
Zero Knowledge Arguments for the Post-Quantum Era

Randy Kuang¹, Daniel Johnson^{2,*} and Daniel Panario³

¹*Quantropi Inc., Canada*

²*NC-CIPSeR, Carleton University, Canada*

³*School of Mathematics and Statistics, Carleton University, Canada*

*E-mail: randy.kuang@quantropi.com; danieljohnson2@cmail.carleton.ca;
daniel@math.carleton.ca*

**Corresponding Author*

Received 24 July 2026; Accepted 27 July 2026

Abstract

The approaching threat of quantum computing to conventional cryptography underscores the urgent need to reassess information security and develop practical quantum-safe primitives. In this paper, we present concrete implementations of Zero-Knowledge Arguments (ZKAs), analogous to but distinct from traditional Σ -protocols. Our implementation, termed the Cryptographic Primitive Argument (CPA), can be realized securely using any post-quantum Key Exchange Mechanism (KEM) and Digital Signature (DS) scheme. We describe two variants: one with dynamic public key encapsulation (CPA-DE) and another that is non-interactive (CPA-NI). We further detail efficient, NIST-standard-compliant CPAs instantiated with ML-KEM and ML-DSA, and provide a comparison with a post-quantum Σ -protocol. The lightweight and modular nature of our CPAs makes them suitable for a wide range of applications, including quantum-secure authentication, blockchain systems, and digital currencies.

Keywords: Post-quantum cryptography, public-key cryptography, PQC, key encapsulation mechanism, KEM, multivariate polynomials, PQC performance, hidden ring, diophantine equation problem.

Quantum Information Technologies Journal, Vol. 2-1, 41–60.

doi: 10.13052/qitj2795-0492.213

© 2026 River Publishers

1 Introduction

The evolution of computing paradigms heavily involves itself within the domain of information security. While public key cryptography has for years relied on algorithms like RSA [35] and variants of Diffie-Hellman [20, 6], theoretical advances in quantum computing have undermined general confidence in these defenses. Among the most conspicuous examples, large-scale implementations of Shor’s factorization algorithm [38] would be lethal to RSA encryption.

In preparation for the impending quantum threat, the National Institute of Standards and Technology (NIST) initiated its groundbreaking Post-Quantum Cryptography (PQC) standardization competition in 2016. This process has meticulously evaluated candidate algorithms submitted by the global cryptographic community. Since then, numerous cryptologists, mathematicians, and cybersecurity experts have worked toward establishing a robust suite of algorithms to seamlessly replace existing ones, thereby ensuring continued digital security in the post-quantum era.

NIST has divided its submissions into categories of primitives, such as Key Encapsulation Mechanisms (KEMs) and Digital Signature (DS) schemes, encompassing diverse mathematical approaches. The KEM category has included cryptosystems based on lattices [14, 5], codes [31, 11], multivariate polynomials [22, 39], and supersingular isogenies [26, 25, 18]. Likewise, digital signatures have included lattice-based [29, 34], multivariate polynomial [21], and hash-based schemes [4]. In late 2022, NIST announced its first competition winners: Kyber for KEM, and Dilithium, Falcon, and SPHINCS for DSs [32]. In 2024, finalized versions of Kyber and Dilithium were released, renamed ML-KEM [2] and ML-DSA [19]. Later, in 2025, HQC [1] was also selected for standardization.

These cryptographic primitives are founded on well-studied NP-hard problems, such as the Shortest Vector Problem [30] for lattice-based cryptography, the Multivariate Quadratic Problem for MPKC [15], and the Syndrome Decoding Problem for code-based schemes. While these problems have been broadly assessed for their resistance to quantum attack, specific cryptographic implementations may still face vulnerabilities. Notably, in 2022, with the NIST Round 3 finalists, Robert [36] reported a polynomial-time attack on Supersingular Isogeny Diffie–Hellman. Shortly thereafter, Castryck and Decru developed a more streamlined key-recovery attack [16], successful against NIST security level V in under two hours using only a standard laptop.

Beyond KEMs and DSs, other cryptographic protocols have been developed to address different needs. Of particular interest here are Zero-Knowledge Proofs (ZKPs). In the 1980s, Goldwasser, Micali, and Rackoff laid the groundwork for ZKPs by introducing interactive proofs, which allow one party (the prover) to convince another (the verifier) of a statement’s truth without revealing any additional information about the statement itself [27].

In the literature, *Zero-Knowledge Proof (ZKP)* and *Zero-Knowledge Argument (ZKA)* are sometimes used interchangeably. ZKPs guarantee information-theoretic soundness, whereas a ZKA guarantees soundness only against computationally bounded adversaries. In practice, most modern constructions (including those in this paper) are computationally secure, and the distinction is often blurred. For clarity, we consistently use *ZKA* to refer to computational zero-knowledge protocols.

ZKP research evolved further as Non-Interactive Zero-Knowledge Proofs (NI-ZKPs), introduced by Blum, Feldman, and Micali in 1988 [13], removing the need for real-time interaction. NI-ZKPs underpin protocols such as zk-SNARKs [12], widely used in blockchain and cryptocurrencies, though they require a trusted setup.

Additional developments include Zero-Knowledge Scalable Transparent Arguments of Knowledge (zk-STARKs), introduced by Ben-Sasson et al. in 2018 [9], which offer transparent ZKP protocols without a trusted setup. An elliptic-curve-based version was later proposed in 2022 [10].

Other ZKP variants have emerged along different lines. Basso et al. proposed a statistical ZKP of Isogeny knowledge in 2023 [8], Chase et al. presented a post-quantum ZKP in an efficient Σ -protocol in 2017 [17], and Almuhammadi and Neuman proposed ZKP-1, a single-round, challenge-response ZKP based on the discrete logarithm in 2005 [3], suitable for identity authentication.

