

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

Title: ATMACA (A Threshold Multiparty Algorithm for Compact Authentication)

Subtitle: Lattice-Based Threshold Signing for FALCON

Version: Preview Writeup 1.0 (2026-09-30)¹

Team name: ATMACA Team

Team members: Erdem Alkım, Metin Bektaş, Erem Ersoy, Muhammed Said Gündoğan

Abstract: Fast-Fourier Lattice-based Compact Signatures over NTRU (FALCON) is the basis of the planned Fast-Fourier-Transform-over-NTRU-Lattice-Based Digital Signature Algorithm (FN-DSA) standard of NIST. ATMACA is a threshold signing protocol for FALCON: the signing key is secret-shared among n parties, and any t of them sign jointly by multi-party computation (MPC) without reconstructing the key. FALCON’s fast Fourier sampler draws $2d$ Gaussian samples in sequence, so its MPC evaluation needs a number of online rounds linear in the ring degree d . ATMACA instead evaluates the hybrid sampler of Ducas and Prest, in which only two nearest-plane steps are sequential and the samples within each step are drawn in parallel. Keys from annular NTRU trapdoor generation (ANTRAG) keep these signatures within FALCON’s verification bound, so the unchanged FALCON verifier accepts them. We estimate fewer than 180 online rounds per signing attempt in expectation for FALCON-512 and FALCON-1024, and fewer than 350 in the worst case under our iteration cap. These round counts are more than 30 times lower than those reported for prior threshold FALCON signing.

We target Category N1, at the NIST security categories 1 and 5, against a dishonest majority, with profiles $n2$, $n3fD$, $nSfD$, $nMfD$, and $nLfD$.

Proposed crypto-systems: ATMACA: Category N1.

Keywords: FALCON, Threshold Cryptography, Threshold Signatures, NIST Threshold Call, Multi-Party Computation, Lattice Cryptography, Discrete Gaussian Sampling



Logo AI-Generated with Grok Imagine (xAI’s Aurora model).

¹Version submitted to NIST-MPTC for publication; revises version 0.1 of 2026-08-07

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: Erdem Alkım^{i1,a1}, Metin Bektaş^{i2,a2}, Erem Ersoy^{i3,a2}, Muhammed Said Gündoğan^{i4,a2}

Open Researcher and Contributor Identifiers (ORCID):

i1 (0000-0003-4638-2422); i2 (0009-0005-9538-6480); i3 (0009-0005-6798-5742); i4 (0000-0001-9450-758X)

Affiliations:

^{a1} Department of Computer Science, Dokuz Eylül University @ İzmir, Türkiye

^{a2} National Research Institute of Electronics and Cryptology (UEKAE), TÜBİTAK BİLGEM @ Gebze, Kocaeli, Türkiye

Main contacts:

- **Team mailing list:** atmaca@googlegroups.com
- **Primary technical contact person:** Erem Ersoy, erem.ersoy@tubitak.gov.tr
- **Secondary contact person:** Metin Bektaş, bektasmetin42@gmail.com

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

Conventional FALCON signing. FALCON [FHKLP+20] is a hash-and-sign signature scheme over NTRU lattices, selected by NIST as the basis of FN-DSA, whose Federal Information Processing Standard (FIPS) 206 is still in development [Nat26]. A signature is a short vector in a coset determined by the salted hash of the message. The signer finds it with its secret NTRU basis and the fast Fourier sampler `ffSampling`, a randomized nearest-plane algorithm that draws $2d$ integer Gaussian samples in sequence: the center of each sample depends on the samples drawn before it.

Threshold setting. In a threshold scheme, the signing key is secret-shared among n parties, and any t of them compute a signature by secure multi-party computation (MPC) without reconstructing the key. Linear operations on shares are local, but each nonlinear step, such as a comparison inside Gaussian sampling, requires the parties to exchange messages. The dependency chain of `ffSampling` thus becomes a chain of communication rounds whose length grows linearly with d : the threshold FALCON protocol of Garg et al. [GEPTW26] reports about 6,189 opening calls per FALCON-512 signature (10,179 for FALCON-1024) in its dishonest-majority evaluation.

