Circle's Post-Quantum Security Roadmap

Securing blockchains, smart contracts, and digital assets for the quantum era

Mira Belenkiy[1], Duc V. Le[1], Gordon Liao[1], Vipin Singh Sehrawat[1], Dragos Rotaru[1], Sergey Gorbunov[1], Milap Sheth[1], Jay Logelin[1], Anthony De Abreu[1], and Dan Boneh[2]

[1] *Circle* [2] *Stanford University*

1 Introduction

Quantum computers fundamentally break the security foundations of modern blockchains by rendering the elliptic curve cryptography (ECC) they rely on obsolete. Any account that has ever signed a transaction has already revealed its public key, enabling a sufficiently capable quantum adversary to recover the corresponding private key. Unlike classical cryptographic failures, this is not a gradual erosion of security but a potential cliff event. Today, millions of funded addresses—including an estimated 14 million Bitcoin addresses—are exposed to this risk [54]. Beyond the blockchain itself, a vast layer of supporting security infrastructure rests on cryptographic assumptions that will no longer hold in a post-quantum world.

Circle's exposure to this transition is multi-faceted. As the developer of the Arc blockchain, Circle must evolve its entire development stack to be quantum-secure while ensuring a smooth transition for smart contract developers and account holders. As the operator of USDC smart contracts deployed across more than 30 blockchains, Circle must be prepared to support a diverse and evolving set of APIs as each chain addresses the quantum threat on its own timeline. Finally, as a blockchain service provider, Circle must assess and strengthen its internal infrastructure as well as its dependencies on external vendors.

1.0.1 The Signature Migration Challenge. The central challenge facing the crypto industry is the transition of transaction signatures from elliptic curve cryptography to post-quantum schemes. This shift is not merely a cryptographic upgrade, but a multi-dimensional systems problem.

At the protocol layer, larger post-quantum signatures place immediate pressure on blockchain throughput, increasing costs and impacting user experience. There is also a risk of fragmentation, as different blockchains adopt different signature schemes on different timelines. At the same time, post-quantum support for threshold signatures and MPC remains immature. Applications such as USDC and DeFi protocols must therefore operate securely across increasingly heterogeneous environments.

At the operational layer, the challenge centers on custody and key management. Every wallet will need to migrate its funds, including the vast amounts held in cold storage which must be moved securely. Organizations will need to upgrade their backend infrastructure and assess the post-quantum readiness of their vendors and cloud service providers.

At the ecosystem level, the lack of standardization introduces fragmentation risk. Developers seek interoperability across chains and tools. Hardware also complicates adoption: secure hardware wallets require multi-year development cycles, creating a lag between new cryptographic standards and real-world deployment. All this creates a fundamental tension—early adopters risk isolation on incompatible systems, while those who delay may miss critical migration windows. Navigating this transition requires technical leadership and thoughtful decision-making that balances ecosystem compatibility with the urgency of proactive migration.

1.0.2 Migrating Smart Contracts. Blockchains will need to provide smart contracts with native post-quantum signature verification. Even minimal support—such as a single post-quantum verification primitive—would allow developers to leverage account abstraction and begin transitioning users without waiting for full protocol redesigns. This is why Arc will support SLH-DSA.

At the same time, deployed smart contracts introduce structural constraints. Many rely on ECDSA-based primitives such as `ecrecover` for signature verification, which become insecure in the presence of quantum adversaries. While contracts that use upgradeable patterns (such as USDC) can be adapted, many others cannot be modified post-deployment, leaving portions of the ecosystem exposed. Addressing this class of risk requires solutions at the blockchain level. We describe Arc's approach to mitigating `ecrecover`-related vulnerabilities in Section 4.3.

1.0.3 The Risk of Rushing to Post-Quantum. The transition to post-quantum cryptography in the blockchain space is a complex process that requires both an engineering effort and user action. There is some concern that rushing the transition will result in a design flaw that will expose the system to a conventional attack or permanently locked assets. This is a bigger danger than the risk of an attack by a quantum computer whose time horizon is unknown.