In 2024, Zhou et al. [42] highlighted the need for further advances in ZKPs to balance privacy and security in blockchains, and Ernstberger et al. [24] emphasized the potential of distinct ZKP varieties for practical applications.

Regarding Σ -protocols, while Schnorr’s original design [37] was based on the discrete logarithm, alternative and quantum-resistant methods have been explored. For example, Xue et al. employed lattices in Compressed Σ -protocols [41], Bartoli and Cascudo studied module homomorphisms [7], and in 2023, the LRMC Σ -protocol [40] was also introduced for post-quantum

implementation. This particular protocol uses the computationally hard Low-Rank Matrix Completion Problem, there proven equivalent to the MinRank Problem, to allow for execution with greater degrees of conceptual clarity and efficiency.

Of particular importance here is the concept of a Zero Knowledge Argument (ZKA), equivalently defined in 2007 [33] as a Computational Zero Knowledge Argument (CZKA). Our paper is situated within this milieu of ZKPs, Σ -protocols, KEMs, and DS schemes. Section 2 outlines and motivates our contributions. Section 3 presents our framework for ZKAs, with Subsection 3.1 discussing our formal definition in the context of Σ -protocols. Subsection 3.2 introduces a specific and highly flexible form of ZKA, the Cryptographic Primitive Argument (CPA), which can employ any KEM and DS scheme. We also describe two variant implementations, including a non-interactive version. Section 4 details efficient adaptations of the ML-KEM and ML-DSA for our CPAs, comparing their memory requirements with the LRMC Σ -protocol. Conclusions are given in Section 5.

2 Motivation and Contributions

It is broadly accepted within the cryptographic community that an attacker’s overwhelming probability of failure constitutes sufficient security, as reflected in the criteria required of the NIST PQC candidates.

This principle is embodied in the notion of a ZKA. To illustrate, we start by providing informal but intuitive definitions of its three requisite properties: correctness, soundness, and zero-knowledge.

Correctness: With overwhelming probability, an honest verifier accepts the testimony of an honest prover who holds the secret knowledge.

Soundness: A dishonest prover, lacking the secret knowledge, has negligible probability of convincing an honest verifier.

Zero-Knowledge: By executing the protocol, any verifier learns only that the prover holds the desired secret knowledge, while gaining negligible additional information about it.

Two points should be emphasized. First, these criteria naturally extend to the post-quantum setting, where a probability is considered negligible if it remains so even against fully quantum adversaries. Second, the ZKA is an abstract framework: its security is written into the definition, but the cryptographer must still rigorously prove that the conditions hold for any proposed protocol.

To this end, we define the CPA, which adapts any pair of existing post-quantum KEM and DS schemes into a secure ZKA implementation. Moreover, by instantiating CPAs with the ML-KEM and ML-DSA, we obtain efficient constructions that are ready for deployment. The possibility of using other KEMs and DSs remains entirely open.

To state our contributions with greater detail and precision:

- **Basic CPA Definition:** We propose the novel CPA protocol within the space of ZKAs. The CPA requires a single execution and allows implementation with any post-quantum KEM and DS, offering a secure and efficient interaction between prover and verifier. This protocol establishes the prover’s knowledge of the private key pair associated with a public KEM encapsulation and DS verification key pair.
- **Two Extra CPA Variants:** We also introduce two CPA variants useful for differing practical circumstances. The first variant saves on static memory by allowing dynamic generation of the public KEM encapsulation key. The second is a non-interactive CPA for scenarios with heavy restrictions on real-time exchange of information. Using any secure post-quantum primitives, our CPAs are readily applicable to real-world scenarios of quantum-secure authentication, blockchain and digital currency.
- **Concrete Post-Quantum CPA Implementations:** We adapt the NIST PQC’s successful ML-KEM and ML-DSA to the above three CPA models, ready for use, detailing their competitive memory requirements. These implementations are called ML-CPAs.

We thus show the CPA, and particularly the ML-CPAs, as robust post-quantum cryptographic solutions.

3 Zero-Knowledge Arguments in the Post-Quantum Era

The following subsections present the essential concepts and definitions for a Computational Zero-Knowledge Argument, as well as the Cryptographic Primitive Argument, its variants, and Module-Lattice implementations.

3.1 Essential Concepts and Definition

The core purpose of a ZKA is to demonstrate possession of some specific knowledge with only negligible information leakage.

A ZKA, as defined in the 2007 [33], operates similarly to a Σ -protocol, though instead requires computational correctness, computational soundness

and computational zero knowledge (given formally below), rather than the typical three conditions of correctness, n -special soundness and Special Verifier Zero-Knowledge [8].

In giving a more rigorous presentation we subsequently use the general framework of probabilistic polynomial time (PPT) functions. The output x of a probabilistic algorithm $\mathcal{A}$ is denoted by $x \leftarrow \mathcal{A}$, whereas $x = \mathcal{A}$ means the algorithm is deterministic (i.e., a probability of 1).

In specifying our essential Definition 1 to the post-quantum setting we consider $\text{negl}(\Lambda)$ as within the minimum NIST security standards, assuming adversaries operate with fully quantum capacities. With $\text{negl}(\Lambda) \leq 2^{-128}$ for NIST Security level I, with $\text{negl}(\Lambda) \leq 2^{-192}$ and $\text{negl}(\Lambda) \leq 2^{-256}$ for levels III and V, respectively.

With adaptations from the definition in [33] as allowing for concrete implementation, we now provide a formal description.