This submission. ATMACA is a threshold signing protocol for FALCON that shortens this chain by changing the sampler. The signers hold authenticated shares of an NTRU trapdoor and jointly evaluate the hybrid sampler of Ducas and Prest [Pre15], in the form used by Mitaka [EFGRTTWY22]. Only its two ring-level nearest-plane steps are sequential; within each step, the d scalar Gaussian samples have independent centers and are drawn in parallel. Mitaka adopted this sampler because it is simpler and easier to mask, but it needed its own parameters, so its signatures do not pass an unchanged FALCON verifier. Annular NTRU trapdoor generation (ANTRAG) [ENSTW23] closes most of this gap: its keys have hybrid-sampler quality $\alpha = 1.15$ at $d = 512$ and 1.23 at $d = 1024$, close to FALCON's 1.17 , so that the signatures can be kept within FALCON's verification bound. In ATMACA, a trusted dealer runs ANTRAG key generation once, publishes h , and gives each party authenticated shares of the trapdoor and of the sampler's preparation data. ATMACA's public keys and signatures are ordinary FALCON v1.2 keys and signatures, which the unchanged verifier accepts.

Expected interaction. Each sequential step waits for the slowest of its d parallel rejection samplers; with four trials per unfinished coefficient in each iteration, this takes about 1.8 iterations at $d = 512$. We estimate fewer than 180 online rounds per signing attempt in expectation for both parameter sets, and at most 350 under an iteration cap (Section 4), more than 30 times fewer than the count above, although the two counts are not measured the same way.

Scope. We submit in Category N1, which covers FN-DSA once it is standardized [NIST-IR8214C]; until FIPS 206 is published, our target is FALCON v1.2. We aim at NIST security categories 1 ($d = 512$) and 5 ($d = 1024$), for the dishonest-majority threshold profiles $n2$, $n3fD$, $nSfD$, $nMfD$ and $nLfD$. The package will contain a Technical Specification, an open-source Reference Implementation, a Report on Experimental Evaluation and Notes on Patent Claims.

2. Specification

2.1. Organization

The Technical Specification will define the parameters, the dealer's key initialization, the MPC building blocks, the samplers and the signing protocol, followed by the security analysis and the cost accounting.

Key initialization. The dealer runs ANTRAG key generation, checks the NTRU and smoothing conditions, and forms FALCON's basis and public key $h = gf^{-1} \bmod q$. It also prepares the data of the hybrid sampler: for each nearest-plane step $i \in \{1, 2\}$, a projection vector β_i and a ring-valued perturbation multiplier σ_i , and the Gram–Schmidt coefficient $\bar{\mu}$ of the size-reduced basis.

Signing. A signing session produces one candidate signature:

1. The parties open a 320-bit salt ρ , generated and authenticated before the message was known, and compute the target c° from FALCON's hash-to-point of (ρ, M) .
2. For $i = 2$ and then $i = 1$, they compute the shared center of step i , subtract a perturbation $y_i = \sigma_i u_i$ prepared offline from a fixed-point normal sample u_i , and call the ring sampler Π_{RingDG} . It runs d one-dimensional samplers in parallel; each uses half-Gaussian proposals drawn offline from a reverse cumulative distribution table (CDT) and accepts them by Bernoulli-exponential rejection. Each iteration runs $m_{\text{tr}} = 4$ trials per unfinished coefficient and opens only the accept bits. The sampled ring element, multiplied by the shared basis vectors, is subtracted from the target.
3. The parties open the candidate s_2 and recompute $s_1 = c^\circ - s_2 h \bmod q$ in public. They output the FALCON signature (ρ, s_2) if $\|(s_1, s_2)\|^2 \leq \lfloor \beta^2 \rfloor$ and s_2 compresses; otherwise the session ends without output, and signing restarts in a new session.

