



Notes on recent developments concerning Classic McEliece

Date: 29th September, 2026

Classic McEliece is a cryptographic key agreement scheme that was a candidate in the fourth round of the NIST standardization process for post-quantum cryptography [NIST-PQC] and was standardized by ISO/IEC in [ISO-IEC] in June 2026. Since 2020, the BSI has recommended Classic McEliece, currently with selected parameter sets from [ABC+22a], in its Technical Guideline TR-02102-1 [BSI-TR-02102]. Classic McEliece is a variant of the McEliece cryptosystem, which was published in 1978 [McE78] and is thus one of the oldest public-key cryptosystems. However, because of its comparatively very large public keys, it has not found widespread use.

In recent months, a number of publications have made progress in the cryptanalysis of the McEliece cryptosystem. The BSI itself has also contributed to the analysis of the McEliece scheme [HW26], [Hem26]. In this document, the BSI provides an initial assessment of the recent results and a preliminary evaluation of how to handle Classic McEliece.

What does this mean for the use of Classic McEliece?

Classic McEliece has particularly small ciphertexts but very large public keys. It is therefore usually used only in special deployment scenarios, e.g. applications with long-lived static keys in which many ciphertexts have to be exchanged.

Classic McEliece should currently no longer be used in new developments or when planning new cryptographic applications. For key agreement, it is advised to use FrodoKEM, ML-KEM or HQC (once the corresponding standard has been published) in accordance with the recommendations of TR-02102-1.

The current recommendation of TR-02102-1 for Classic McEliece is limited to hybrid use. Here, the negotiated Classic McEliece key is combined with a further key negotiated using a classical scheme in such a way that both schemes must be broken in order to obtain the hybrid key. For the sole use of classical schemes, the general recommendation for use remains in effect until the end of 2031. At present, such a hybrid combination with Classic McEliece therefore still provides at least the security of the classical scheme.

The recommendation on Classic McEliece will be revised in the next version of TR-02102-1 based on the ongoing analysis of the results discussed below and, if applicable, further results. This version is expected to be published in early 2027.

What does this mean for the migration to post-quantum cryptography?

The new assessment of the security of Classic McEliece does not change the general, urgent recommendation to complete the migration to post-quantum cryptography for key agreement by 2031 at the latest. The new results (see below) underline how important it is to pay attention to crypto-agility during the migration because cryptanalytic advances can never be ruled out.

The current situation shows that the recommendation to use post-quantum cryptography in hybrid mode has proven its worth. We are witnessing a dynamic development that must continue to be monitored. Because each of the new post-quantum schemes is to be combined with a further scheme, the impact that a weakened scheme could have is mitigated.

The cryptanalysis results discussed below relate explicitly only to the McEliece cryptosystem and thus to the post-quantum scheme Classic McEliece. The other recommended post-quantum schemes, FrodoKEM and ML-KEM, are based on different mathematical principles and are not affected by the results. The HQC scheme, which is planned to be recommended in the future, is not affected either, even though, like Classic McEliece, it is based on error-correcting codes. HQC can therefore offer an important alternative to the lattice-based post-quantum schemes FrodoKEM and ML-KEM going forward.

Assessment of the results on the security of Classic McEliece

A central component of Classic McEliece is so-called binary Goppa codes. These are a family of error-correcting codes for which efficient error-correction algorithms are known. Roughly speaking, key generation in Classic McEliece first randomly selects a binary Goppa code with specified parameter sizes. The generated secret key consists of information about the Goppa code structure of the generated code that allows for efficient error correction, while the public key essentially consists of a random basis of the generated code. The security of Classic McEliece rests on the assumption that error correction for this code is not efficiently possible given only this randomly chosen basis.

For a long time, the best attack against the McEliece cryptosystem was considered to be treating the code in question as a general linear code and applying generic error-correction algorithms (so-called information set decoding, ISD for short) to it. Decoding general linear codes is an NP-hard problem, and apart from small improvements, the complexity of the best ISD algorithms has remained remarkably stable despite intensive research over the last 60 years, see [ABC+22b, Chapter 3].