Definition 3.1. *A Zero-Knowledge Argument for a binary relation $\mathcal{R}$ with security parameter $\Lambda \in \mathbb{N}$ consists of PPT algorithms $(\mathbf{P1}, \mathbf{P2}, \mathbf{V1}, \mathbf{V2})$, where $\mathbf{P1}, \mathbf{P2}$ belong to the prover and $\mathbf{V1}, \mathbf{V2}$ belong to the verifier, along with the following protocol:*

1. *With $(x, w) \leftarrow \mathcal{R}$, the prover computes the commitment, $\mathbf{com} \leftarrow \mathbf{P1}(x, w)$; $\mathbf{com}$ is then sent to the verifier.*
2. *The verifier generates $M \leftarrow \{0, 1\}^\Lambda$ with uniform randomness, computing the challenge, $\mathbf{chall} \leftarrow \mathbf{V1}(\mathbf{com}, M)$, and sending it to the prover.*
3. *The prover computes the response, $\mathbf{resp} \leftarrow \mathbf{P2}(\mathbf{chall}, w)$, which is sent to the verifier.*
4. *The verifier accepts the proof of witness w if $\mathbf{V2}(\mathbf{com}, M, \mathbf{resp}) = 1$, rejecting otherwise.*

A ZKA must also satisfy the following three security criteria:

- **Computational Correctness:**

For an honest verifier engaged by an honest prover having any $(x, w) \in \mathcal{R}$,

$$\mathcal{P}[\mathbf{V2}(\mathbf{com}, \mathbf{chall}, \mathbf{resp}) = 1] \geq 1 - \text{negl}(\Lambda).$$

- **Computational Soundness:**

For a dishonest prover without valid $(x, w) \in \mathcal{R}$,

$$\mathcal{P}[\mathbf{V2}(\mathbf{com}, \mathbf{chall}, \mathbf{resp}) = 1] \leq \text{negl}(\Lambda).$$

- **Computational Zero-Knowledge:**

For a dishonest verifier, assuming an unbounded PPT adversary algorithm $\mathcal{A}$ and uniform random generator $\mathcal{U}$ for $\mathcal{R}$,

$$|\mathcal{P}[(x, w) \leftarrow \mathcal{A}(\mathbf{com}, M, \mathbf{resp})] - \mathcal{P}[(x, w) \leftarrow \mathcal{U}]| \leq \text{negl}(\Lambda).$$

The above conditions strongly guarantee the success of an honest prover and honest verifier, the failure of a dishonest prover, and the failure of a dishonest verifier, respectively.

For further contrasts to Σ -protocols (and also the generic ZKA definition) we allow two degrees of greater generality. First, the witness may be used to compute **P2**. Second, M , and not **chall**, is used as input for **V2**.

The first contrast allows the burden of proof of knowledge for w to be split over two different tasks, while the second guarantees a verifier the flexibility of using the preimage of **chall**, knowing this latter value is computable in polynomial time.

3.2 The Cryptographic Primitive Argument (CPA)

We now give our specific implementation of the ZKA abstraction, the CPA, usable with any pair of quantum-secure DS and KEM.

3.2.1 Definition and security proofs

This subsection contains the details for our elementary CPA, and also the proof that it satisfies the three security conditions of a ZKA.

Specifying notation, for any tuple v and $j \in \mathbb{N}$ define v_j as the j -th coordinate of v . For the given primitive $\mathbf{p}$, either KEM or DS, let $\mathcal{SK}_{\mathbf{p}}$ be the secret keyspace.

We define several internal functions: Let $\mathbf{pkgen}_{\mathbf{p}}(\alpha)$ generate a DS or KEM public key from (presumed) secret key α . Let $\mathbf{Encap}(\alpha, \beta)$ be the KEM encapsulation function using public key α to transform the binary string β into its encapsulated (encrypted) form. For inversion, $\mathbf{Decap}(\alpha, \beta)$ uses secret KEM key α to recover the bit-string within an encrypted β . Let $\mathbf{Sign}(\alpha, \beta)$ be the DS signature function which treats α as a private DS key to sign the binary string β . Lastly, let $\mathbf{SigVerify}(\alpha, \beta, \gamma)$ be the DS verification function that uses public DS key α to verify that β is a valid signature for bit-string γ .

We assume, in accordance with the general ZKA setup, that an (honest) prover begins with a uniform $(x, w) \leftarrow \mathcal{R}$, and that the verifier uniformly generates $M \leftarrow \{0, 1\}^\Lambda$ as needed. Our last assumption is in having

appropriate input sizes for all given functions, using a post-quantum hash function (like SHA-256) if needed. Our fundamental definition follows.

Definition 3.2. *The Cryptographic Primitive Argument:*

$$\begin{aligned}\mathcal{R}_{CPA} = \{ & (x, w) : x = (pk_{kem}, pk_{ds}), w = (sk_{kem}, sk_{ds}), \\ & pk_p = \mathbf{pkgen}_p(sk_p), sk_p \in \mathcal{SK}_p, \\ & p \in \{kem, ds\}\},\end{aligned}$$

$\mathbf{com} \leftarrow \mathbf{P1}(x, w) = x,$
 $\mathbf{chall} \leftarrow \mathbf{V1}(\mathbf{com}, M) = \mathbf{Encap}(\mathbf{com}_1, M),$
 $\mathbf{resp} \leftarrow \mathbf{P2}(\mathbf{chall}, w) = \mathbf{Sign}(sk_{ds}, \mathbf{Decap}(sk_{kem}, \mathbf{chall})),$
 $\mathbf{V2}(\mathbf{com}, M, \mathbf{resp}) = 1$ if and only if $\mathbf{SigVerify}(\mathbf{com}_2, \mathbf{resp}, M) = 1.$

The above protocol has the following back-and-forth flow from prover **P** to verifier **V**, as

$$(x, w) \rightarrow \mathbf{P1} \xrightarrow{\mathbf{com}} \mathbf{V1} \xrightarrow{\mathbf{chall}} \mathbf{P2} \xrightarrow{\mathbf{resp}} \mathbf{V2} \rightarrow \{0, 1\},$$

with 1 being the result of $(x, w) \in \mathcal{R}_{CPA}$, 0 otherwise. We also reiterate that a CPA using secure post-quantum primitives effectively defends against even fully quantum adversaries. As our proofs show, an adversary can break the CPA protocol only if they can successfully attack a post-quantum KEM or DS scheme.