Fixed-point arithmetic. Signing uses no floating-point arithmetic. Each secret non-integer value x is represented as $\lfloor 2^\kappa x \rfloor$, with $\kappa = 55$ fractional bits for both parameter sets, in a prime field large enough that no intermediate value wraps around. The reverse-CDT and Bernoulli-exponential tables are public integers of $\kappa' = 110$ bits, computed once per parameter set. Key generation runs outside the MPC and may use any sufficiently precise arithmetic.

Sampler. ATMACA uses only the power-of-two cyclotomic rings $\mathcal{R} = \mathbb{Z}[X]/(X^d + 1)$ with $d \in \{512, 1024\}$, and follows the hybrid sampler as specified for Mitaka, which also gives a corrected analysis of it [\[EFGRTTWY22\]](#).

2.2. System Model

Parties and thresholds. The primary profile has $n \geq 2$ registered parties and participation threshold t , with $2 \leq t \leq n$: any signing set $S \subseteq [n]$ with $|S| \geq t$ is authorized. A static, active adversary corrupts $f < t$ parties before signing starts and may make them deviate arbitrarily.

Setup. A trusted dealer runs key initialization once per key. It Shamir-shares, with degree $t - 1$, every coefficient of the trapdoor basis and of the preparation data $(\beta_1, \beta_2, \bar{\mu}, \sigma_1, \sigma_2)$, each with its message authentication code (MAC) value, shares the global MAC keys, and then erases all secret data. The dealer distributes nothing else: the correlated randomness that signing consumes is generated by the parties of each signing set before the message is known (Section 2.4).

Communication topology. Every opening can be a direct all-to-all exchange among the active parties, which our cost figures assume. Alternatively, a participant or an untrusted relay can act as the hub of a star: each party sends its share of an opened value to the hub, which adds the shares and returns only the sum, so each party sends and receives one element per opened value instead of $|S| - 1$, at the price of two message delays per round. The hub learns nothing that an all-to-all participant does not, and it cannot change an opened value undetected, because every party checks the MACs of the result.

Network. Unforgeability and key secrecy do not depend on timing: they hold even if the adversary delays, drops, duplicates or reorders messages, as in an asynchronous network. A session finishes only if messages between honest parties are eventually delivered; parties abort sessions that exceed a timeout, so corrupt parties can stall a session.

Output guarantees. ATMACA is secure with abort: a corrupt party can make a session abort, and it may learn the signature while the honest parties abort. Fairness and guaranteed output delivery are impossible in general without an honest majority [Cle86], which our profiles do not assume. Identifiable abort is achievable against a dishonest majority [IOZ14; BOS16], but it needs a broadcast channel, per-party authentication of every share and identifiable generation of the correlated randomness, which is the expensive part [BOS16]; we keep it as an option for the package. Crashed or unreachable parties do not block signing, since any t registered parties can form a new signing set once its preprocessing is available. Against actively misbehaving parties we make no availability claim.

2.3. Security

ATMACA keeps FALCON's ring, modulus $q = 12289$, hash-to-point function, encoding and compression, and it outputs a candidate only if it passes FALCON's verification test. A forgery against ATMACA must therefore satisfy the same verification equation as a FALCON forgery. We claim that ATMACA is unforgeable against an adversary that corrupts up to $t - 1$ signers and obtains signatures from the honest signers on adaptively chosen messages. The claim rests on the security of FALCON and of the MPC layer. Because ATMACA changes FALCON's key generator and sampler, the package will also argue security categories 1 and 5 from the hardness of ANTRAG keys [ENSTW23] and from the closeness of the sampler's fixed-point output to the ideal lattice Gaussian.

Detected deviations. Every opening is MAC-checked, in batches.

- The salt, the accept bits and s_2 are used only after a passing check.

- After a failed check, or a missing or malformed message, the honest parties abort the session, erase its state and discard its preprocessing.
- Without broadcast, honest parties may disagree on whether a session aborted, but not on its signature, which is publicly verifiable and can be forwarded.