To give a concrete example, consider an enterprise that manages its signing keys in a secure hardware, an HSM. The HSM is designed to ensure that an attacker who compromises the enterprise will not be able to steal the signing key in the clear. A common problem is that deployed HSM hardware does not support post-quantum signing. In a rush to post-quantum, the enterprise may decide that they have no choice but to allow their signing key to be exported from the HSM to a non-hardened CPU so that post-quantum signing can take place on the CPU. In doing so, the enterprise is more likely to lose the secret key due to a conventional attack than it is to lose the secret key due to a quantum attacker. Instead, we recommend that the enterprise lean on its HSM provider to add post-quantum support, and only transition once this change has been audited. This clearly takes time, but is a safer path than downgrading security.

We also note that the NIST process for "additional post-quantum digital signature schemes" is ongoing. This process will likely standardize additional schemes with new properties. A rush to post-quantum may end up enshrining a digital signature scheme that is sub-optimal.

While the transition to post-quantum blockchains needs to happen, we recommend that blockchains take their time and only

transition once all the components to support the transition are in place, without taking shortcuts that can harm security.

1.0.4 A Phased Strategy for Quantum Transition. Circle's approach to quantum security is structured in three phases. In the initial readiness phase Section 3, we focus on assessing and preparing for quantum risk across the organization. This includes a comprehensive inventory of vulnerabilities spanning our development stack, infrastructure, and third-party vendors. We prioritize protections against harvest-now-decrypt-later attacks, ensuring that sensitive data and user activity remain secure against future decryption. Arc will contain a privacy layer that uses an encrypted trusted execution environment with quantum-resistant privacy guarantees from day one. In addition, Arc will support SLH-DSA signature verification to enable developers and users to begin their migration.

During the transition phase Section 4, systems operate in dual mode for efficiency and compatibility. For example, USDC smart contracts will continue to support ECDSA-based interactions while also enabling post-quantum signatures. This hybrid approach allows gradual ecosystem migration while maintaining performance and usability. At the same time, new systems are designed for rapid switchover in the event of an accelerated threat timeline.

The final phase Section 5 completes the transition to fully post-quantum-secure systems. The timing of this switchover will depend on ecosystem readiness, regulatory requirements, and evolving quantum risk assessments. This phase may require deprecating support for non-quantum-secure blockchains and accounts. Circle is developing the necessary transition and recovery mechanisms to support users throughout this process, including addressing the legal and policy considerations associated with large-scale migration.

1.0.5 Account Recovery. Circle believes that migration policies must protect users and emphasize account recovery. Migration plans should distinguish between disabling unsafe cryptographic control and extinguishing an asset holder's economic interest. For Circle-issued assets, such as USDC, a cutoff for vulnerable signatures may be necessary to prevent quantum-enabled forgery. However, where technically feasible and legally supportable, assets locked after such a cutoff should remain subject to recovery if entitlement can be established through reliable evidence.

Regulatory guidance would be helpful on what notice issuers should provide before a post-quantum cutoff, what evidence should be sufficient to recover assets from vulnerable EOAs or smart contracts, how long locked assets should remain recoverable before being treated as abandoned or unclaimed, and how unclaimed-property, escheat, custody, redemption, sanctions, AML/CFT, estate, and court-order frameworks should apply. There is a meaningful window—potentially on the order of 5–10 years—to develop such regulatory guidance.

1.0.6 Algorithm Choices. Circle believes in following NIST standards and industry best-practices around security. We also prioritize privacy: you can always upgrade your signatures but you can never get your privacy back due to harvest-now-decrypt-later. For this reason, we chose to use hybrid encryption algorithms at NIST security level 3 such as TLS 1.3 with X25519MLKEM768 and HPKE with X-Wing. A hybrid algorithm combines two ciphers, and is secure as long as either the classic cipher or the quantum-safe cipher is secure. When we picked the SLH-DSA-SHA2-128s signature, we still chose the more conservative approach based on well-studied hash functions, but we optimized for efficiency in choosing level 1 parameters.

1.0.7 Outlook. Quantum risk presents a serious and systemic challenge to blockchain infrastructure, but it is one that can be managed with deliberate, phased action. Rather than waiting for perfect solutions or complete standardization, Circle's strategy emphasizes technical leadership, incremental upgrades, and operational readiness. By preparing early and designing for flexibility, we aim to ensure that users, developers, and assets remain secure throughout the transition to a post-quantum world.