Theorem 3.3. *The Cryptographic Primitive Argument is a Zero Knowledge Argument.*

Proof. We first show the CPA has Computational Correctness. Given the assumed existence of functional DSs and KEMs, the PPT functions **P1**, **P2**, **V1** and **V2** all have a failure probability $\text{negl}(\Lambda)$ by assumption. We have

$$\begin{aligned}\mathcal{P}_{\text{Success}(CPA)} &= (1 - \mathcal{P}_{\text{Fail}(\mathbf{P1})}) (1 - \mathcal{P}_{\text{Fail}(\mathbf{P2})}) \\ &\quad \times (1 - \mathcal{P}_{\text{Fail}(\mathbf{V1})}) (1 - \mathcal{P}_{\text{Fail}(\mathbf{V2})}) \\ &\geq (1 - \text{negl}(\Lambda))^4 \approx 1 - \text{negl}(\Lambda),\end{aligned}$$

where we assume minor allowances are acceptable for the possible slight increase of having four negligible internal failure probabilities at their absolute acceptable limit of $\text{negl}(\Lambda)$. An honest prover and verifier thus have no tangible difficulty executing the protocol.

Next we show the CPA has Computational Soundness. With a dishonest prover possessing a CPA public key pair $(pk_{\mathbf{kem}}, pk_{\mathbf{ds}})$, he can indeed send a valid **com** to the verifier. Though, to grant acceptance for the whole protocol, the verifier must still obtain some **resp** from the prover such that $\text{Verify}(pk_{\mathbf{ds}}, M, \mathbf{resp}) = 1$.

This amounts to one of the following two cases: either successfully forging a signature or making a dual key-recovery attack to successfully execute the entire protocol. Both malicious operations of key-recovery and signature forging are assumed as having success probability $\text{negl}(\Lambda)$ given secure post-quantum primitives. Hence, let $\mathcal{A}_{F(DS)}$, $\mathcal{A}_{KR(DS)}$ and $\mathcal{A}_{KR(KEM)}$ be the optimal quantum forging and key-recovery algorithms against the given DS and KEM. As the original signed message M is only interceptable as the encrypted **chall**, in the case of mere forgery they must guess $M' \in \{0, 1\}^\Lambda$ uniformly. Hence, we must have

$$\begin{aligned} & \mathcal{P}[\mathbf{V2}(\mathbf{com}, \mathbf{chall}, \mathbf{resp}) = 1] \\ &= \begin{cases} \mathcal{P}[\mathbf{resp} \leftarrow \mathcal{A}_{F(DS)}(\mathbf{com}_2, M' \leftarrow \{0, 1\}^\Lambda)] & (\text{forgery}) \\ \mathcal{P}[sk_{\mathbf{kem}} \leftarrow \mathcal{A}_{KR(KEM)}(\mathbf{com}_1), \\ sk_{\mathbf{ds}} \leftarrow \mathcal{A}_{KR(DS)}(\mathbf{com}_2)] & (\text{key recovery}) \end{cases} \\ &= \begin{cases} \text{negl}(\Lambda) & (\text{forgery}) \\ \text{negl}(\Lambda)^2 & (\text{key recovery}) \end{cases} \leq \text{negl}(\Lambda). \end{aligned}$$

We observe that each $(pk_{\mathbf{kem}}, pk_{\mathbf{ds}})$ is uniquely related to one private key pair $(sk_{\mathbf{kem}}, sk_{\mathbf{ds}})$, where moreover there is no pre-specified relation between $sk_{\mathbf{kem}}$ and $sk_{\mathbf{ds}}$.

Lastly, we show the CPA is Computational Zero-Knowledge. Similar to the previous case, recovering $(x, w) \in \mathcal{R}_{CPA}$ from merely x means, optimally, using $x = \mathbf{com} = (pk_{\mathbf{kem}}, pk_{\mathbf{ds}})$ along with **resp** for a dual key-recovery attack against $w = (sk_{\mathbf{kem}}, sk_{\mathbf{ds}})$ as assisted by a genuine signature. Therefore, again assuming the security of input primitives we must have

$$\begin{aligned} & \mathcal{P}[(x, w) \leftarrow \mathcal{A}(\mathbf{com}, M, \mathbf{resp})] \\ &= \mathcal{P}[sk_{\mathbf{kem}} \leftarrow \mathcal{A}_{KR(KEM)}(\mathbf{com}_1), sk_{\mathbf{ds}} \leftarrow \mathcal{A}_{KR(DS)}(\mathbf{com}_2)] \\ &\leq \text{negl}_1(\Lambda)^2, \end{aligned}$$

where the subscript “1” (and later “2”) is to emphasize acceptable variance in this negligible probability value.

