

“Preview Writeup”: In anticipation of a package submission to the NIST Threshold Call

Title: PQarrots: Macaw, Kea and Kakapo

Subtitle: Threshold primitives from (isogeny-based) group actions

Version: 0.2 (2026-01-18)¹

Team name: PQarrots Team: Post Quantum Action for Round RObin Threshold Schemes

Team members: Marius A. Aardal, Shahla Atapoor, Karim Baghery, Andrea Basso, Xavier Bonnetain, Giacomo Borin, Daniele Cozzo, Pierrick Dartois, Luca De Feo, Max Duparc, Jonathan K. Eriksen, Tako Boris Fouotsa, Arthur Herlédan Le Merdy, Riccardo Invernizzi, Samuel Jaques, Yi-Fu Lai, Dania Lazzarini, Jason T. LeGrow, Luciano Maino, Jonas Meers, Michael Meyer, Sikhar Patranabis, Robi Pedersen, Giacomo Pope, Doreen Riepel, Damien Robert, Ryan Rueger, Sina Schaeffler, André Schrottenloher, Frederik Vercauteren

Abstract: Cryptographic group actions offer a flexible framework for instantiating plausibly post-quantum schemes, effectively generalizing core ideas behind classical discrete logarithm cryptography. In particular, the group structure allows for an (almost) immediate application of well-known threshold secret-sharing techniques, to obtain distributed post-quantum cryptographic protocols such as digital signatures and public key encryption.

This document previews PQarrots, a planned package submission to the NIST Multi-Party Threshold Cryptography (MPTC) Call, based on isogeny cryptographic group actions, and containing an instantiation of a threshold signing primitive, a threshold public key encryption and a distributed key generation procedure. We explain advantages and limitations of group actions threshold schemes in general and of isogenies in particular. We also present preliminary results on the expected performance of our schemes.

Proposed crypto-systems: (I) Macaw (Category S1); (II) Kea (Category S2); (III) Kakapo (Category S4).

Keywords: Threshold Cryptography; NIST Threshold Call; Isogenies; Group Actions; Distributed Key Generation

¹Preliminary version submitted to NIST-MPTC for review. Supersedes version 0.1 sent on 2025-01-12.

Preview writeup. This document is provided to NIST for online publication, to foster public awareness and support public discussion within the scope of the NIST First Call for Multi-Party Threshold Schemes [NIST-IR8214C]. This “preview writeup” represents a good-faith plan for a subsequent “package submission”. However, until the deadline for package submission, the team may still modify its own composition and the submission plan, including possible changes to the technical scope, and/or the used techniques or achieved results.

Team members: Marius A. Aardal^{i1,a1a2*}, Shahla Atapoor^{i2,a3}, Karim Baghery^{i3,a3}, Andrea Basso^{i4,a2}, Xavier Bonnetain^{i5,a4}, Giacomo Borin^{i6,a2a5*}, Daniele Cozzo^{i7,a6}, Pierrick Dartois^{i8,a7}, Luca De Feo^{i9,a2}, Max Duparc^{i10,a8*}, Jonathan K. Eriksen^{i11,a3}, Tako Boris Fouotsa^{i12,a9}, Arthur Herlédan Le Merdy^{i13,a3}, Riccardo Invernizzi^{i14,a3*}, Samuel Jaques^{i15,a10}, Yi-Fu Lai^{i16,a3}, Dania Lazzarini^{i17,a11*}, Jason T. LeGrow^{i18,a12}, Luciano Maino^{i19,a13a11}, Jonas Meers^{i20,a14*}, Michael Meyer^{i21,a15}, Sikhar Patranabis^{i22,a16}, Robi Pedersen^{i23,a17}, Giacomo Pope^{i24,a18a19}, Doreen Riepel^{i25,a20}, Damien Robert^{i26,a21}, Ryan Rueger^{i27,a2a23*}, Sina Schaeffler^{i28,a22a2*}, André Schrottenloher^{i29,a7}, Frederik Vercauteren^{i30,a3}

Open Researcher and Contributor Identifiers (ORCID): i1 (0009-0003-0674-6460); i2 (0000-0002-6035-9520); i3 (0000-0001-7213-8496); i4 (0000-0002-3270-1069); ; i6 (0009-0001-7311-3802); i7 (0000-0001-5289-3769); i8 (0009-0008-2808-9867); i9 (0000-0002-9321-0773); i10 (0009-0001-4179-9547); i11 (0009-0000-3040-2965); i12 (0000-0003-1821-8406); i13 (0009-0007-6116-6863); i14 (0000-0002-2271-6822); i15 (0000-0003-0966-8114); i16 (0000-0002-1346-9372); i17 (0009-0008-7103-3658); i18 (0000-0002-6239-6616); i19 (0009-0005-4495-5102); i20 (0000-0002-1755-8153); i21 (0009-0004-5680-8962); i22 (0000-0002-2309-7939); i23 (0000-0001-5120-5709); i24 (0009-0004-0394-3650); i25 (0000-0002-4990-0929); i26 (0000-0003-4378-4274); i27 (0000-0002-4349-8609); i28 (0009-0004-8039-2100); i29 (0000-0002-1329-8630); i30 (0000-0002-7208-9599)