Rushing adversaries. A rushing adversary sees the honest parties' messages in each round before sending its own, but without aborting it cannot bias the signatures. Every random value used online (the salt, the perturbations and the proposal records) is generated and authenticated before the message is known, the salt is opened only after the message is fixed, and the batch-check coefficients are derived only after the batch is fixed. Changing a share after seeing the honest shares changes an authenticated value, which is detected except with probability at most about 2^{-135} in total. The remaining choice, to abort after seeing opened values, is permitted by security with abort; the deployment bound Q_{sig} counts every session, including aborted ones.

2.4. Realization of the MPC Layer

Authenticated sharing. Integer, fixed-point and ring values are additively shared over $\mathbb{F}_p$, with SPDZ-style MACs in $\mathbb{F}_{p^2}$ [DPSZ12; KOS16]; Boolean values are XOR-shared. A signing set S turns the dealer's Shamir shares into additive shares by multiplying them with its Lagrange coefficients, without communication. Linear operations are local; products use Beaver triples, ANDs use Boolean triples, and domain conversions use daBits and edaBits [RW19; EGKRS20].

Composition. The secure floor and the ties-to-even rounding use bit decomposition in the style of Nishide and Ohta [NO07], with canonical edaBits and a carry-lookahead adder. A logarithmic-depth comparator against public thresholds implements the reverse-CDT proposal and prepares the Bernoulli factors, and the Bernoulli-exponential step ANDs the factors selected by the bits of the rejection exponent. The d samplers of a ring call run in one synchronized loop, so the round count depends on d only through the number of iterations, which grows logarithmically.

Offline and online phases. All message-independent work is done in advance: the fixed-point normal samples, the two perturbations, the proposal records and all correlated randomness, which is generated from oblivious transfer (OT), with MASCOT [KOS16] for the arithmetic triples and its binary counterpart [FKOS15] for the Boolean ones. The offline phase ends with a MAC check. Online, the parties open the salt, compute the two centers, run the two ring samplers and open s_2 ; the norm test and the compression are public.

3. Open-Source Implementation

The reference implementation will be written in C and published in a public Git repository under the MIT license (Section 5). The main modules will be field and fixed-point arithmetic, authenticated sharing with batched MAC checks, the preprocessing protocols, the samplers and the signing protocol. Each party runs as a separate process, and parties communicate all-to-all or through a relay. Every signature will be checked with the unmodified FALCON reference verifier. The parts

shared with FALCON (hash-to-point, encoding and verification) will be tested against FALCON's known-answer tests. Because ATMACA's sampler differs from FALCON's, its signatures cannot reproduce FALCON's signing vectors, so the signing tests will derive all randomness from fixed seeds and compare transcript hashes. Test sessions will also include a corrupted party that sends inconsistent shares or withholds messages, and dropped or delayed messages; each must end in a valid signature or an abort.

4. Experimental Performance Evaluation

Platform and metrics. We will measure on the NIST baseline platform: sixteen x86-64 cores, 64 GB of memory, two terabytes of storage, and Ubuntu Desktop 24.04 or 26.04 long-term support (LTS). For all-honest runs and runs with a corrupted party, we will report per phase and per party the computation time, memory, bytes sent, messages and rounds, and the elapsed time.

Communication. Table 1 gives our estimates for one signing attempt, per ordered pair of active parties: the online communication of a FALCON-512 attempt ($\kappa = 55$, $\kappa' = 110$, 120-bit prime, four trials per iteration) and upper estimates of the online rounds for both parameter sets. For $|S|$ active parties, the total online communication is $|S|(|S| - 1)$ times the per-pair figure; a three-party session moves about 18 MB online. The figures exclude offline preprocessing, which also runs between pairs of parties and grows the same way. An attempt yields a signature with probability at least 0.999.

Table 1: Estimated cost of one signing attempt, per ordered pair of parties.