2 Background

As Circle has begun conducting its quantum risk assessment and identifying potential threats, we have found that new vulnerabilities and unexpected dependencies continue to emerge. This highlights the complexity of the problem and underscores that fully addressing quantum risk will require coordinated effort across the industry. For example, the use of ecrecover in deployed smart contracts presents a significant vulnerability that has not yet been widely discussed. For this reason, we believe it is important to clearly articulate the quantum threat landscape and document currently known vulnerabilities. We encourage even industry experts to at least skim the title headings of this section.

2.1 Cryptographic Impact of Quantum Computers

A machine is cryptographically relevant when it can run attacks that break deployed schemes at relevant parameters. A CRQC needs fault-tolerant *logical* qubits. Due to high error rates, each logical qubit requires multiple physical qubits. In some cases, the logical qubit must support non-Clifford operations (T-gates), a higher bar than many demos. There are varying estimates on when a CRQC may emerge.

Shor's Algorithm. Shor's algorithm [58] solves integer factorization and discrete logarithms in polynomial time on a fault-tolerant quantum computer. Shor's algorithm breaks all ECC, DH, and RSA cryptography.

Recent estimates show that running Shor's algorithm on 256-bit ECC curves used in Bitcoin and Ethereum requires roughly 1,200–1,450 [8] logical qubits. For comparison Quantinuum's Helios H2 supports 48 logical qubits.

Grover's Algorithm. Grover's algorithm [31] gives a quadratic speedup for unstructured search. In practice oracle and fault-tolerance costs limit the advantage [35], and NIST's PQ FAQ treats AES-128 as secure for the foreseeable future [46]. Standard symmetric ciphers and mainstream hashes (SHA-256, SHA-3, Keccak-256, Blake2/3) are also considered secure against Grover.

2.2 Vulnerable Public Key Cryptography

The quantum threat only affects public-key cryptography. Symmetric schemes are not currently believed to be vulnerable to quantum attacks.

Table 1: Quantum impact on cryptographic primitives.

Primitive	Examples of Blockchain Use	PQ Status
ECDSA / Ed25519 / BLS	Tx sigs, consensus	Broken (Shor)
ECDH / DH / ElGamal	Key exchange, KEM, privacy	Broken (Shor)
EC-pairing SNARKs	Groth16, PlonK, Halo2	Broken (Shor)
Pedersen / ElGamal Encryption	Zcash, Bulletproofs, conf. tokens	Broken (Shor)
AES / SHA-256 / Keccak	Symmetric enc., hashing	Safe
X-Wing / X25519MLKEM768	key encapsulation	Safe
STARK (FRI/STIR/WHIR) [5, 6, 13]	ZK proofs, Merkle trees	Safe

Digital Signatures. Blockchains use digital signatures to authorize transactions. Proof-of-stake and proof-of-authority also use them to secure the consensus protocol. All common elliptic curve signatures ECDSA, Ed25519, and BLS are vulnerable to quantum algorithms.
Key Encapsulation Mechanisms. Blockchain protocols rely on quantum-vulnerable ECDH for key exchange. Peer-to-peer networking stacks (libp2p, RLPx, BIP 324) use it to establish session keys between nodes; client–node communication rides on JSON-RPC over TLS, which also negotiates session keys via ECDH; and privacy constructions such as Zcash-style encrypted notes and Monero-style stealth addresses derive per-transaction secrets from ECDH as well.
Public Key Encryption. Public-key encryption is used less often than signatures or KEMs on blockchains, but it does appear. Confidential-token designs typically rely on quantum-vulnerable ElGamal-style encryption over an elliptic-curve group.
Zero-Knowledge Proof Systems. Blockchains use zero-knowledge proofs for scaling (rollups), privacy, threshold signing, and other multi-party computations. Two families are deployed today. zk-SNARKs (e.g., Groth16, PlonK, Halo2) produce short proofs but derive their security from elliptic-curve assumptions and are vulnerable to a quantum adversary. zkSTARKs such as FRI [13], STIR [5], and WHIR [6] rely only on hash functions; their proofs are larger but remain secure against quantum attack.
Other Building Blocks. Other vulnerable building blocks include Pedersen-style commitments verifiable random function, and discrete-log-based linkable ring signatures.
Summary of Impact. Table 1 summarizes the impact of quantum computing on blockchain cryptography.