Affiliations: ^{a1} Aarhus University. ^{a2} IBM Research Europe. ^{a3} COSIC - KU Leuven. ^{a4} Inria Nancy. ^{a5} University of Zurich. ^{a6} IMDEA Software Institute. ^{a7} Inria Rennes. ^{a8} EPFL Lausanne. ^{a9} The University of Manchester. ^{a10} University of Waterloo. ^{a11} Université Libre de Bruxelles. ^{a12} Virginia Polytechnic Institute and State University. ^{a13} University of Birmingham. ^{a14} Ruhr University Bochum. ^{a15} University of Regensburg. ^{a16} IBM Research India. ^{a17} Technical University of Denmark. ^{a18} NCC Group. ^{a19} Bristol University. ^{a20} CISA Helmholtz Center for Information Security. ^{a21} Inria Bordeaux. ^{a22} ETH Zurich. ^{a23} Technische Universität München

Associateship clarifications: * Ph.D. student.

Main contacts:

- **Team mailing list:** tga-nist@groupes.renater.fr
- **Primary technical contact person:** Giacomo Borin, pqarrots@gbor.in
- **Secondary contact person:** Luca De Feo, pqarrots@defeo.lu

Produced by humans. The team hereby confirms that the content in this preview writeup: (i) was produced by the team members, and (ii) was not produced by generative artificial intelligence (AI), with the possible exception of AI-proposed grammar improvements, minor integrated suggestions, or some well-identified and short localized portions of auxiliary content (e.g., some illustration); and (iii) was proofread by the team members.

1. Introduction

Cryptographic group actions offer a flexible framework for instantiating plausibly post-quantum schemes [Cou06; ADMP20]. A commutative group action $\star : G \times X \rightarrow X$ of a commutative group G onto a set X is called *cryptographic* if the operation \star is efficiently computable and hard to invert. Formally, the **gaDLOG** states that given elements $x, y \in X$ such that $y = g \star x$ it is hard to recover g . Every group action mentioned in this document will be cryptographic and commutative, even when the adjectives are omitted.

Cryptographic group actions can be instantiated from several pre- and post-quantum hard problems. The PQarrots submission leverages post-quantum *isogeny-based group actions* [CLMPR18; BKV19; DEFMI+25] to instantiate the following threshold primitives:

- Macaw, Signature Scheme (category S1) targeting small numbers of parties with Dishonest majority.
- Kea, Public Key Encryption (PKE) Scheme with threshold decryption (category S2), also targeting small numbers of parties with Dishonest majority.
- Kakapo, Distributed Key Generation (DKG) mechanism (category S4), compatible with both the signature and PKE schemes. For the larger security levels Kakapo is practical for up to 4 parties.

We argue PQarrots is relevant to the NIST call for the following reasons:

1. **Quantum-safety, Diversity:** The schemes are plausibly quantum-safe, based on substantially different assumptions compared to lattice schemes, and with a clear research track record.
2. **Compactness:** across various trade-offs, signatures can be as compact as 2.8 KB, public keys as 256 B, ciphertexts as 768 B at the highest security level, although not all minima are necessarily simultaneously attainable.
3. **Syntactic Compatibility:** Group actions naturally generalize the existing discrete logarithm framework. In particular, their strong syntactical similarities lead to a streamlined protocol design and security analysis. Moreover, all our schemes remain agnostic to the specific implementation of the underlying group action, allowing for future performance improvements without affecting their correctness or security.

We also want to highlight the limitations of the submission:

1. **Quantum sub-exponential attack:** it is well known that Kuperberg's algorithm [Kup05; Reg04; Kup13; BS20; Pei20] breaks commutative **gaDLOG** in quantum subexponential time. However, at the present moment the community has reached no consensus on the practical quantum security of the isogeny instantiations.
2. **Larger sizes:** to mitigate the impact of the quantum attacks, but also because of intrinsic limitations of the group action framework [BGZ23], sizes are not as compact as what the community has come to expect from isogeny-based schemes [AAABC+24].

3. Inherently **sequential structure** of the threshold primitives (aka *Round Robin* communication structure), leading to a high number of rounds and increased protocol latency. Some of this can be mitigated through preprocessing.

At the moment of writing, the best candidate for a scalable isogeny-based group action is qt-Pegasis [DEFMI+25; DEIV25]. We plan to base our implementation on it, leaving open the possibility to upgrade to a more efficient one as research progresses. Additionally, we include some isogeny-specific gadgets [Fou25] that let us greatly speed up some of the protocols.