Quantity	Estimate
Online communication (FALCON-512)	about 3 MB
Online rounds, upper estimates: minimum / expected / cap	110 / 180 / 350

Rounds. A trial accepts with probability $p_{\text{DG}} \approx 0.77$, so a coefficient is still unfinished after an iteration with probability $(1 - p_{\text{DG}})^4 \approx 2.8 \cdot 10^{-3}$, and a ring call on 512 coefficients needs about 1.77 iterations in expectation. We cap the two ring calls of an attempt at nine iterations in total, which keeps every attempt below 350 rounds for both parameter sets; the cap is exceeded with probability below 2^{-45} , and the session then ends without a signature. The precisions enter these depths only logarithmically, through the field size and κ' . Garg et al. report about 6,189 opening calls per FALCON-512 signature [GEPTW26], more than 30 times our expected depth; their figure is an implementation counter averaged over a batch, so the ratio is indicative.

5. Licensing, Patent Claims, and Funding

The reference implementation will be released under the permissive MIT license. We have no funding, and we make no patent claims on these techniques. It is our intent that they be available for universal use.

6. Comparison with Other Submissions

Among the earlier preview writeups, Mithril, Quorus and TALUS thresholdize the Module-Lattice-Based Digital Signature Algorithm (ML-DSA, Category N1), and Hermine and Tanuki build on Raccoon-style signatures (Category S1); Lemur [Lemur-PW] and LoTRS [LoTRS-PW], also lattice-based, target multi-signatures and threshold ring signatures. Garg et al. [GEPTW26] give the closest threshold FALCON work we know of, as a paper rather than a preview. Table 2 summarizes the previews as they describe themselves (DKG: distributed key generation; TEE: trusted execution environment).

Table 2: Lattice-based threshold signature previews, as described in their writeups.

Scheme	Verifier	Corruptions and size	Online rounds	Key setup
Mithril [Mithril-PW]	ML-DSA (FIPS 204)	dishonest majority; up to about 6–8 parties	3 per attempt, about 6 on average	dealer, DKG, or sharing an existing key
Quorus [Quorus-PW]	ML-DSA (FIPS 204)	honest majority; up to about 64 signers	not stated	DKG
TALUS [TALUS-PW]	ML-DSA (FIPS 204)	honest majority, or a TEE coordinator	1 broadcast round	DKG (planned)
Hermine [Hermine-PW]	Raccoon-like	dishonest majority; up to 64 signers	1 after preprocessing	DKG, proactive refresh
Tanuki [Tanuki-PW]	Raccoon-like	dishonest majority; up to 1024 signers	1 after preprocessing	trusted dealer
ATMACA	FALCON v1.2	dishonest majority, up to 1024 parties	under 180 expected, at most 350	trusted dealer

Aborts. Hermine identifies misbehaving signers without extra interaction, and TALUS adds a blame round on failure. Mithril and ATMACA do not identify them; Quorus is secure with abort in the universal composability (UC) framework, and Tanuki leaves identifiable aborts to later analysis.

Positioning. ATMACA is the only scheme in Table 2 that targets FN-DSA. Its outputs are standard FALCON signatures, checked by the unchanged verifier, at the price of more rounds and more offline preprocessing than the other schemes.