However, general standards require $|\mathcal{SK}_{\mathbf{p}}| \geq 2^\Lambda$ to guarantee negligible success for a brute-force search. A uniform random selection $\mathcal{U}$ for both keys thus succeeds with probability at most $2^{-2\Lambda}$, and hence $\mathcal{P}[(x, w) \leftarrow \mathcal{U}] \leq 2^{-2\Lambda} = \text{negl}_2(\Lambda)^2$, and hence

$$\begin{aligned} & |\mathcal{P}[(x, w) \leftarrow \mathcal{A}(\mathbf{com}, M, \mathbf{resp})] - \mathcal{P}[(x, w) \leftarrow \mathcal{U}]| \\ & \leq |\text{negl}_1(\Lambda)^2 - \text{negl}_2(\Lambda)^2| \leq \text{negl}(\Lambda). \end{aligned}$$

□

Beyond our assumption that the input pair of KEM and DS are themselves secure, the CPA makes no atypical revelation (by prover or verifier) of information for either involved primitive.

The proposed single-execution CPA has potential applications for quantum safe identity authentication, as in typical client-server access. Here client IDs would be associated with their public DS verification key $pk_{\mathbf{ds}}$, whereas their private keys can be generated in any number of meaningful ways that involve their private information. Among these applications, blockchains and digital currency take precedence.

Yet simple modifications to the proposed single-execution CPA can alter its scope of use, and moreover without compromising security. We subsequently outline two such variants: the first eliminates the shared static $pk_{\mathbf{kem}}$, while the second is non-interactive.

3.2.2 First variant: CPA with a dynamic public encapsulation key

In Section 3.1, we discussed the CPA with a private key pair $(sk_{\mathbf{kem}}, sk_{\mathbf{ds}})$ held exclusively by the prover, with its related public key pair $(pk_{\mathbf{kem}}, pk_{\mathbf{ds}})$ being available to all verifiers. However, static sharing of $pk_{\mathbf{kem}}$ and $pk_{\mathbf{ds}}$ requires doubled memory for verifiers. Hence to lighten the server’s storage burdens we adjust the CPA protocol to begin sharing only the public signature verification key, $pk_{\mathbf{ds}}$.

More explicitly, after randomly generating and storing $(sk_{\mathbf{kem}}, sk_{\mathbf{ds}})$, whenever a commitment is required the prover dynamically computes $pk_{\mathbf{kem}}$ which is sent to the verifier as $\mathbf{Sign}(sk_{\mathbf{ds}}, pk_{\mathbf{kem}})$. From which point the original CPA definition requires only simple changes for the protocol to terminate successfully.

We now formally define this variant, named CPA-DE for “Dynamic Encapsulation (Key)”.

Definition 3.4. *CPA-DE*

$$\begin{aligned}
\mathcal{R}_{CPA-DE} = \{ & (x, w) : x = (pk_{\mathbf{kem}}, pk_{\mathbf{ds}}), w = (sk_{\mathbf{kem}}, sk_{\mathbf{ds}}), \\
& pk_p = \mathbf{pkgen}_p(sk_p), sk_p \in \mathcal{SK}_p, \\
& p \in \{\mathbf{kem}, \mathbf{ds}\}\}, \\
\mathbf{com} \leftarrow \mathbf{P1}(x, w) = & (pk_{\mathbf{kem}}, \mathbf{Sign}(sk_{\mathbf{ds}}, pk_{\mathbf{kem}}), pk_{\mathbf{ds}}), \\
\mathbf{chall} \leftarrow \mathbf{V1}(\mathbf{com}) = & \mathbf{Encap}(\mathbf{com}_1, M) \text{ if and only if} \\
& \mathbf{SigVerify}(\mathbf{com}_3, \mathbf{com}_1, \mathbf{com}_2) = 1, \\
\mathbf{resp} \leftarrow \mathbf{P2}(\mathbf{chall}) = & \mathbf{Sign}(sk_{\mathbf{ds}}, \mathbf{Decap}(sk_{\mathbf{kem}}, M)), \\
\mathbf{V2}(\mathbf{com}, M, \mathbf{resp}) = & 1 \text{ if and only if } \mathbf{SigVerify}(\mathbf{com}_2, M, \mathbf{resp}) = \\
1. &
\end{aligned}$$

For formal clarity we note that, to expect success, we must have

$$\begin{aligned}
& \mathbf{SigVerify}(\mathbf{com}_3, \mathbf{com}_1, \mathbf{com}_2) \\
& = \mathbf{SigVerify}(pk_{\mathbf{ds}}, pk_{\mathbf{kem}}, \mathbf{Sign}(sk_{\mathbf{ds}}, pk_{\mathbf{kem}}))
\end{aligned}$$

with the above function's variables in proper order. We also observe this protocol's adherence to the ZKA security criteria is essentially unchanged, yet CPA-DE has implementation level advantages over the basic CPA:

1. The public component $pk_{\mathbf{kem}}$ will not require indefinite storage but can be generated at need without changing basic commitment structures, since signature verification and encapsulation keys relate to one unique private key pair $(sk_{\mathbf{kem}}, sk_{\mathbf{ds}})$. If a dishonest prover merely pretends to hold either real private key they yet fail by **P2**.
2. The verifier has probability $1 - \text{negl}(\Lambda)$ of rejecting a dishonest prover after merely receiving **com**. Indeed, $\mathbf{SigVerify}(\mathbf{com}_3, \mathbf{com}_1, \mathbf{com}_2) \neq 1$ with overwhelming probability if no valid $sk_{\mathbf{ds}}$ is held for the given $pk_{\mathbf{ds}} = \mathbf{com}_3$.

These optimizations enhance the efficiency of our original CPA's client-server authentication.

3.2.3 Second variant: Non-interactive CPA

A non-interactive CPA is highly desirable for such applications as proof-of-work systems. Building upon the first variant in Section 3.2.2 our second variant, CPA-NI, offers enhanced efficiency and flexibility.