By contrast, structural attacks, which aim to reconstruct the secret Goppa code structure, and thus the secret key, from the public key, long had a far higher complexity than ISD algorithms. Until recently, the best such key-recovery attack essentially consisted of a brute-force search combined with the so-called support splitting algorithm [Sen00]. Even for the smallest Classic McEliece parameters, this algorithm had an estimated complexity of over 2^{3000} , far above the security level of at least 120 bits recommended in TR-02102.

In the recent past, particularly in the last few months, several papers have been published that fundamentally change this situation. Some of them build on earlier ideas and work that cannot all be presented in detail in this document.

- [Ran24] describes the first algorithm that distinguishes binary Goppa codes from general linear codes with a complexity that is asymptotically slightly subexponential in the error-correction capacity of the underlying code. For binary Goppa codes with the parameters recommended in TR-02102-1, however, the stated estimated complexity of this distinguisher is at least 2^{683} , according

to the estimates in [Ran24]. It should be noted that such a distinguisher alone does not threaten the security assumption of Classic McEliece. However, it can be an indication of a structural weakness that could possibly be developed further into a key-recovery attack.

- [HW26] likewise describes a distinguisher between binary Goppa codes and general linear codes that yields a lower complexity than [Ran24] for Classic McEliece parameters. At no less than 2^{550} for the Classic McEliece parameter sets recommended in TR-02102-1, however, the complexity is still above the targeted security level.
- [BLT26] describes an efficient key-recovery algorithm for the McEliece cryptosystem when binary Goppa codes with very specific parameter choices, namely very high code rates, are used. However, the attack is not applicable to Classic McEliece parameters.
- [BLRT26] describes another distinguisher and a key-recovery algorithm based on it, which the authors state should also be applicable to Classic McEliece parameter sets. For the Classic McEliece parameter sets recommended in TR-02102-1, however, the estimated complexity is at least 2^{742} .
- [Hem26] extends the distinguisher from [HW26] into a key-recovery algorithm which, according to the author, should also be applicable to Classic McEliece parameter sets and have roughly the same complexity as the distinguisher [HW26]. [Saa26a] describes an extension of this attack, which, however, does not yield a significant improvement in complexity for Classic McEliece parameter sets.
- [GIJS26] describes a distinguisher with claimed asymptotically quasi-polynomial complexity and states a complexity of between 2^{120} and 2^{124} operations for the Classic McEliece parameter sets recommended in TR-02102-1. In addition, the authors describe a heuristic key-recovery algorithm that relies on assumptions still to be verified and, according to the authors, has an estimated complexity of between 2^{140} and 2^{148} for the Classic McEliece parameter sets recommended in TR-02102-1.
- [Ved26] likewise describes a key-recovery algorithm that is based on the distinguisher from [GIJS26] and which, according to the author, is applicable to Classic McEliece parameter sets. The algorithm requires several runs of the distinguisher from [GIJS26], whose complexity thus provides a lower bound.
- [Saa26b] describes how an adaptation of the approach from [GIJS26] was used to break a specific key-recovery challenge of the Technology Innovation Institute.
- [Wei26] describes, according to the author, an improvement of the distinguisher from [GIJS26] as well as a key-recovery algorithm based on it. For the parameter sets recommended in TR-02102-1, a complexity of between 2^{99} and 2^{102} operations for key recovery is stated.

It is important to note that the most recent publications in particular have not yet completed a full peer-review process and require thorough analysis. In particular, it must be examined carefully to what extent the claimed complexity estimates are accurate and the cost model is realistic, for example with regard to memory access costs. The BSI is also monitoring and analyzing the cryptanalysis results mentioned. **So far, the results do not enable a practical attack on Classic McEliece with the parameter sets recommended in TR-02102-1.** However, a significant improvement in structural attacks against the McEliece cryptosystem with binary Goppa codes can be observed over the last few months. **Likewise, further developments and improvements of the results published so far are conceivable and to be expected.**

Literature

[ABC+22a] M. R. Albrecht, D. J. Bernstein, T. Chou, C. Cid, J. Gilcher, T. Lange, V. Maram, I. von Maurich, R. Misoczki, R. Niederhagen, K. G. Paterson, E. Persichetti, C. Peters, P. Schwabe, N. Sendrier, J. Szefer, C. J. Tjhai, M. Tomlinson, and W. Wang (2022). Classic McEliece: conservative code-based cryptography: cryptosystem specification. <https://classic.mceliece.org/mceliece-spec-20221023.pdf>