2. Specification

The package will be organized as follows:

1. An in-depth analysis of the threshold schemes and their security properties, relying only on the general *Cryptographic group action framework*. The security of these schemes considers active adversaries and relies on assumptions from the group action literature.
2. A *group-action-agnostic* specification of the schemes. We follow a *Plug and play* design choice: the schemes can be instantiated with any implementation of a group action satisfying the listed axioms.
3. The specification of the isogeny group action following [DEIV25] and of other isogeny-specific gadgets.
4. A report on the (quantum) security of our assumptions.

Let T be the threshold and N the number of parties. We summarize the main design choices for our schemes.

Security assumptions. There are other standard assumption of relevance for our submission, closely related to **gaDLOG**, that generalize well-known ones from the discrete logarithm literature: the Group Action Decisional Diffie-Hellman (**gaDDH**) assumption and the Group Action Computational Diffie-Hellman (**gaCDH**) assumption. Note: for quantum adversaries the latter is equivalent to **gaDLOG** [MZ22]. Lastly, the Group Action Strong Computational Diffie-Hellman (**gaStCDH**) assumption, a variant of **gaCDH** where the adversary has access to an additional fixed-input **gaDDH** oracle [DHKCLR23].

Depending on the secret-sharing mechanism used we may also consider the use of the *Power DDH assumption* [DM20]. For the security proof we also use the Random Oracle Model and, if needed, the *Algebraic Group Action Model* (AGAM) [DHKCLR23]. The core idea of the AGAM is that whenever any algorithm provides a set element it needs to also add an explanation of it, in the form of a pair in $G \times X$.

Scheme design for the Threshold Signature. The main idea of the signature scheme is to have a *Round Robin* message-independent preprocessing phase. This phase is run by T parties, inherently sequential, and terminates with the generation of the commitment. The online signing

phase instead requires 2 rounds of online interactions, one to generate randomness and one to generate the final signature, neither requiring expensive group action evaluations.

Different strategies can be employed to authenticate the information sent during the commitment. For this we propose an instantiation using the interactive proof-of-knowledge from [MJ23], a more efficient solution with respect to the ones proposed in the literature, but whose security relies on the AGAM [DHKKLR23] and **gaStCDH** assumption. The cost of the procedure is dominated by the computation of $2T \cdot t_r$ sequential group actions. The parameter t_r grows linearly with the security parameter and is inversely proportional to the logarithm of the public key size. In practice, t_r ranges between 11 and 49. As a recovery mechanism, the scheme can be augmented with an interactive identifiable abort protocol [PKNRT25].

Scheme design for the Threshold PKE. On a high level, the threshold PKE follows the hybrid encryption paradigm (PKE+ symmetric key encryption). Encryption requires a single group action computation, whereas decryption is performed in a *Round Robin* style computation with the number of rounds linear in the threshold. The ciphertext contains a non-interactive proof-of-knowledge using the efficient WaterSQI [Fou25] framework. Decapsulation, on the other hand, requires the interactive proof-of-knowledge from [MJ23], resulting in an overall computational cost that is quadratic in the participation threshold. Notably, both ciphertext and public key size are independent of the group size and only scale linearly in the security parameter. Security can be proven under minimal assumptions, featuring static corruptions and identifiable aborts. Due to the lack of existing security notions for Round Robin TPKE, we furthermore develop novel, (*restricted*) *chosen-ciphertext attack*-style security notions and compare them with the existing literature. For the security proof we rely on an idealized model for the group action.

In addition to the above construction, an alternative construction is feasible featuring a silent setup, security under adaptive corruptions, updatable public keys and constant round complexity for decryption. However, this comes at the expense of public key and ciphertext sizes linear in the number of parties. At the moment we do not plan to include this alternative construction in the package.

Scheme design for KeyGen and Secret Sharing. The kind of secret-sharing scheme used depends on the parameters. For our smallest set of parameters, the order of the group is known, allowing us to use standard Shamir secret-sharing as outlined in [DM20]. For larger parameters, where the order of the group is unknown, we instead use a black-box secret-sharing, as defined in [CF02]. Different instantiations of the secret-sharing [CF02; DDB95; DT06] are possible, and for each profile we consider the one providing better performance.

To instantiate the DKG, we implement (a generalization of) the protocol CSI-Rashi++ [ABCP23]. To achieve the identifiable aborts property, we propose an extended version of CSI-Rashi++ that also outputs the public verification shares. It achieves robustness and adaptive security with Honest-Majority in the QROM, assuming the hardness of **gaDLOG**.

The Group action implementation we specify follows the design of qt-Pegasis [PR23; Dar25; DEIV25], an improvement of Pegasis [DEFMI+25] and CSIDH [CLMPR18]. We shall introduce further improvements to the state of the art, in particular regarding isogeny computations.

Ad-hoc gadgets. As shown in [Fou25], under some specific parameter choices for our group action instantiations, it is possible to leverage additional algebraic structure, beyond the group action framework, to improve the performance of some subroutines of the schemes. When possible, we plan to include these optional speed-ups in our implementation, still allowing the possibility of removing them to just rely on the group action framework, if wanted.