With CPA-NI, the prover randomly generates a standing, public challenge, to be subsequently inspected for acceptance (or rejection) by potential

verifiers. It relies fully on the strength of a DS algorithm, making no use of KEMs.

It repeats the CPA-DE protocol though, rather than using pk_{kem} within its commitment, instead supplies $M \leftarrow \{0, 1\}^\Lambda$, and necessarily terminates after CPA-DE's use of **V1** (here renamed **V**).

This variant eliminates the several step back-and-forth of the previous CPA protocols, and now employs only two functions: proof **P**, and verification **V**. Yet as we may consider **P2** and **V2** to be empty or zero functions that vacuously accept, CPA-NI remains a ZKA, satisfying all attendant security criteria (assuming a valid post-quantum DS). It has the following formal description.

Definition 3.5. *CPA-NI*

$$\begin{aligned} \mathcal{R}_{CPA-NI} &= \{(x, w) : x = \text{pkgen}_{ds}(w), w \in \mathcal{SK}_{ds}\}, \\ \text{com} &\leftarrow \mathbf{P}(x, w) = (M, \text{Sign}(sk_{ds}, M), pk_{ds}), \\ \mathbf{V}(\text{com}) &= 1 \text{ if and only if } \text{SigVerify}(\text{com}_3, \text{com}_1, \text{com}_2) = 1. \end{aligned}$$

Again, one compelling application of CPA-NI is in blockchain networks, where transaction validations often require proof of ownership or authorization without direct interaction between parties. Our variant can be used here by embedding a newly generated transaction within the proof request, together with a signature of the transaction signed by the sender using sk_{ds} . In validating this transaction, a verifier simply uses the prover's related and statically stored pk_{ds} . The embedded proof would seamlessly ensure the transaction's validity and authenticity within the decentralized environment of blockchains.

Similarly, CPA-NI offers further benefits for digital currency systems, and particularly with batch verification of transactions. By incorporating the CPA-NI protocol directly into each transaction, these systems can simultaneously validate multiple transactions, reducing processing overhead and enhancing network scalability. Each transaction would be accompanied by a valid proof of ownership or authorization, signed with a genuine DS private key, thereby reinforcing the security and integrity of the overall system.

4 Post-Quantum Implementations

We use the accepted standards for the ML-KEM [2] and the ML-DSA Scheme [19] to instantiate all three of our CPAs as the ML-CPA, ML-CPA-

Table 1 ML-CPA/ML-CPA-DE specific memory requirements

NIST Security Level	Prover Storage (bytes)	Communications (bytes)
I	4192	5300
III	6432	7533
V	8064	10355

Table 2 ML-CPA/ML-CPA-DE vs LRMC Σ -protocol total memory requirements

NIST Security Level	ML-CPA/CPA-DE (bytes)	LRMC Sigma-Protocol (bytes)
I	9492	24652
III	13965	55376
V	18419	99488

Table 3 ML-CPA-NI specific memory requirements

NIST Security Level	Prover Storage (bytes)	Communications (bytes)
I	2560	3732
III	4032	5261
V	4896	7219

DE and ML-CPA-NI. Memory requirements are computable from the NIST specification documents' parameter sets.

For Table 1, we adjudge communications costs of our CPAs by the sum of memory requirements for all exchanged items, being **com**, **chall**, and **resp**, which relate as (pk_{ds}, pk_{kem}) , an encapsulated message (ciphertext) and signature, respectively. Whereas, the prover's storage simply represents the remaining memory required by both secret keys (sk_{ds}, sk_{kem}) .

For Table 2, we merely sum the memory requirements of both columns for the ML-CPA/CPA-DE. Regarding the LRMC [40], their total memory requirements are likewise the sum of communications costs (directly computed using their own formula) with the public key sizes. We observe that public key sizes are almost trivial relative to communications costs, with public keys of 44, 80 and 96 bytes for the respective three cases of NIST levels I, III and V. Moreover, though formulae are only given for the DS scheme developed upon their Σ -protocol, the Fiat-Shamir Transform [23] entails that memory costs are identical between the two for obtaining post-quantum security. Plainly evident, the ML-CPA and ML-CPA-DE provide significant improvements for memory in all cases.

Lastly, the information from Table 3 is taken only from the ML-DSA specification document, as sk_{ds} for the prover storage, summing pk_{ds} and signature sizes as the communications costs.

We emphasize again that, given our CPAs are adaptable to any KEM/DS pair, this enables circumstance-specific choices and further improvements in use of computational resources as cryptographic primitives evolve.

While our implementations above use ML-KEM and ML-DSA, the CPA framework is general and can accommodate any compact post-quantum cryptographic scheme. In particular, emerging schemes such as HPPK cryptography [28] offer a promising option for scenarios requiring minimal memory and communication overhead, such as constrained devices or blockchain systems.

5 Conclusion

This paper presents three varieties of quantum-safe Cryptographic Primitive Argument (CPA) schemes, instantiated using post-quantum cryptographic primitives, and suitable for applications such as authentication, blockchain, and digital currency systems.

The key contribution lies in the design of a unified framework for Computational Zero-Knowledge Proofs (CZKPs) or Zero-knowledge Arguments (ZKAs), integrating well-studied KEM and DS schemes to achieve strong post-quantum security guarantees. The three variants—CPA, CPA-DE, and CPA-NI—demonstrate flexibility in communication patterns, storage requirements, and interactivity, accommodating diverse application scenarios, including constrained devices and high-throughput blockchain environments.

Our analysis shows that, with careful selection of PQC primitives, these ZKA protocols can provide both security and efficiency. Moreover, the CPA framework is general and adaptable, allowing integration with any compact post-quantum scheme, including emerging approaches such as HPPK cryptography, for resource-sensitive implementations.

