



FuLeakage: Breaking FuLeeca by Learning Attacks

Felicitas Hörmann^{1,2}  and Wessel van Woerden³ 

¹ Institute of Communications and Navigation, German Aerospace Center (DLR),
Oberpfaffenhofen–Wessling, Germany

`felicitas.hoermann@dlr.de`

² School of Computer Science, University of St. Gallen, St. Gallen, Switzerland

³ Université de Bordeaux, IMB, Inria, Talence, France

`wessel.van-woerden@math.u-bordeaux.fr`

Abstract. We show that the *code-based* signature scheme FuLeeca is vulnerable to *lattice-based* cryptanalysis. A classical attack using lattice-basis reduction lowers the claimed security levels significantly, whereas learning techniques allow to recover the secret key from the leakage of less than 175,000 signatures in practice. The exploitation of ideal structures and efficient quantum algorithms further yields a full quantum break.

Keywords: Lee metric · lattices · cryptanalysis · learning attack.

FuLeeca is the first signature scheme based on *Lee-metric* codes and was presented at CBCrypto 2023 [4]. Moreover, FuLeeca was submitted [3] to the additional call for digital signatures, that NIST announced in 2022 after three rounds of their first standardization project for post-quantum cryptography had led to little diversity in the used security primitives. Even though FuLeeca is *code-based*, we show that it is closely related to known *lattice* schemes such as NTRUSIGN. This proximity allows us to mount multiple key-recovery attacks that exploit techniques from lattice-based cryptography and fully break the system for all proposed parameter sets.

The *Lee weight* of an element $x \in \mathbb{F}_p$ can be defined as $\text{wt}_L(x) := |x|$, if we identify the finite field \mathbb{F}_p of odd prime order with the set $\{-\frac{p-1}{2}, \dots, \frac{p-1}{2}\}$. The Lee weight extends additively to a vector $\mathbf{x} \in \mathbb{F}_p^n$ and induces the *Lee metric* between two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{F}_p^n$ as $d_L(\mathbf{x}, \mathbf{y}) = \text{wt}_L(\mathbf{x} - \mathbf{y}) = \sum_{i=1}^n \text{wt}_L(x_i - y_i)$. Let us consider a Lee-metric code $\mathcal{C} \subset \mathbb{F}_p^n$ and define the full-rank lattice $\mathcal{L}_1 := \mathcal{C} + p\mathbb{Z}^n \subset \mathbb{R}^n$. Any codeword $\mathbf{c} \in \mathcal{C}$ can implicitly be lifted to $\tilde{\mathbf{c}} \in \mathcal{L}_1$ and, in particular, $\text{wt}_L(\mathbf{c}) = \|\tilde{\mathbf{c}}\|_1 := \sum_{i=1}^n |\tilde{c}_i|$ applies. The ℓ_1 -norm is further closely related to the Euclidean ℓ_2 -norm, and therefore, shortness and closeness in terms of the Lee metric on \mathcal{C} translate more or less directly into shortness and closeness in terms of the Euclidean ℓ_2 -metric on \mathcal{L}_1 .

FuLeeca can be interpreted as a hash-and-sign scheme which uses the hashed message as an erroneous codeword of a quasi-cyclic code and a decoded low-weight codeword as the signature. The secret key is a generator matrix $\mathbf{G}_{\text{sec}} \in \mathbb{F}_p^{k \times n}$ with $n = 2k$ and can be fully described by means of the secret vector

$\mathbf{g} = (\mathbf{a} \mid \mathbf{b})$ with $\mathbf{a}, \mathbf{b} \in \mathbb{F}_p^k$. The i -th row of \mathbf{G}_{sec} is $(\text{shift}^{i-1}(\mathbf{a}) \mid \text{shift}^{i-1}(\mathbf{b}))$, where $\text{shift}(\mathbf{x}) = (x_k, x_1, \dots, x_{k-1})$ denotes the circular shift of a vector $\mathbf{x} = (x_1, \dots, x_k)$. The goal for key-recovery attacks is thus to find the secret vector \mathbf{g} or any of its quasi-circular shifts, which we consider equivalent for simplicity.

