

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

## **Title:** (Red)ETA: Refreshable Extensible DLOG Enhanced Threshold Algorithms

**Subtitle:** Discrete-Log Extensible and Refreshable Decentralized Key Generation and Threshold Signing Algorithms with Enhanced Access Control

**Version:** 0.1 (2026-01-21)<sup>1</sup>

**Team name:**  $\eta$ : (Red)ETA-Project

**Team members:** Alessandro Barengi, Michele Battagliola, Riccardo Longo, Alessio Meneghetti, Gerardo Pelosi, Edoardo Signorini

**Abstract:** The (Red)ETA project (or  $\eta$ -project) is a research effort carried on to create a wide basis of documentation, security assessments, efficiency analyses and optimizations, and reference implementations, for decentralized cryptographic protocols beyond the classic threshold setting. In this submission we present  $\eta$ -keygen,  $\eta$ -Schnorr,  $\eta$ -EdDSA, and  $\eta$ -ECDSA, a complete set of algorithms to deal with Elliptic-Curve Cryptography and in particular with Digital Signature Schemes (ECC DSSs). A key feature of this submission is  $\eta$ -keygen, an enhanced DKG allowing for highly flexible key-management from linear secret sharing techniques, which includes decentralization, extensibility, resistance against adaptive, snapshot, and mobile attackers, and compatibility with basic  $(t,n)$ -threshold scenarios as well as more complex access structures. By integrating this enhanced DKG with threshold signing algorithm, our submission presents the specification of  $\eta$ -Schnorr,  $\eta$ -EdDSA, and  $\eta$ -ECDSA, which are threshold ECC DSSs compatible, thanks to  $\eta$ -keygen, with general monotone access structures.

**Proposed crypto-systems:** ECC KeyGen (N4.1); Threshold Schnorr Signature (N1.1); Threshold EdDSA Signature (N1.1); Threshold ECDSA Signature (N1.2);

**Keywords:** Threshold Cryptography; NIST Threshold Call; DKG; EdDSA; Schnorr; ECDSA



---

<sup>1</sup>Preliminary version submitted to NIST-MPTC for review

**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:** Alessandro Barengi<sup>i1,a1</sup>, Michele Battagliola<sup>i2,a2</sup>, Riccardo Longo<sup>i3,a3</sup>, Alessio Meneghetti<sup>i4,a4</sup>, Gerardo Pelosi<sup>i5,a1</sup>, Edoardo Signorini<sup>i6,a5</sup>

### Open Researcher and Contributor Identifiers (ORCID):

i1 ([0000-0003-0840-6358](#)); i2 ([0000-0002-8269-2148](#)); i3 ([0000-0002-8739-3091](#)); i4 ([0000-0002-5159-7252](#)); i5 ([0000-0002-3812-5429](#)); i6 ([0000-0002-1224-6732](#))

### Affiliations:

<sup>a1</sup> Department of Electronics Information and Bioengineering, Polytechnic University of Milan, Milan, Italy

<sup>a2</sup> Department of Engineering, Marche Polytechnic University, Ancona, Italy

<sup>a3</sup> Center for Cybersecurity, Aleph research unit, Fondazione Bruno Kessler, Trento, Italy

<sup>a4</sup> Department of Mathematics, University of Bari Aldo Moro, Bari, Italy

<sup>a5</sup> Telsy SpA, Turin, Italy

### Associateship clarifications:

.

**Note on Author Contributions:** Authors listed in alphabetical order

### Main contacts:

- **Team mailing list:** [team@eta-project.org](mailto:team@eta-project.org)
- **Primary technical contact person:** Alessio Meneghetti, [alessio.meneghetti@uniba.it](mailto:alessio.meneghetti@uniba.it)
- **Secondary contact person 1:** Riccardo Longo, [rlongo@fbk.eu](mailto:rlongo@fbk.eu)

**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

Decentralized systems are becoming the de-facto standard in data-management, especially in applications involving sensitive data and collaborative decision-making (e.g. e-voting), due to their security against single-points of failures and their ability to protect the users' privacy while offering protection from key-loss and similar drawbacks present in centralized systems. In this contest, (decentralized) Elliptic-Curve Cryptography, in particular digital signatures such as ECDSA and EdDSA, constitutes the core of many state-of-the-art decentralized architectures (see e.g. [DLMR20]). Particularly relevant is the application in the metaverse, where the enforcing of regulation is very tricky due to transnational domains and the lack of central governance (or the security and trust risk that centralization would pose)[Ranar]. In this context many application would benefit from control policies that are finer-grained than a simple threshold, and their dynamic nature would also appreciate the possibility to add actors and proactively restore security in case of partial compromise.