Quantum security analysis. The security of most isogeny-based group actions, including qt-Pegasis, is parameterized by the characteristic of the chosen finite field. Their pre-quantum security strength is equivalent to that of the discrete logarithm in an elliptic curve over a finite field of half the size: thus qt-Pegasis over a field of 512 bits has comparable security to curve P-256 and AES-128.

Because of Kuperberg’s algorithm, there is much greater uncertainty surrounding the quantum security of commutative group actions in general. Estimates of the quantum security of 512-bits qt-Pegasis vary from “above 2^{80} quantum T -gates” [BLMP19, § 1.3] to “not much more than 2^{60} quantum T -gates” [Pei20, § 1.3].

Our specification document will contain a detailed cost analysis of classical and quantum attacks against our isogeny group action, taking into account the most recent algorithmic developments. Based on it, we will issue three parameter sets, dubbed *optimist*, *prudent* and *pessimist*, targeting quantum security at least as strong as AES-128 according to our costing models. For the sake of this preview, we provisionally set at 512, 1024 and 2048 bits the size of the finite fields corresponding to these parameters.

3. Open-Source Implementation

We propose a Python implementation of the threshold protocols and networking model, which calls efficient low-level pure-C implementations of the group action and other isogeny-related gadgets. We additionally include Intel-assembly-optimised implementations of the low-level routines.

Source code, build instructions and issue-tracking will be made available for early review at <https://github.com/Threshold-Group-Actions/>. The code depends on [GNU Multiple Precision Arithmetic Library](#) for multi-precision integers and ships code for \mathbb{F}_p -arithmetic machine-generated through modarith².

The networking model is implemented in Python in a group-action-agnostic way. This allows for agility in the implementation, including the ability to mock-up isogeny-based group actions with pre-quantum discrete logarithm groups.

The implementation ships unit-tests for all local operations, as well as protocol-level testing for the processing of valid, corrupted and malicious data packets. We ensure that inputs of the protocols are valid using efficient algorithms [BGS22]. A canonical representation of public data is used to aid in the development of detection of corrupted data and early-abort within the threshold protocols.

²<https://github.com/mcarrickscott/modarith>

4. Experimental Performance Evaluation

Performance of our schemes is impacted by two main factors: the number of group actions evaluated by the protocol, and the individual cost of a group action evaluation. For the latter, we report in the table below the runtimes of our preliminary C implementation of qt-Pegasis (median of 100 runs on a i7-11850H with turboboost disabled).

| Parameter set | <i>optimist</i> | <i>prudent</i> | <i>pessimist</i> |
|----------------------------------|-----------------|----------------|------------------|
| Size of p (in bits) | 512 | 1024 | 2048 |
| Estimated C timings (in MCycles) | 103.11 | 510.35 | 3633.87 |
| Estimated C timings (in ms) | 41 | 204 | 1454 |

To obtain an estimate for the overall runtime, we multiply the numbers above by the total number of group actions, although this ignores potential speed-ups achieved through parallelization and ad-hoc gadgets. The results are summarized in the table below.

| Size of p | 512 bits | | | 1024 bits | | | 2048 bits | | |
|--|----------|-------|--------|-----------|-------|--------|-----------|--------|---------|
| profile | (2,3) | (3,5) | (8,16) | (2,3) | (3,5) | (8,16) | (2,3) | (3,5) | (8,16) |
| Threshold Signing Preprocessing estimates for 3 pk size choices (in sec) | | | | | | | | | |
| Small | 6.03 | 10.04 | 30.14 | 29.99 | 49.98 | 149.94 | 213.74 | 356.23 | 1068.69 |
| Medium | 2.34 | 3.90 | 11.69 | 11.63 | 19.38 | 58.14 | 82.88 | 138.13 | 414.39 |
| Large | 1.35 | 2.25 | 6.76 | 6.73 | 11.22 | 33.66 | 47.98 | 79.97 | 239.91 |
| Threshold PKEstimates (in ms) | | | | | | | | | |
| Dec. | 123 | 246 | 1476 | 612 | 1224 | 7344 | 4362 | 8724 | 52344 |
| Enc. | 82 | | | 408 | | | 2908 | | |
| Distributed Key Generation estimates (in min) | | | | | | | | | |
| DKG | 0.4 | 0.5 | 1.4 | 1.7 | 2.6 | 7.0 | 12.5 | 18.7 | 49.8 |
| DKG IA | 1.4 | 3.2 | 23.9 | 17.5 | 55.0 | 678.1 | 124.6 | 392.4 | 4832.9 |

Macaw (category S1) offers a range of compromises between public key and signature size. These are shown in Figure 1. More precisely, a linear increase in the public keys size gives a logarithmic improvement on the signature size, preprocessing time and verification time (see [BPPRS25, §3.1]). We note the preprocessing time only weighs on the parties actively participating to the signing protocol, whose number we take equal to the threshold. The table shows the estimated preprocessing time, dominated by the latency of the sequential group action evaluations. The online phase time is instead dominated by the network latency of 2 online rounds of interaction.