Future work will focus on optimizing these protocols further, exploring additional PQC instantiations, developing practical software implementations, evaluating real-world performance, and extending applications of quantum-safe ZKAs to broader cryptographic and distributed systems contexts.

Acknowledgments

D.J. was funded by a Mitacs Accelerate internship. D.P. was funded by NSERC of Canada, grant number RPGIN-2024-05341.

References

- [1] Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loïc Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Edoardo Persichetti, and Gilles Zémor. Hamming Quasi-Cyclic (HQC), November 2017. Submission to the NIST post quantum standardization process. 2017.
- [2] Gorjan Alagic, Quynh Dang, Dustin Moody, Angela Robinson, Hamilton Silberg, and Daniel Smith-Tone. Module-lattice-based key-encapsulation mechanism standard, 2024-08-13 04:08:00 2024.
- [3] S. Almuhammadi and C. Neuman. Security and privacy using one-round zero-knowledge proofs. In *Seventh IEEE International Conference on E-Commerce Technology (CEC'05)*, pages 435–438, 2005.
- [4] Jean-Philippe Aumasson, Daniel J. Bernstein, Ward Beullens, Christoph Dobraunig, Maria Eichlseder, Scott Fluhrer, Stefan-Lukas Gazdag, Andreas Hülsing, Panos Kampanakis, Stefan Kölbl, Tanja Lange, Martin M. Lauridsen, Florian Mendel, Ruben Niederhagen, Christian Rechberger, Joost Rijneveld, Peter Schwabe, and Bas Westerbaan. SPHINCS+. Tech. rep. available at <https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions>, 2020. National Institute of Standards and Technology.
- [5] Roberto Avanzi, Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, Gregor Seiler, and Stehlé Damien. CRYSTALS-KYBER. Technical report available at <https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions>, 2020. National Institute of Standards and Technology.
- [6] Elaine Barker, Don Johnson, and Miles Smid. Recommendation for pair-wise key establishment using discrete logarithm cryptography (revised), 2007-03-14 2007.
- [7] Claudia Bartoli and Ignacio Cascudo. On sigma-protocols and (packed) black-box secret sharing schemes. In Qiang Tang and Vanessa Teague, editors, *Public-Key Cryptography – PKC 2024*, pages 426–457, Cham, 2024. Springer Nature Switzerland.

- [8] Andrea Basso, Giulio Codogni, Deirdre Connolly, Luca de Feo, Tako Bories Fouotsa, Guido Maria Lido, Travis Morrison, Lorenz Panny, Sikhar Patranabis, and Benjamin Wesolowski. Supersingular curves you can trust. *EUROCRYPT 2023*, LNCS 14005:405–437, 2023.
- [9] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Scalable, transparent, and post-quantum secure computational integrity. *Cryptology ePrint Archive*, 2018.
- [10] Eli Ben-Sasson, Dan Carmon, Swastik Kopparty, and David Levit. Scalable and transparent proofs over all large fields, via elliptic curves (ecfft part ii). *Cryptology ePrint Archive*, Paper 2022/1542, 2022.
- [11] E. Berlekamp, R. McEliece, and H. van Tilborg. On the inherent intractability of certain coding problems (corresp.). *IEEE Transactions on Information Theory*, 24(3):384–386, 1978.
- [12] Nir Bitansky, Ran Canetti, Alessandro Chiesa, and Eran Tromer. From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, ITCS '12*, page 326–349, New York, NY, USA, 2012. Association for Computing Machinery.
- [13] Manuel Blum, Paul Feldman, and Silvio Micali. Non-interactive zero-knowledge and its applications. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 103–112, New York, NY, USA, 1988. Association for Computing Machinery.
- [14] Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical hardness of learning with errors. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '13, page 575–584, New York, NY, USA, 2013. Association for Computing Machinery.
- [15] Stanislav Bulygin, Albrecht Petzoldt, and Johannes Buchmann. Towards provable security of the unbalanced oil and vinegar signature scheme under direct attacks. In Guang Gong and Kishan Chand Gupta, editors, *Progress in Cryptology - INDOCRYPT 2010*, pages 17–32, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- [16] Wouter Castryck and Thomas Decru. An efficient key recovery attack on sidh (preliminary version). *Cryptology ePrint Archive*, Paper 2022/975, 2022.
- [17] Melissa Chase, David Derler, Steven Goldfeder, Claudio Orlandi, Sebastian Ramacher, Christian Rechberger, Daniel Slamanig, and Greg Zaverucha. Post-quantum zero-knowledge and signatures from symmetric-key primitives. In *Proceedings of the 2017 ACM SIGSAC*