2.3 Quantum Threats Across the Blockchain Stack

The blockchain stack can be viewed a composition of three layers:

- *Application:* signing, contracts, custody, privacy (wallets, EOAs, encrypted on-chain data).
- *Consensus:* proposals, attestations, finality, history (validator keys, BLS, etc.).
- *Network:* inter-node protocols (libp2p, RLPx) and JSON-RPC over TLS—ECDH for session keys.

Application Layer Attacks.

- **(A1) At-rest forgery.** A CRQC that sees an account's public key can recover the private key and forge arbitrary signatures against it. On EVM chains, every account reveals its public key the first time it broadcasts a transaction, so every address that has ever signed is exposed—including token transfers, EIP-712 permits, and any other signature a smart contract checks through ecrecover. The same threat extends to pairing-based SNARKs with trusted setups: a CRQC can recover the trapdoors used during setup and forge new proofs. Stablecoin minting keys are a narrower but higher-impact target since compromising a single key lets an attacker issue unbacked supply.
- **(A2) Retroactive privacy loss.** Harvest-now-decrypt-later exposure is the greatest risk for blockchain cryptography today. Encrypted on-chain data, shielded notes, and confidential tokens are vulnerable to a quantum attack on ECDH KEM and cryptographic commitments.

Consensus Layer Attacks.

- **(A3) Real-time consensus disruption.** A CRQC can recover validator signing keys (ECDSA, Ed25519, BLS) from on-chain signatures, enabling equivocation, double-signing, and censorship without slashing deterrence since the attacker holds no stake. For chains that use ECC-based verifiable random functions (VRFs) for leader election or committee sampling—e.g., Algorand, Ouroboros, and some Cosmos chains—the VRF itself will become predictable, letting the attacker grind identities to bias proposer selection; Tendermint-family BFT consensus (including Arc's Malachite) selects proposers by round-robin and is not vulnerable to this variant. Since an attacker can precompute private keys, real-time disruption can occur even within subsecond consensus protocols.
- **(A4) History Rewrite.** An attacker can use compromised validator keys to forge the blockchain history for a proof-of-stake chain. This attack is more likely to succeed against nodes that were offline and missed a state checkpoint [7].

Network Layer Attacks.

- **(A5) P2P session compromise.** Breaking ECDH in p2p communication protocols like libp2p/Noise enables traffic analysis, impersonation, and eclipse attacks.
- **(A6) RPC interception.** Breaking the TLS ECDH handshake for JSON-RPC calls would expose queries and submitted transactions.
- **(A7) On-spend attacks.** Mempool-visible signatures may allow key recovery before confirmation. This would be a race condition between the attacker and the user attempting to use key-rotation to avoid migrating to a post-quantum transaction signature scheme. Fast-clock CRQCs could complete attacks in seconds–minutes [8]; slow-clock machines are limited to longer-window at-rest scenarios.

Table 2 summarizes threats by layer.

2.4 Regulatory and Standardization Context

Governments and bodies have issued PQ migration guidance; some predates NIST FIPS 203–205 [43–45]. See Table 3 for a brief survey.

Binding deadlines cluster around 2030–2035 in several jurisdictions; we believe this is a reasonable timeline and safely ahead of a plausible quantum computing breakthrough.