Kea (category S2) features relatively fast encryption and small bandwidth. The number of sequential group actions performed during decryption scales as $T(T + 1)/2$, whereas encryption requires only two group action computations. The bandwidth estimates are contained in Figure 1, the ciphertext includes a WaterSQL proof and its size is computed using the values in [Fou25].

For Kakapo (category S4) we report the cost of generating one public key. This needs to be repeated multiple times (from 1 to hundreds, depending on the chosen parameters) if generating

public keys for Macaw. Kakapo requires additional computations to also generate public verification shares for the identifiable aborts protocol. We are currently working on reducing the overhead of these additional computations.

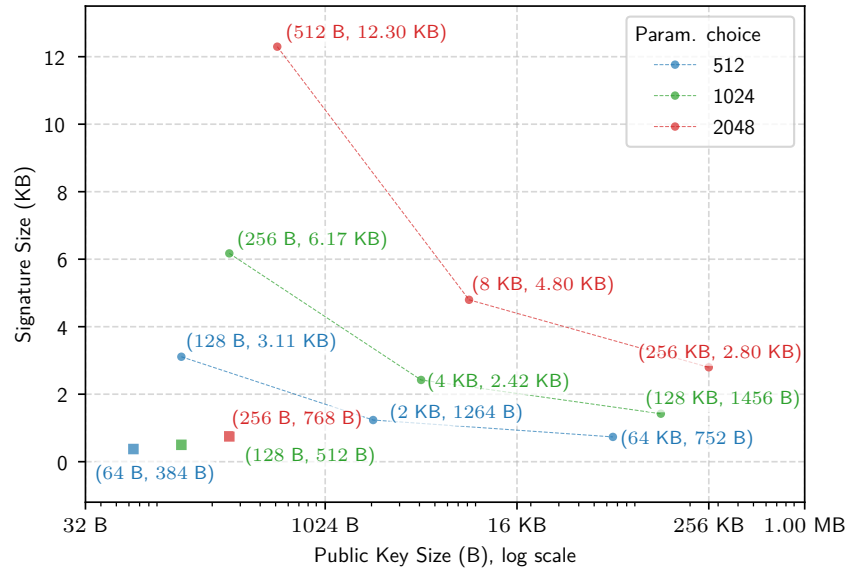


Figure 1: Estimated trade-offs in size between public keys and signatures for Macaw (circles) and between public keys and ciphertexts for Kea (squares).

5. Licensing, Patent Claims, and Funding

All source code will be released under open-sources licenses compatible with those of eventual bundled dependencies. We are not aware of any patent claims covering the contents of the submission.

We acknowledge sponsoring by Cryptonls, [SNSF Consolidator Grant 213766](#); Danish Independent Research Council under project number 1026-00350B (RENAIS); ERC European Union’s Horizon 2020 research and innovation programme (grant agreement ISOCRYPT – No. 101020788); Council KU Leuven grant C14/24/099; CyberSecurity Research Flanders, reference number VOEWICS02; FWO PhD Fellowship fundamental research project n. 1138925N; grant PID2022-142290OB-I00, funded by MCIN/AEI/10.13039/501100011033/ FEDER, UE; grant JDC2022-049711-I, funded by MCIN/AEI/10.13039/501100011033 and the European Union «NextGenerationEU/PRTR»; PICOCRYPT project under the European Union’s Horizon 2020 research and innovation programme (Grant agreement No. 101001283); Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy – EXC 2092 CASA – 390781972; Research grant VIL53029 from VILLUM FONDEN Natural Sciences and Engineering Research Council (Canada) Discovery Grant RGPIN-2024-03996; France 2030 program under grants ANR-22-PETQ-0007 (EPiQ) and ANR-22-PETQ-0008 (PQ-TLS); BPI France project RESQUE; EPSRC grant number EP/V011324/1; FRIA grant by the National Fund for Scientific Research (F.N.R.S.) of Belgium.