It is vital to the signing procedure that \mathbf{g} has low Lee weight and is thus short in the lattice \mathcal{L}_1 . We can use lattice-reduction algorithms such as BKZ [5] to find a short vector in \mathcal{L}_1 of similar Lee weight and potentially use it as an equivalent secret key to forge signatures. For these parameters BKZ has a heuristic runtime of $2^{0.292\beta + o(n)}$ for $\beta \geq 0.95n$. This attack was already considered in the FuLeeca specification and performs worse than code-based approaches. However, the same idea can be improved by considering a sublattice of lower dimension in which the secret vector \mathbf{g} is *unusually* short and thus uniquely and more efficiently recoverable. The key observation is that no wrapping modulo p takes place during FuLeeca’s signing step, due to the very large modulus $p = 65,521$ that is chosen for all proposed parameter sets. As a result, all FuLeeca signatures lie in the sublattice $\mathcal{L}_2 \subset \mathcal{L}_1$ that is generated by the rows of the secret generator matrix \mathbf{G}_{sec} . Even though we have a-priori no access to a basis of \mathcal{L}_2 , a small sample of FuLeeca signatures of size, say, 100 is enough to construct one and proceed with the BKZ attack. Note further that \mathcal{L}_2 has rank $n/2$ and that the Euclidean length of \mathbf{g} is approximately 15% of the Gaussian heuristic of \mathcal{L}_2 and thus unusually short. This allows to heuristically reduce the cost of BKZ to find \mathbf{g} down to $2^{\frac{0.292n}{4} + o(n)}$ and hence the security levels of the parameter sets FuLeeca-I, FuLeeca-III, and FuLeeca-V from 160, 224, and 288 bits to 111, 155, and 199 bits, respectively. A full break of FuLeeca with the same amount of signatures is feasible in quantum-polynomial time, when the ideal structure of \mathcal{L}_2 is exploited even further.

We further derived a polynomial-time learning attack that recovers the secret key with less than 175,000 available FuLeeca signatures for every parameter set. Learning attacks originate from the break of the lattice-based GGH and NTRUSIGN schemes [2]. There, they abuse the fact that all signatures lie in the parallelepiped spanned by the short vectors of the secret basis. With enough signatures at hand, one can thus learn the outline of this parallelepiped and hence recover the secret.

In the FuLeeca setting, a similar bias is introduced by the *concentration step* within the signing algorithm. This part tries to alter the signature such that its Lee weight and the number of sign matches with the message hash lie in prescribed intervals. The trial-and-error process successively adds or subtracts the rows of \mathbf{G}_{sec} and checks for improvements. However, the first row is always considered first, then the second row, and so on. This introduces a bias in the signature distribution that is visualized for two dimensions in Figure 1. We average over the outer product of the signature vectors and exploit some properties of the scheme to recover the FuLeeca keys in polynomial time. A sample of 175,000 FuLeeca signatures is enough to break instances of every parameter set.

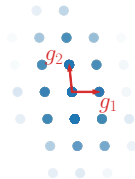


Fig. 1: Signature bias in dimension 2.

In summary, FuLeeca is not secure for any parameter set proposed in [3]. The described attacks show once more that code- and lattice-based cryptography are indeed closely related, even though they *seem* different at first sight. Thus, attacks from both sides should be taken into account whenever a new scheme is suggested to ensure reliable security estimates.

Acknowledgments. This abstract is based on [1] and all experiments presented therein were carried out using the PlaFRIM experimental testbed, supported by Inria, CNRS (LABRI and IMB), Université de Bordeaux, Bordeaux INP and Conseil Régional d'Aquitaine. W. van Woerden was supported by the CHARM ANR-NSF grant (ANR-21-CE94-0003).

References

1. Hörmann, F., van Woerden, W.: FuLeakage: Breaking FuLeeca by learning attacks. Cryptology ePrint Archive, Paper 2024/353 (2024)
2. Nguyen, P.Q., Regev, O.: Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. In: EUROCRYPT 2006. LNCS, vol. 4004, pp. 271–288 (2006)
3. Ritterhoff, S., Maringer, G., Bitzer, S., Weger, V., Karl, P., Schamberger, T., Schupp, J., Wachter-Zeh, A.: FuLeeca. Round 1 of the NIST call for additional digital signatures (2023)
4. Ritterhoff, S., Maringer, G., Bitzer, S., Weger, V., Karl, P., Schamberger, T., Schupp, J., Wachter-Zeh, A.: FuLeeca: A Lee-based signature scheme. In: CBCrypto 2023. LNCS, vol. 14311, pp. 56–83 (2023)
5. Schnorr, C., Euchner, M.: Lattice basis reduction: Improved practical algorithms and solving subset sum problems. Math. Program. **66**, 181–199 (1994)