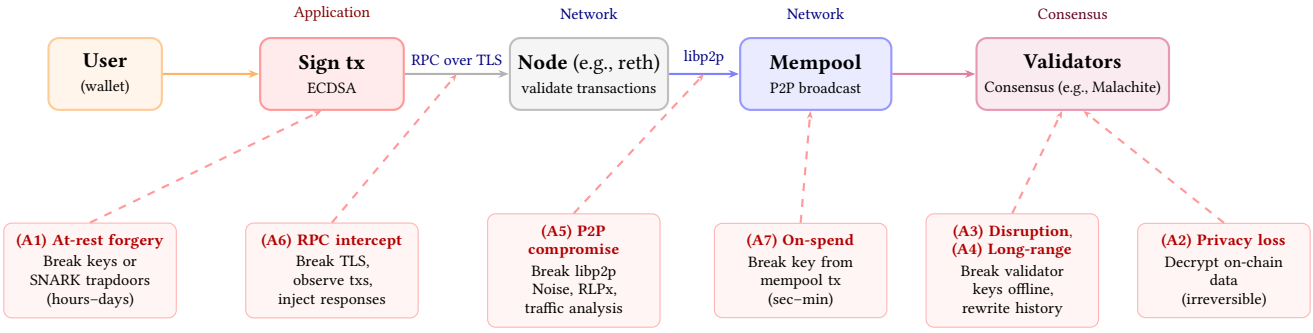

Figure 1: Transaction lifecycle and quantum attacks. *Top:* A user submits a transaction to the network. *Bottom:* Seven quantum attacks targeting different stages of the transaction lifecycle.

Table 2: Quantum threats by blockchain layer.

Layer	Attack	Time Constraint
Application	At-rest forgery (public key, trapdoor)	Offline
Application	Retroactive privacy loss	Offline (irreversible)
Consensus	Long-range / posterior	Offline
Consensus	Real-time disruption	Offline
Network	P2P session compromise	Offline
Network	RPC interception	Offline
Network	On-spend (mempool)	Seconds–min.

2.5 Quantum Safe Building Blocks

We review some post-quantum secure building blocks. These may be unfamiliar to some readers, so we thought it would be helpful to introduce them in one place.

Post-quantum cryptographic primitives.

- *ML-KEM* [43] (FIPS 203) is a lattice-based key encapsulation mechanism derived from CRYSTALS-Kyber, with security based on the module learning-with-errors problem. ML-KEM-768 is the most widely used in the industry.

Table 3: PQ migration mandates, standards, and guidance documents.

Organization	Document / Regulation	Deadline	Ref.
Government			
EU	PQC Roadmap (2024/1101)	2030	[27]
United States	NSM-10 / M-23-02	2035	[47, 59]
Canada (CCCS)	PQC Migration Roadmap	2035	[20]
Germany (BSI)	PQ recommendations	2030	[19]
UK (NCSC)	Next Steps in Preparing for PQC	Advisory	[41]
Standards bodies & agencies			
NIST	CSWP migration roadmap	Advisory	[42]
ETSI	Quantum-safe whitepaper	Advisory	[28]
ENISA	PQC mitigation report	Advisory	[29]
GSM Association	Telco PQ impact assessment	Advisory	[32]
Industry & research			
World Economic Forum	Quantum-secure economy	Advisory	[60]
SandboxAQ / Google	Nature paper	Advisory	[36]
PQShield	Quantum threat series	Advisory	[50–52]
Fraunhofer SIT	Practical PQC	Advisory	[30]
CWI/TNO/AIVD (NL)	PQ migration handbook	Advisory	[24]
CSIRO (Australia)	PQC whitepaper	Advisory	[23]

- *X-Wing* [10] is a hybrid KEM construction, under IETF CFRG review, that combines X25519 with ML-KEM-768. The hybrid design means confidentiality survives a cryptanalytic break in either component alone. X-Wing is safe for any general purpose use.
- *X25519MLKEM768* [37] is also a hybrid KEM construction that combines X25519 with ML-KEM-768. It is specifically designed for TLS 1.3 (general purpose encryption should use X-Wing). It is under IETF CFRG review and is becoming the industry standard.
- *HPKE* [12] (Hybrid Public Key Encryption, RFC 9180) is the IETF-standardized framework for public key encryption. It is post-quantum secure when it uses X-Wing for its KEM.
- *Falcon* [53] is the shortest of the post-quantum signatures NIST has selected for standardization; Falcon-512 signatures are encoding-dependent and commonly reported around 666 bytes. It will eventually be published as FN-DSA.
- *ML-DSA* [44] (FIPS 204) is a lattice-based digital signature scheme derived from CRYSTALS-Dilithium, with security based on module-LWE and module-SIS. ML-DSA-44 uses 2420-byte signatures and comparatively fast verification.
- *SLH-DSA* [45] (FIPS 205) is a stateless hash-based signature scheme constructed from Lamport/WOTS$^+$ one-time signatures stitched together through Merkle trees (the SPHINCS$^+$ design). Its security rests only on the underlying hash function, making the security assumptions the most conservative among the post-quantum standards; the cost is larger 7856-byte signatures and slower signing.
- *Hybrid signatures* [15–17] combine a classical signature (e.g., ECDSA) with a post-quantum signature (e.g., ML-DSA or SLH-DSA) so that forgery requires breaking both schemes. Signatures can be combined using concatenation, nesting, or more complex entanglement through a shared Fiat–Shamir challenge. IETF PQUIP has an active draft classifying the design space [16].
- *zkSTARKs* are transparent proof systems built on low-degree testing protocols such as FRI [13], *STIR* [5], and *WHIR* [6]. Their security rests only on collision-resistant hash functions, so they require no trusted setup and remain secure against a quantum adversary; the resulting proofs are roughly an order of magnitude larger than pairing-based SNARKs.