References

- [AAABC+24] Marius A. Aardal, Gora Adj, Diego F. Aranha, Andrea Basso, Isaac Andrés Canales Martínez, Jorge Chávez-Saab, Maria Corte-Real Santos, Pierrick Dartois, Luca De Feo, Max Duparc, Jonathan Komada Eriksen, Tako Boris Fouotsa, Décio Luiz Gazzoni Filho, Basil Hess, David Kohel, Antonin Leroux, Patrick Longa, Luciano Maino, Michael Meyer, Kohei Nakagawa, Hiroshi Onuki, Lorenz Panny, Sikhar Patranabis, Christophe Petit, Giacomo Pope, Krijn Reijnders, Damien Robert, Francisco Rodríguez Henríquez, Sina Schaeffler, and Benjamin Wesolowski. *SQIsign*. Tech. rep. available at <https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures>. National Institute of Standards and Technology, 2024.
- [ABCP23] Shahla Atapoor, Karim Baghery, Daniele Cozzo, and Robi Pedersen. “VSS from Distributed ZK Proofs and Applications”. In: *Advances in Cryptology – ASIACRYPT 2023, Part I*. Ed. by Jian Guo and Ron Steinfeld. Vol. 14438. Lecture Notes in Computer Science. Guangzhou, China: Springer, Singapore, Singapore, December 2023, pp. 405–440. DOI: [10.1007/978-981-99-8721-4_13](https://doi.org/10.1007/978-981-99-8721-4_13). Also at ia.cr/2023/992.
- [ADMP20] Navid Alamati, Luca De Feo, Hart Montgomery, and Sikhar Patranabis. “Cryptographic Group Actions and Applications”. In: *Advances in Cryptology – ASIACRYPT 2020, Part II*. Ed. by Shiho Moriai and Huaxiong Wang. Vol. 12492. Lecture Notes in Computer Science. Daejeon, South Korea: Springer, Cham, Switzerland, December 2020, pp. 411–439. DOI: [10.1007/978-3-030-64834-3_14](https://doi.org/10.1007/978-3-030-64834-3_14). Also at ia.cr/2020/1188.
- [BGS22] Gustavo Banegas, Valerie Gilchrist, and Benjamin Smith. “Efficient supersingularity testing over $GF(p)$ and CSIDH key validation”. In: *Transactions on Mathematical Cryptology* 2.1 (October 2022), pp. 21–35. URL: <https://journals.flvc.org/mathcryptology/article/view/132125>.
- [BGZ23] Dan Boneh, Jiaxin Guan, and Mark Zhandry. “A Lower Bound on the Length of Signatures Based on Group Actions and Generic Isogenies”. In: *Advances in Cryptology – EUROCRYPT 2023, Part V*. Ed. by Carmit Hazay and Martijn Stam. Vol. 14008. Lecture Notes in Computer Science. Lyon, France: Springer, Cham, Switzerland, April 2023, pp. 507–531. DOI: [10.1007/978-3-031-30589-4_18](https://doi.org/10.1007/978-3-031-30589-4_18). Also at ia.cr/2023/250.
- [BKV19] Ward Beullens, Thorsten Kleinjung, and Frederik Vercauteren. “CSI-FiSh: Efficient Isogeny Based Signatures Through Class Group Computations”. In: *Advances in Cryptology – ASIACRYPT 2019, Part I*. Ed. by Steven D. Galbraith and Shiho Moriai. Vol. 11921. Lecture Notes in Computer Science. Kobe, Japan: Springer, Cham, Switzerland, December 2019, pp. 227–247. DOI: [10.1007/978-3-030-34578-5_9](https://doi.org/10.1007/978-3-030-34578-5_9). Also at ia.cr/2019/498.
- [BLMP19] Daniel J. Bernstein, Tanja Lange, Chloe Martindale, and Lorenz Panny. “Quantum Circuits for the CSIDH: Optimizing Quantum Evaluation of Isogenies”. In: *Advances in Cryptology – EUROCRYPT 2019, Part II*. Ed. by Yuval Ishai and Vincent Rijmen.