- [ABC+22b] M. R. Albrecht, D. J. Bernstein, T. Chou, C. Cid, J. Gilcher, T. Lange, V. Maram, I. von Maurich, R. Misoczki, R. Niederhagen, K. G. Paterson, E. Persichetti, C. Peters, P. Schwabe, N. Sendrier, J. Szefer, C. J. Tjhai, M. Tomlinson, and W. Wang (2022): Classic McEliece: conservative code-based cryptography: guide for security reviewers. <https://classic.mceliece.org/mceliece-security-20221023.pdf>
- [BLRT26] Pierre Briaud, Axel Lemoine, Hugues Randriambololona, Jean-Pierre Tillich: A Heuristic Subexponential Attack on the McEliece Cryptosystem. Cryptology ePrint Archive, <https://eprint.iacr.org/2026/1232>
- [BLT26] Magali Bardet, Axel Lemoine, Jean-Pierre Tillich (2026). An Attack on the CFS Scheme and on TII McEliece Challenges. In: Heninger, N., Rosulek, M. (eds) Advances in Cryptology – CRYPTO 2026. CRYPTO 2026. Lecture Notes in Computer Science, vol 16803. Springer, Cham.
- [BSI-TR-02102] Bundesamt für Sicherheit in der Informationstechnik (2026). BSI-TR-02102-1: Kryptographische Verfahren: Empfehlungen und Schlüssellängen, <https://www.bsi.bund.de/TR-02102>
- [GIJS26] Ashrujit Ghoshal, Yuval Ishai, Aayush Jain, Nuo Zhou Sun: Quasipolynomial Cryptanalysis of the McEliece Cryptosystem (or: PIR Meets McEliece). Cryptology ePrint Archive, <https://eprint.iacr.org/2026/1630>
- [Hem26] Tobias Hemmert: Key recovery for the McEliece cryptosystem using higher-order vanishing. Cryptology ePrint Archive, <https://eprint.iacr.org/2026/1339>
- [HW26] Tobias Hemmert, Andreas Wiemers (2026). Distinguishing Goppa Codes Using Higher-Order Vanishing. In: Heninger, N., Rosulek, M. (eds) Advances in Cryptology – CRYPTO 2026. CRYPTO 2026. Lecture Notes in Computer Science, vol 16803. Springer, Cham.
- [ISO-IEC] ISO/IEC 18033-2:2006/Amd 2:2026 Information technology – Security techniques – Encryption algorithms – Part 2: Asymmetric ciphers, Amendment 2
- [McE78] Robert J. McEliece (1978). A public-key cryptosystem based on algebraic. Coding Thv, 4244(1978), 114-116.
- [NIST-PQC] <https://csrc.nist.gov/projects/post-quantum-cryptography>
- [Ran24] Hugues Randriambololona (2025). The syzygy Distinguisher. In: Fehr, S., Fouque, PA. (eds) Advances in Cryptology – EUROCRYPT 2025. EUROCRYPT 2025. Lecture Notes in Computer Science, vol 15606. Springer, Cham.
- [Saa26a] Markku-Juhani O. Saarinen: HOVER: Higher-Order Vanishing Endomorphism Recovery. Cryptology ePrint Archive, <https://eprint.iacr.org/2026/1778>
- [Saa26b] Markku-Juhani O. Saarinen: Two-Anchor Holdout/Hermite: Solving the TII-254 McEliece Key Recovery Challenge. Cryptology ePrint Archive, <https://eprint.iacr.org/2026/1986>
- [Sen00] Nicolas Sendrier (2000). Finding the permutation between equivalent linear codes: The support splitting algorithm. IEEE Transactions on Information Theory, 46(4), 1193-1203.
- [Ved26] Kirill Vedenev: Extending Distinguishing to Key Recovery for Subfield Subcodes of GRS codes. Cryptology ePrint Archive, <https://eprint.iacr.org/2026/1747>
- [Wei26] Stephen A. Weis: Improving GIJS Key Recovery for Classic McEliece. Cryptology ePrint Archive, <https://eprint.iacr.org/2026/1984>