This submission focuses on threshold ECC DSSs, i.e. EdDSA/Schnorr and ECDSA, and comprises the key generation, called  $\eta$ -keygen, and the signing algorithms, respectively called  $\eta$ -Schnorr,  $\eta$ -EdDSA, and  $\eta$ -ECDSA. In particular, the key generation is derived from [BLM25; LMM25] and is a key feature of the submission, as it allows for highly flexible key-management from linear secret sharing techniques, and it includes features such as decentralization, extensibility, refreshability (resistance against adaptive, snapshot, mobile attackers), and compatibility with basic  $(t,n)$ -threshold scenarios as well as general access structures (threshold access trees). The key generation is discussed in detail for the general case of decentralized secret generation [BLM25; LMM25], specialized to the case of ECC, instantiated and implemented for EdDSA/Schnorr (derived from [BGLM+22; BLMS23; BLM25]) and ECDSA (derived from [BLMS21]). The key generation  $\eta$ -keygen fits in NIST Category N4.1, however some features (e.g. threshold random values generation, key refreshing, addition of new participants, management of complex access policies) might be described and implemented as independent packages fitting Categories N4.5 and S7.  $\eta$ -Schnorr (Category N1.1),  $\eta$ -EdDSA (Category N1.1, even though some features, e.g. the deterministic nonce generation, might be described as an independent gadget), and  $\eta$ -ECDSA (Category N1.2) signing algorithms are then presented and implemented, following respectively the work in [BLMS23], [BGLM+22; BLM25] and [BLMS21]. A package containing key-generation and signing algorithms will be provided and used for performance evaluation and comparison.

# 2. Specification

- **Organization:** First, the key generation algorithm will be presented in detail for the general case of decentralized secret generation, alongside its extended features such as extensibility, share refreshing, and generalization from simple threshold to general monotone access structures. The construction will then be specialized for the case of ECC and implemented.

Subsequently, Schnorr, EdDSA and ECDSA will be presented and implemented each one in its own module. All three will depend on the Key Generation module, and EdDSA will build on the Schnorr module adding the deterministic nonce generation.

Each part of the submission is managed by a dedicated subteam (possibly including researchers not yet listed as team members).

- **System model:** ETA DKG enables a trustless distributed setup that is extensible to parties that join later on, allows for refreshing to achieve proactive security, and supports general monotone access structures to allow for finer-grained control with respect to simple thresholds. This enhanced DKG exploits a homomorphic commitment scheme to enforce fairness and enable full verification, in particular Pedersen commitment [Ped92] satisfies the requirements and fits well within the application to ECC-based cryptographic schemes such as ECDSA, and EdDSA/Schnorr.

ETA ECDSA uses also the Paillier Cryptosystem [Pai] in the multi-party signature generation. For the deterministic nonce generation, ETA EdDSA exploits the Elliptic-Curve-based Pseudo-Random Function (PRF) Purify [NRSW20].

- **Security:** We consider the following definitions:
  - **Static Adversary:** An adversary whose strategy is fixed and determined before the protocol begins.
  - **Adaptive Adversary:** An adversary that can adapt their strategy based on the information gathered during the protocol's execution.
  - **Mobile Adversary:** An adversary that can move between different players over time (corrupting them), as long as the total number of corrupt players is less than or equal to a fixed threshold. A mobile adversary is adaptive, but an adaptive adversary is not necessarily mobile.
  - **Snapshot Adversary:** An adversary that captures and analyzes the system's state and the secrets of corrupted players at a single point in time, without the ability to continuously monitor or interfere with the protocol.

$\eta$ -keygen is secure against an adaptive, mobile, and snapshot adversary [LMM25].

$\eta$ -EdDSA, and  $\eta$ -ECDSA are proven secure against static adversaries who, contrary to the mainstream approach, are also involved in the key-gen [BLMS21; BLMS23], while their security against adaptive and mobile adversaries is currently under investigation.  $\eta$ -Schnorr is secure against static adversaries participating in both the key-generation and the signing algorithm [BGLM+22; BLM25], and is secure against adaptive adversaries not participating in the key generation [CKM23]. Its security against mobile adversaries is currently under investigation.

The security assumptions required to prove the security properties listed above are mostly dependent on the implementation choices and will be discussed in detail in the submission (e.g. if in  $\eta$ -keygen we adopt the Pedersen commitment then the security depends on the hardness of DLOG; Similarly,  $\eta$ -EdDSA requires a verifiable PRF on ECC points such as Purify).