- Vol. 11477. Lecture Notes in Computer Science. Darmstadt, Germany: Springer, Cham, Switzerland, May 2019, pp. 409–441. DOI: [10.1007/978-3-030-17656-3_15](https://doi.org/10.1007/978-3-030-17656-3_15). Also at ia.cr/2018/1059.
- [BPPRS25] Giacomo Borin, Edoardo Persichetti, Federico Pintore, Krijn Reijnders, and Paolo Santini. “A Guide to the Design of Digital Signatures based on Cryptographic Group Actions”. In: *J. Cryptol.* 38.3 (2025), p. 23. DOI: [10.1007/S00145-025-09542-9](https://doi.org/10.1007/S00145-025-09542-9). Also at ia.cr/2023/718.
- [BS20] Xavier Bonnetain and André Schrottenloher. “Quantum Security Analysis of CSIDH”. In: *Advances in Cryptology – EUROCRYPT 2020, Part II*. Ed. by Anne Canteaut and Yuval Ishai. Vol. 12106. Lecture Notes in Computer Science. Zagreb, Croatia: Springer, Cham, Switzerland, May 2020, pp. 493–522. DOI: [10.1007/978-3-030-45724-2_17](https://doi.org/10.1007/978-3-030-45724-2_17). Also at ia.cr/2018/537.
- [CF02] Ronald Cramer and Serge Fehr. “Optimal Black-Box Secret Sharing over Arbitrary Abelian Groups”. In: *Advances in Cryptology – CRYPTO 2002*. Ed. by Moti Yung. Vol. 2442. Lecture Notes in Computer Science. Santa Barbara, CA, USA: Springer Berlin Heidelberg, Germany, August 2002, pp. 272–287. DOI: [10.1007/3-540-45708-9_18](https://doi.org/10.1007/3-540-45708-9_18). Also at ia.cr/2002/036.
- [CLMPR18] Wouter Castryck, Tanja Lange, Chloe Martindale, Lorenz Panny, and Joost Renes. “CSIDH: An Efficient Post-Quantum Commutative Group Action”. In: *Advances in Cryptology – ASIACRYPT 2018, Part III*. Ed. by Thomas Peyrin and Steven Galbraith. Vol. 11274. Lecture Notes in Computer Science. Brisbane, Queensland, Australia: Springer, Cham, Switzerland, December 2018, pp. 395–427. DOI: [10.1007/978-3-030-03332-3_15](https://doi.org/10.1007/978-3-030-03332-3_15). Also at ia.cr/2018/383.
- [Cou06] Jean-Marc Couveignes. *Hard Homogeneous Spaces*. Cryptology ePrint Archive, Report 2006/291. 2006. URL: <https://eprint.iacr.org/2006/291>.
- [Dar25] Pierrick Dartois. “Fast computation of 2-isogenies in dimension 4 and cryptographic applications”. In: *Journal of Algebra* 683 (2025), pp. 449–514. DOI: [10.1016/j.jalgebra.2025.06.033](https://doi.org/10.1016/j.jalgebra.2025.06.033). URL: <https://www.sciencedirect.com/science/article/pii/S0021869325003771>.
- [DDB95] Yvo Desmedt, Giovanni Di Crescenzo, and Mike Burmester. “Multiplicative Non-abelian Sharing Schemes and their Application to Threshold Cryptography”. In: *Advances in Cryptology – ASIACRYPT’94*. Ed. by Josef Pieprzyk and Reihaneh Safavi-Naini. Vol. 917. Lecture Notes in Computer Science. Wollongong, Australia: Springer Berlin Heidelberg, Germany, November 1995, pp. 21–32. DOI: [10.1007/BFb0000421](https://doi.org/10.1007/BFb0000421).
- [DEFMI+25] Pierrick Dartois, Jonathan Komada Eriksen, Tako Boris Fouotsa, Arthur Herlédan Le Merdy, Riccardo Invernizzi, Damien Robert, Ryan Rueger, Frederik Vercauteren, and Benjamin Wesolowski. “PEGASIS: Practical Effective Class Group Action using 4-Dimensional Isogenies”. In: *Advances in Cryptology – CRYPTO 2025, Part I*. Ed. by

- Yael Tauman Kalai and Seny F. Kamara. Vol. 16000. Lecture Notes in Computer Science. Santa Barbara, CA, USA: Springer, Cham, Switzerland, August 2025, pp. 67–99. DOI: [10.1007/978-3-032-01855-7_3](https://doi.org/10.1007/978-3-032-01855-7_3). Also at ia.cr/2025/401.
- [DEIV25] Pierrick Dartois, Jonathan Komada Eriksen, Riccardo Invernizzi, and Frederik Vercauteren. *qt-Pegasis: Simpler and Faster Effective Class Group Actions*. Cryptology ePrint Archive, Paper 2025/1859. 2025. URL: <https://eprint.iacr.org/2025/1859>.
- [DHKKLR23] Julien Duman, Dominik Hartmann, Eike Kiltz, Sabrina Kunzweiler, Jonas Lehmann, and Doreen Riepel. “Generic Models for Group Actions”. In: *PKC 2023: 26th International Conference on Theory and Practice of Public Key Cryptography, Part I*. Ed. by Alexandra Boldyreva and Vladimir Kolesnikov. Vol. 13940. Lecture Notes in Computer Science. Atlanta, GA, USA: Springer, Cham, Switzerland, May 2023, pp. 406–435. DOI: [10.1007/978-3-031-31368-4_15](https://doi.org/10.1007/978-3-031-31368-4_15). Also at ia.cr/2023/186.
- [DM20] Luca De Feo and Michael Meyer. “Threshold Schemes from Isogeny Assumptions”. In: *PKC 2020: 23rd International Conference on Theory and Practice of Public Key Cryptography, Part II*. Ed. by Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas. Vol. 12111. Lecture Notes in Computer Science. Edinburgh, UK: Springer, Cham, Switzerland, May 2020, pp. 187–212. DOI: [10.1007/978-3-030-45388-6_7](https://doi.org/10.1007/978-3-030-45388-6_7). Also at ia.cr/2019/1288.
- [DT06] Ivan Damgård and Rune Thorbek. “Linear Integer Secret Sharing and Distributed Exponentiation”. In: *PKC 2006: 9th International Conference on Theory and Practice of Public Key Cryptography*. Ed. by Moti Yung, Yevgeniy Dodis, Aggelos Kiayias, and Tal Malkin. Vol. 3958. Lecture Notes in Computer Science. New York, NY, USA: Springer Berlin Heidelberg, Germany, April 2006, pp. 75–90. DOI: [10.1007/11745853_6](https://doi.org/10.1007/11745853_6). Also at ia.cr/2006/044.
- [Fou25] Tako Boris Fouotsa. *WaterSQI and PRISMO: Quaternion Signatures for Supersingular Isogeny Group Actions*. Cryptology ePrint Archive, Paper 2025/1737. 2025. URL: <https://eprint.iacr.org/2025/1737>.
- [Kup05] Greg Kuperberg. “A subexponential-time quantum algorithm for the dihedral hidden subgroup problem”. In: *SIAM Journal on Computing* 35.1 (2005), pp. 170–188. DOI: <https://doi.org/10.1137/S0097539703436345>. Also at [arXiv:quant-ph/0302112](https://arxiv.org/abs/quant-ph/0302112).
- [Kup13] Greg Kuperberg. “Another Subexponential-time Quantum Algorithm for the Dihedral Hidden Subgroup Problem”. In: *8th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2013)*. Ed. by Simone Severini and Fernando Brandao. Vol. 22. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2013, pp. 20–34. DOI: [10.4230/LIPIcs.TQC.2013.20](https://doi.org/10.4230/LIPIcs.TQC.2013.20).
- [MJ23] Youcef Mokrani and David Jao. “Generating Supersingular Elliptic Curves over \mathbb{F}_p with Unknown Endomorphism Ring”. In: *Progress in Cryptology - INDOCRYPT 2023: 24th*