References

- [BOS16]** Carsten Baum, Emmanuela Orsini, and Peter Scholl. “Efficient Secure Multiparty Computation with Identifiable Abort”. In: *Theory of Cryptography—TCC 2016-B, Part I*. Vol. 9985. Lecture Notes in Computer Science. Springer, 2016. DOI: [10.1007/978-3-662-53641-4_18](https://doi.org/10.1007/978-3-662-53641-4_18).
- [Cle86]** Richard Cleve. “Limits on the Security of Coin Flips When Half the Processors Are Faulty”. In: *Proceedings of the 18th Annual ACM Symposium on Theory of Computing (STOC 1986)*. ACM, 1986, pp. 364–369.
- [DPSZ12]** Ivan Damgård, Valerio Pastro, Nigel P. Smart, and Sarah Zakarias. “Multiparty Computation from Somewhat Homomorphic Encryption”. In: *Advances in Cryptology—CRYPTO 2012*. Vol. 7417. Lecture Notes in Computer Science. Springer, 2012, pp. 643–662. Also at ia.cr/2011/535.
- [EFGRTTWY22]** Thomas Espitau, Pierre-Alain Fouque, François Gérard, Mélissa Rossi, Akira Takahashi, Mehdi Tibouchi, Alexandre Wallet, and Yang Yu. “MITAKA: A Simpler, Parallelizable, Maskable Variant of FALCON”. In: *Advances in Cryptology—EUROCRYPT 2022, Part III*. Ed. by Orr Dunkelman and Stefan Dziembowski. Vol. 13277. Lecture Notes in Computer Science. Cham: Springer, 2022, pp. 222–253. DOI: [10.1007/978-3-031-07082-2_9](https://doi.org/10.1007/978-3-031-07082-2_9). Also at ia.cr/2021/1486.
- [EGKRS20]** Daniel Escudero, Satrajit Ghosh, Marcel Keller, Rahul Rachuri, and Peter Scholl. “Improved Primitives for MPC over Mixed Arithmetic-Binary Circuits”. In: *Advances in Cryptology—CRYPTO 2020, Part II*. Vol. 12171. Lecture Notes in Computer Science. Springer, 2020, pp. 823–852. DOI: [10.1007/978-3-030-56880-1_29](https://doi.org/10.1007/978-3-030-56880-1_29). Also at ia.cr/2020/338.
- [ENSTW23]** Thomas Espitau, Thi Thu Quyen Nguyen, Chao Sun, Mehdi Tibouchi, and Alexandre Wallet. “Antrag: Annular NTRU Trapdoor Generation”. In: *Advances in Cryptology—ASIACRYPT 2023, Part VII*. Ed. by Jian Guo and Ron Steinfeld. Vol. 14444. Lecture Notes in Computer Science. Singapore: Springer, 2023, pp. 3–36. DOI: [10.1007/978-981-99-8739-9_1](https://doi.org/10.1007/978-981-99-8739-9_1). Also at ia.cr/2023/1335.
- [FHKLP+20]** Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Prest, Thomas Ricosset, Gregor Seiler, William Whyte, and Zhenfei Zhang. *Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU. Specification v1.2*. October 1, 2020. URL: <https://falcon-sign.info/falcon.pdf> (visited on 08/04/2026).
- [FKOS15]** Tore Kasper Frederiksen, Marcel Keller, Emmanuela Orsini, and Peter Scholl. “A Unified Approach to MPC with Preprocessing Using OT”. In: *Advances in Cryptology – ASIACRYPT 2015*. Lecture Notes in Computer Science. Springer, 2015. Also at ia.cr/2015/901.
- [GEPTW26]** Radhika Garg, Daniel Escudero, Antigoni Polychroniadou, Akira Takahashi, and Xiao Wang. *Thresholdizing Standardized FALCON Signatures*. Cryptology ePrint Archive, Paper 2026/1300. Accepted to ACM CCS 2026. 2026. URL: <https://eprint.iacr.org/2026/1300>.

[Hermine-PW] Hermine Team. *Hermine: An Efficient Raccoon-Style Non-Interactive Threshold Signature with Advanced Properties*. *Preview Writeup 1.0, NIST Threshold Call*. 2026. URL: <https://csrc.nist.gov/csrc/media/Projects/threshold-cryptography/documents/TCall-1/Hermine-PW01.pdf>.

[IOZ14] Yuval Ishai, Rafail Ostrovsky, and Vassilis Zikas. “Secure Multi-Party Computation with Identifiable Abort”. In: *Advances in Cryptology—CRYPTO 2014, Part II*. Vol. 8617. Lecture Notes in Computer Science. Springer, 2014. DOI: [10.1007/978-3-662-44381-1_21](https://doi.org/10.1007/978-3-662-44381-1_21).

