Extending Distinguishing to Key Recovery for Subfield Subcodes of GRS Codes,” IACR ePrint 2026/1747, September 2026.

[11] D. Apon, “An Algebraic-Geometry Lower Bound against the ePrint:2026/1747 McEliece Key-Recovery Attack,” IACR ePrint 2026/1810, August 2026.

[12] M.-J. O. Saarinen, “HOVER: Higher-Order Vanishing Endomorphism Recovery,” IACR ePrint 2026/1778, August 2026.

[13] M.-J. O. Saarinen, “Bit Operation Cost of ‘Holdout’ Key-Recovery Attacks Against Classic McEliece,” IACR ePrint 2026/1786, August 2026, revised 7 September 2026.

[14] S. Jafarzade Mojaveri and A. Khosravi, “The McEliece Cryptosystem After Nearly Five Decades: A Survey of Security, Cryptanalysis, and Future Directions,” IACR ePrint 2026/1512, July 2026.

[15] A. Esser, A. May, and F. Zweydinger, “McEliece Needs a Break: Solving McEliece-1284 and Quasi-Cyclic-2918 with Modern ISD,” IACR ePrint 2021/1634, 2021.

[16] Classic McEliece Team, “Notes on Recent Speculation That a mceliece348864 Key-Recovery Attack Uses Only 2^{610} Operations,” 23 June 2026. https://classic.mceliece.org/mceliece-610-20260623.pdf

[17] Classic McEliece Team, “Notes on a Recent Claim That a mceliece348864 Distinguisher Uses Only 2^{529} Operations,” 17 April 2025. https://classic.mceliece.org/mceliece-529-20250417.pdf

[18] Classic McEliece Team, “Guide for Security Reviewers,” Round-4 submission, 23 October 2022. https://classic.mceliece.org/mceliece-security-20221023.pdf

[19] Classic McEliece Team, “Design Rationale,” Round-4 submission, 23 October 2022. https://classic.mceliece.org/mceliece-rationale-20221023.pdf

[20] Federal Office for Information Security (BSI), “Cryptographic Mechanisms: Recommendations and Key Lengths,” BSI TR-02102-1, version 2026-01, 23 January 2026.

[21] National Institute of Standards and Technology, “Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process,” NIST IR 8545, March 2025.

[22] ISO/IEC 18033-2:2006/Amd 2:2026, “Information technology, Security techniques, Encryption algorithms, Part 2: Asymmetric ciphers, Amendment 2,” June 2026.

[23] D. J. Bernstein, “Grover vs. McEliece,” PQCrypto 2010.

[24] G. Kachigar and J.-P. Tillich, “Quantum Information Set Decoding Algorithms,” PQCrypto 2017. arXiv:1703.00263.

[25] E. Kirshanova, “Improved Quantum Information Set Decoding,” PQCrypto 2018. arXiv:1808.00714.

[26] V. M. Sidelnikov and S. O. Shestakov, “On Insecurity of Cryptosystems Based on Generalized Reed-Solomon Codes,” Discrete Mathematics and Applications, vol. 2, no. 4, pp. 439-444, 1992.

[27] A. Couvreur, A. Otmani, and J.-P. Tillich, “Polynomial Time Attack on Wild McEliece over Quadratic Extensions,” EUROCRYPT 2014. IACR ePrint 2014/112.