- International Conference in Cryptology in India, Part I*. Ed. by Anupam Chattopadhyay, Shivam Bhasin, Stjepan Picek, and Chester Rebeiro. Vol. 14459. Lecture Notes in Computer Science. Goa, India: Springer, Cham, Switzerland, December 2023, pp. 159–174. DOI: [10.1007/978-3-031-56232-7_8](https://doi.org/10.1007/978-3-031-56232-7_8). Also at ia.cr/2023/984.
- [MZ22] Hart Montgomery and Mark Zhandry. “Full Quantum Equivalence of Group Action DLog and CDH, and More”. In: *Advances in Cryptology – ASIACRYPT 2022, Part I*. Ed. by Shweta Agrawal and Dongdai Lin. Vol. 13791. Lecture Notes in Computer Science. Taipei, Taiwan: Springer, Cham, Switzerland, December 2022, pp. 3–32. DOI: [10.1007/978-3-031-22963-3_1](https://doi.org/10.1007/978-3-031-22963-3_1). Also at ia.cr/2022/1135.
- [Pei20] Chris Peikert. “He Gives C-Sieves on the CSIDH”. In: *Advances in Cryptology – EUROCRYPT 2020, Part II*. Ed. by Anne Canteaut and Yuval Ishai. Vol. 12106. Lecture Notes in Computer Science. Zagreb, Croatia: Springer, Cham, Switzerland, May 2020, pp. 463–492. DOI: [10.1007/978-3-030-45724-2_16](https://doi.org/10.1007/978-3-030-45724-2_16). Also at ia.cr/2019/725.
- [PKNRT25] Rafaël Del Pino, Shuichi Katsumata, Guilhem Niot, Michael Reichle, and Kaoru Takemure. “Unmasking TRaccoon: A Lattice-Based Threshold Signature with An Efficient Identifiable Abort Protocol”. In: *Advances in Cryptology – CRYPTO 2025, Part VI*. Ed. by Yael Tauman Kalai and Seny F. Kamara. Vol. 16005. Lecture Notes in Computer Science. Santa Barbara, CA, USA: Springer, Cham, Switzerland, August 2025, pp. 423–456. DOI: [10.1007/978-3-032-01887-8_14](https://doi.org/10.1007/978-3-032-01887-8_14). Also at ia.cr/2025/849.
- [PR23] Aurel Page and Damien Robert. *Introducing Clapoti(s): Evaluating the isogeny class group action in polynomial time*. Cryptology ePrint Archive, Report 2023/1766. 2023. URL: <https://eprint.iacr.org/2023/1766>.
- [Reg04] Oded Regev. *A Subexponential Time Algorithm for the Dihedral Hidden Subgroup Problem with Polynomial Space*. 2004. arXiv: [quant-ph/0406151](https://arxiv.org/abs/quant-ph/0406151) [[quant-ph](https://arxiv.org/abs/quant-ph/0406151)]. URL: <https://arxiv.org/abs/quant-ph/0406151>.
- [NIST-IR8214C] Luís T. A. N. Brandão and René Peralta. *NIST First Call for Multi-Party Threshold Schemes*. (National Institute of Standards and Technology) NIST Internal Report (NISTIR) 8214C. 2026. DOI: [10.6028/NIST.IR.8214C](https://doi.org/10.6028/NIST.IR.8214C).