[KOS16] Marcel Keller, Emmanuela Orsini, and Peter Scholl. “MASCOT: Faster Malicious Arithmetic Secure Computation with Oblivious Transfer”. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS 2016)*. ACM, 2016, pp. 830–842. DOI: [10.1145/2976749.2978357](https://doi.org/10.1145/2976749.2978357). Also at ia.cr/2016/505.

[Lemur-PW] Lemur Multi-Signature Team. *Lemur: Scalable Post-Quantum Synchronized Multi-Signatures*. *Preview Writeup 0.2, NIST Threshold Call*. 2026. URL: <https://csrc.nist.gov/csrc/media/Projects/threshold-cryptography/documents/TCall-1/Lemur-PW02.pdf>.

[LoTRS-PW] LoTRS Team. *LoTRS: Practical Post-Quantum Structured Threshold Ring Signatures from Lattices*. *Preview Writeup 0.1, NIST Threshold Call*. 2026. URL: <https://csrc.nist.gov/csrc/media/Projects/threshold-cryptography/documents/TCall-1/LoTRS-PW02.pdf>.

[Mithril-PW] Mithril Team. *Mithril: Efficient Threshold ML-DSA from Secret Sharing with Short Shares*. *Preview Writeup 1.0, NIST Threshold Call*. 2026. URL: <https://csrc.nist.gov/csrc/media/Projects/threshold-cryptography/documents/TCall-1/Mithril-PW01.pdf>.

[Nat26] National Institute of Standards and Technology. *Post-Quantum Cryptography Standardization Process*. 2026. URL: <https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Post-Quantum-Cryptography-Standardization> (visited on 09/29/2026).

[NO07] Takashi Nishide and Kazuo Ohta. “Multiparty Computation for Interval, Equality, and Comparison Without Bit-Decomposition Protocol”. In: *Public Key Cryptography—PKC 2007*. Vol. 4450. Lecture Notes in Computer Science. Springer, 2007, pp. 343–360.

[Pre15] Thomas Prest. “Gaussian Sampling in Lattice-Based Cryptography”. PhD thesis. Paris, France: École Normale Supérieure, 2015. URL: <https://tprest.github.io/pdf/pub/thesis-thomas-prest.pdf> (visited on 08/06/2026).

[Quorus-PW] Quorus Team. *Quorus: Scalable Threshold ML-DSA from MPC*. *Preview Writeup 0.1, NIST Threshold Call*. 2026. URL: <https://csrc.nist.gov/csrc/media/Projects/threshold-cryptography/documents/TCall-1/Quorus-PW01.pdf>.

[RW19] Dragos Rotaru and Tim Wood. “MARbled Circuits: Mixing Arithmetic and Boolean Circuits with Active Security”. In: *Progress in Cryptology—INDOCRYPT 2019*. Vol. 11898. Lecture Notes in Computer Science. Springer, 2019, pp. 227–249. DOI: [10.1007/978-3-030-35423-7_12](https://doi.org/10.1007/978-3-030-35423-7_12). Also at ia.cr/2019/207.

[TALUS-PW] TALUS Team. *TALUS: Threshold ML-DSA with One-Round Online Signing. Preview Writeup 0.21, NIST Threshold Call.* 2026. URL: <https://csrc.nist.gov/csrc/media/Projects/threshold-cryptography/documents/TCall-1/TALUS-PW02.pdf>.

[Tanuki-PW] Tanuki Team. *Tanuki: Two-round Threshold Signatures from Lattices. Preview Writeup 0.1, NIST Threshold Call.* 2026. URL: <https://csrc.nist.gov/csrc/media/Projects/threshold-cryptography/documents/TCall-1/Tanuki-PW01.pdf>.

[NIST-IR8214C] Luís T. A. N. Brandão and René Peralta. *NIST First Call for Multi-Party Threshold Schemes.* National Institute of Standards and Technology Internal Report (IR) 8214C. January 2026. DOI: [10.6028/NIST.IR.8214C](https://doi.org/10.6028/NIST.IR.8214C).