- Conference on Computer and Communications Security*, CCS '17, page 1825–1842, New York, NY, USA, 2017. Association for Computing Machinery.
- [18] Craig Costello, Patrick Longa, and Michael Naehrig. Efficient algorithms for supersingular isogeny diffie-hellman. *Cryptology ePrint Archive*, Paper 2016/413, 2016.
 - [19] Thinh Dang, Jacob Lichtinger, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, Ray Perlner, and Angela Robinson. Module-lattice-based digital signature standard, 2024-08-13 04:08:00 2024.
 - [20] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.
 - [21] Jintai Ding, Joshua Deaton, Kurt Schmidt, Vishakha, and Zheng Zhang. Cryptanalysis of the lifted unbalanced oil vinegar signature scheme. In *Annual International Cryptology Conference*, pages 279–298. Springer, 2020.
 - [22] Jintai Ding and Bo-Yin Yang. *Multivariate Public Key Cryptography*, pages 193–241. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
 - [23] Jelle Don, Serge Fehr, Christian Majenz, and Christian Schaffner. Security of the Fiat-Shamir transformation in the quantum random-oracle model. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 356–383, Cham, 2019. Springer International Publishing.
 - [24] Jens Ernstberger, Stefanos Chaliasos, Liyi Zhou, Philipp Jovanovic, and Arthur Gervais. Do you need a zero knowledge proof? *Cryptology ePrint Archive*, Paper 2024/050, 2024.
 - [25] Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Cryptology ePrint Archive*, Paper 2011/506, 2011.
 - [26] Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.
 - [27] S Goldwasser, S Micali, and C Rackoff. The knowledge complexity of interactive proof-systems. In *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*, STOC '85, page 291–304, New York, NY, USA, 1985. Association for Computing Machinery.
 - [28] Randy Kuang. Optimized HPPK cryptography for post-quantum security. *Cryptology ePrint Archive*, Paper 2025/1467, 2025.
 - [29] V Lyubashevsky, L Ducas, E Kiltz, T Lepoint, P Schwabe, G Seiler, D Stehlé, and S Bai. CRYSTALS-DILITHIUM. Technical report

- available at <https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions>, 2020. National Institute of Standards and Technology.
- [30] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, pages 1–23, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- [31] R. J. McEliece. A Public-Key Cryptosystem Based On Algebraic Coding Theory. *Deep Space Network Progress Report*, 44:114–116, January 1978.
- [32] NIST. Status report on the 3rd round of the NIST pqc cryptography standardization process. <https://csrc.nist.gov/publications/detail/nistir/8413/final>, July 2022.
- [33] Shien Jin Ong and Salil P. Vadhan. Zero knowledge and soundness are symmetric. In *Advances in Cryptology - EUROCRYPT 2007, 26th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Barcelona, Spain, May 20–24, 2007, Proceedings*, volume 4515 of *Springer, Lecture Notes in Computer Science*, pages 187–209, 2007.
- [34] T Prest, P-A Fouque, J Hoffstein, P Kirchner, V. Lyubashevsky, T Pornin, T Ricosset, G Seiler, W Whyte, and Z Zhang. FALCON. Tech. rep. available at <https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions>, 2020. National Institute of Standards and Technology.
- [35] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.
- [36] Damien Robert. Breaking sidh in polynomial time. *Cryptology ePrint Archive*, Paper 2022/1038, 2022.
- [37] Claus-Peter Schnorr. Efficient signature generation by smart cards. *Journal of Cryptology*, 4:161–174, 1991.
- [38] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.
- [39] Chengdong Tao, Adama Diene, Shaohua Tang, and Jintai Ding. Simple matrix scheme for encryption. In Philippe Gaborit, editor, *Post-Quantum Cryptography*, pages 231–242, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

- [40] Jiaming Wen, Houzhen Wang, and Huanguo Zhang. *Post-quantum Sigma Protocols and Signatures from Low-Rank Matrix Completions*, pages 186–206. Springer, Nature. 10 2023.
- [41] Lizhen Zhang, Shang Gao, and Bin Xiao. Lattice-based σ -protocols for polynomial relations with standard soundness. *Cryptology ePrint Archive*, Paper 2025/313, 2025.
- [42] Lu Zhou, Abebe Diro, Akanksha Saini, Shahriar Kaisar, and Pham Cong Hiep. Leveraging zero knowledge proofs for blockchain-based identity sharing: A survey of advancements, challenges and opportunities. *Journal of Information Security and Applications*, 80:103678, 2024.

Biographies



Randy Kuang is Co-Founder and Chief Scientist of Quantropi Inc., where he leads research and development in post-quantum cryptography and quantum-safe communications. He holds a Ph.D. in Atomic and Molecular Physics from Memorial University of Newfoundland. Prior to founding Quantropi, Dr. Kuang held a senior research position at Nortel Networks, where he worked on next-generation networking and security systems, and later served as Co-Founder and Chief Technology Officer at in Bay Technologies, leading the design and deployment of a successful cybersecurity platform. He is the inventor or co-inventor of more than 40 U.S. patents, with foundational contributions including the Quantum Permutation Pad (QPP) for quantum symmetric encryption, the Homomorphic/Multivariate Polynomial Public Key (HPPK/MPPK) framework for post-quantum key encapsulation and digital signatures, and the Quantum Encryption in Phase Space (QEPS) scheme for coherent optical communications. His work on algebraic cryptography and search-based security has been published in leading journals, and he serves on the editorial boards of EPJ Quantum Technology, Scientific Reports (Nature Portfolio), and Academia Quantum. He is widely recognised

for bridging rigorous theoretical foundations with deployable cryptographic systems.



Daniel Johnson holds a PhD in Applied Mathematics from Carleton University (2025) and is a post-quantum cryptology researcher currently working with 01 Quantum and the NC-CIPSeR lab at Carleton. His doctoral work produced strong results in the cryptanalysis of knapsack primitives — including the discovery of the Lattice Reconstitution Attack (LRA) which breaks the HPPK KEM in polylogarithmic time with deterministic success, but also an LRA-variant which breaks the 1978 Merkle-Hellman scheme with great efficiency in cubic polynomial time. He now focuses on defense of AI models using Fully Homomorphic Encryption (FHE), while also evolving attacks versus lattice-based primitives.



Daniel Panario was born in Uruguay. He studied Mathematics and Computer Science in Uruguay. He received a M.Sc. degree from the University of Sao Paulo, Brazil, and the Ph.D. degree from the University of Toronto, Canada. He is currently a Professor in mathematics with Carleton University, Ottawa, ON, Canada. His main research interests are in finite fields and their applications in information theory and communications, and in the analysis of algorithms.