### 3. Open-Source Implementation

1. **Code structure:** The codebase is expected to be organized as a library designed with components having increasing levels of abstraction. In particular, we foresee a component tackling efficient modular arithmetic, another one containing symmetric primitives (namely, cryptographic hashes), and a component employing the one implementing efficient modular arithmetic and realizing elliptic curve arithmetic. Finally, a box-component will employ the aforementioned ones to realize the entire cryptographic primitives. We aim to have a self-contained implementation as a target; in the initial phases of our implementation we will likely employ open source, freely licensed (i.e., Public Domain, CC-0), libraries, such as Libtomcrypto [Tea26] to provide the efficient arithmetic and curve arithmetic, and hash functions. We expect the entire library to be written in ISO standard C11 [Int11], and we will test its correct functionality when built with the latest stable versions of the GNU Compiler Collection (GCC) and the Clang/LLVM compiler.
2. **Code progress and availability:** Currently, we are in the modular design phase, and we are in the process of identifying the interfaces of the library modules.
3. **Implementation of the networking model:** Given the wide variety of the possible networking stacks, we foresee a decoupling of the networking capabilities from the library itself. Therefore we foresee an interface for networking which will have compile-time selectable back-ends. A simple testing back-end, for the purpose of easing portability, will be constituted by files stored on local mass memory, while we foresee that the back-end employed for performance testing will be implemented with POSIX-standard TCP sockets. Realistic network latency testing can be performed easily in this way through network delay simulation suites such as the *netem* one included in Linux.
4. **Testing:** We foresee a test-vector based approach for functionality testing of the library, including both positive- (i.e., functional input-output pairs) and negative test vectors (i.e., non functional input-output pairs to detect missing error signalling). Adversarial testing will be done by preparing test suites which stress the ability of the code to react appropriately to known attacks to the signature scheme. Finally, randomized-fault testing will be performed through random bit spraying on functional test vectors, in the fashion currently applied in the liboqs project by OpenQuantumSafe.

### 4. Experimental performance evaluation

1. **Performance:** We expect that a synthetic performance estimate, obtained by counting the number of elliptic curve point operations and cryptographic hash calls, and multiplying these values by the latency of a single of the aforementioned operations will yield a quite good estimation of the overall runtime. We plan on comparing our performance with other submitters as soon as their codebases are public.
2. **Platform:** We expect the baseline platform (single computer, with 16 cores and 64 GB of RAM, as mentioned in the response to item F2b.3.1 in the compilation of public comments

[PubComs2PD] on the 2pd) to be plentiful in resources for the single-host testing (i.e., benchmarking performance without networking). We also foresee that such a configuration is still resourceful enough to perform tests with limited amounts of hosts (in the low-tens range), although contention on hardware resources (second and third level caches) may somehow have an impact on the results reliability. We would welcome any public baseline for network testing, whether this is simply created by splitting a single host into multiple virtual machines with CPU pinning, or having different physical hosts interconnected by an actual network. We would recommend in this case a simple, relatively low budget setup (e.g., two hosts, one to four 1000-Base-T interconnections) to allow easy reproducibility by submitters and third party implementors likewise.

## 5. Licensing, patent claims, and funding

1. We foresee to release our entire codebase as Public Domain (CC0, <https://creativecommons.org/public-domain/cc0/>), and strive to have similarly licensed libraries as dependencies, if needed, whether bundled or external.
2. US Patent [DSML24] may be relevant since it was granted for an invention stemmed from the preliminary work that led to [BLMS21]. In particular it applies to using extensible threshold signatures for resilient custody of cryptocurrencies. Two of the inventors are authors of this submission, the company who is the current assignee of the patent is not involved in the submission.
3. Michele Battagliola is supported by the Italian Ministry of University and Research (MUR) under the PRIN 2022 program with projects “Mathematical Primitives for Post Quantum Digital Signatures” (CUP I53D23006580001) and “Post quantum Identification and eN-cryption primiTives: dEsign and Realization (POINTER)” (CUP I53D23003670006), by MUR under the Italian Fund for Applied Science (FISA 2022), project “Quantum-safe cryptographic tools for the protection of national data and information technology assets” (QSAFEIT) - No. FISA 2022-00618 (CUP I33C24000520001).

Alessio Meneghetti is supported by the project PRIN 2022SC, title “Algebraic Methods in Cryptanalysis”, Grant Ref. 2022RFAZCJ, CUP H53C24000830006.

Riccardo Longo is supported by project SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union — NextGenerationEU.

## References

- [BGLM+22] Michele Battagliola, Alessio Galli, Riccardo Longo, Alessio Meneghetti, et al. “A provably-unforgeable threshold Schnorr signature with an offline recovery party”. In: *Ceur Workshop Proceedings*. Vol. 3166. CEUR-WS. org. 2022, pp. 60–76. URL: <https://ceur-ws.org/Vol-3166/paper05.pdf>.
- [BLM25] Michele Battagliola, Riccardo Longo, and Alessio Meneghetti. “Extensible decentralized secret sharing and application to Schnorr signatures”. In: *Designs, Codes and Cryptography* 94.1 (December 2025). DOI: [10.1007/s10623-025-01746-1](https://doi.org/10.1007/s10623-025-01746-1). Also at [ia.cr/2022/1551](https://ia.cr/2022/1551).
- [BLMS21] Michele Battagliola, Riccardo Longo, Alessio Meneghetti, and Massimiliano Sala. “Threshold ECDSA with an Offline Recovery Party”. In: *Mediterranean Journal of Mathematics* 19.1 (November 2021). DOI: [10.1007/s00009-021-01886-3](https://doi.org/10.1007/s00009-021-01886-3). Also at [arXiv:2007.04036](https://arxiv.org/abs/2007.04036).
- [BLMS23] Michele Battagliola, Riccardo Longo, Alessio Meneghetti, and Massimiliano Sala. “Provably Unforgeable Threshold EdDSA with an Offline Participant and Trustless Setup”. In: *Mediterranean Journal of Mathematics* 20.5 (June 2023). DOI: [10.1007/s00009-023-02452-9](https://doi.org/10.1007/s00009-023-02452-9). Also at [https://iris.unitn.it/retrieve/handle/11572/384212/660585/BLMS\\_eddsa.pdf](https://iris.unitn.it/retrieve/handle/11572/384212/660585/BLMS_eddsa.pdf).
- [CKM23] Elizabeth Crites, Chelsea Komlo, and Mary Maller. “Fully Adaptive Schnorr Threshold Signatures”. In: *Advances in Cryptology – CRYPTO 2023*. Ed. by Helena Handschuh and Anna Lysyanskaya. Cham: Springer Nature Switzerland, 2023, pp. 678–709.
- [DLMR20] Vincenzo Di Nicola, Riccardo Longo, Federico Mazzone, and Gaetano Russo. “Resilient custody of crypto-assets, and threshold multisignatures”. In: *Mathematics* 8.10 (2020), p. 1773.
- [DSML24] Vincenzo Di Nicola, Massimiliano Sala, Alessio Meneghetti, and Riccardo Longo. “Method and apparatus for a blockchain-agnostic safe multi-signature digital asset management”. US11915314B2. 2024.
- [Int11] International Organization for Standardization. *Information technology — Programming languages — C*. Standard. Geneva, CH: International Organization for Standardization, December 2011.
- [LMM25] Riccardo Longo, Alessio Meneghetti, and Sara Montanari. *Tighter Control for Distributed Key Generation: Share Refreshing and Expressive Reconstruction Policies*. Cryptology ePrint Archive, Paper 2025/277. 2025. URL: <https://eprint.iacr.org/2025/277>.
- [NRSW20] Jonas Nick, Tim Ruffing, Yannick Seurin, and Pieter Wuille. “MuSig-DN: Schnorr Multi-Signatures with Verifiably Deterministic Nonces”. In: *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. CCS '20.

- Virtual Event, USA: Association for Computing Machinery, 2020, pp. 1717–1731. DOI: [10.1145/3372297.3417236](https://doi.org/10.1145/3372297.3417236). Also at [ia.cr/2020/1057](https://ia.cr/2020/1057).
- [Pai] Pascal Paillier. “Public-Key Cryptosystems Based on Composite Degree Residuosity Classes”. In: *Advances in Cryptology — EUROCRYPT '99*. Springer Berlin Heidelberg, pp. 223–238. DOI: [10.1007/3-540-48910-x\\_16](https://doi.org/10.1007/3-540-48910-x_16).
- [Ped92] Torben Pryds Pedersen. “Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing”. In: *Advances in Cryptology — CRYPTO '91*. Ed. by Joan Feigenbaum. Berlin, Heidelberg: Springer Berlin Heidelberg, 1992, pp. 129–140.
- [Ranar] Silvio Ranise. “Metaverse Justice: A Cybersecurity Perspective”. In: *Ordine Internazionale e Diritti Umani* (2026 (to appear)).
- [Tea26] Team LibTom. *LibTomCrypt Cryptographic Library*. <https://www.libtom.net/LibTomCrypt/>. 2026.
- [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. 2025. DOI: [10.6028/NIST.IR.8214C](https://doi.org/10.6028/NIST.IR.8214C).
- [PubComs2PD] NIST-MPTC. *Compilation of Public Comments on NISTIR 8214C 2pd*. National Institute of Standards and Technology, Multi-Party Threshold Cryptography. June 2025. URL: <https://csrc.nist.gov/files/pubs/ir/8214/c/2pd/docs/nistir-8214c-2pd-public-feedback.pdf>.