Trusted execution environments (TEE). A *trusted execution environment* (TEE) is an isolated region of a processor in which code and data are protected from the host operating system. TEEs can provide a signed attestation document proving that they are running specific code. Examples include Intel SGX [33], Intel TDX [34], AMD SEV-SNP [1], ARM TrustZone [4], and AWS Nitro [2]

3 Quantum Ready Now

Circle's focus today is to protect user privacy against harvest-now-decrypt-later attacks and to begin deploying the necessary features to allow USDC users to transition to post-quantum wallets.

3.1 Post-Quantum Signature

Arc will deploy a precompiled post-quantum signature verifier on mainnet (SLH-DSA-SHA2-128s) so smart accounts can validate post-quantum signatures on-chain. SLH-DSA's security relies solely on the underlying hash function SHA2, making it a conservative choice from the new NIST post-quantum signature suite. We chose the SHA2 variant rather than SHAKE because EVM exposes a SHA2 precompile. Other blockchains are converging on the same choice: Aptos has proposed SLH-DSA-SHA2-128s as the post-quantum signature for its accounts [3].

In Section 3.4, we discuss several techniques developers can use today to leverage native ARC SLH-DSA-SHA2-128s verification to transition their accounts to post-quantum signatures.

Native Arc transaction signatures will remain ECDSA-only for the near future. There are many good reasons to wait since the quantum-threat is not imminent: There is a high performance overhead caused by post-quantum signature length and verification time that will significantly affect transaction throughput. We are also waiting for the pending Falcon standardization into FN-DSA, which will be significantly shorter and faster than other candidates. Finally, many blockchain users rely on hardware wallets, and we want to ensure there is sufficient support for whatever algorithm we pick. We will continue to monitor developments in the industry.

There are two promising techniques for increasing the throughput of signature verification: batch verification and fraud-proofs.

Batch Verification. STARKs can be used to verify all transaction signatures in a block in one go. A single proposer would prepare the proof, and the other validators would check it. Bundlers can also take advantage of STARKs to batch signature verification for smart accounts.

Fraud Proofs. Both hash-based (SLH-DSA) and some lattice-based schemes decompose into independently checkable components. A fraud proof only needs to show that a single component is incorrect, which is much faster to check than verifying the whole signature [18, 38, 57]. This property may provide an opportunity for more efficient optimistic verification. Validators verify signatures off-chain and attest to their validity when proposing a block. Cheating validators can be quickly caught and slashed, and the false transaction rejected.

3.2 Encrypted Memos

Arc allows users to attach memos to any transaction. Circle will provide developers with an SDK to make it easy to encrypt/decrypt memos with post-quantum X-Wing HPKE [11, 12].

HPKE encryption with X-Wing increases the size of a memo payload by an additional 1136 bytes. Arc memos are large enough to handle the overhead - effectively capped only by the block gas limit. The same is true for messages sent by appending calldata to EVM transactions. Other blockchains will need to increase their memo sizes to allow post-quantum public key encryption.

3.3 Private Trusted Execution Environment

Arc's privacy layer provides a confidential execution environment where transactions, contract state, and execution traces remain encrypted throughout their lifecycle. Users prepare standard EVM transactions signed with their existing signature keys, then encrypt them under the network public key before submission. Validators hold shares of a master secret key, reconstruct the key itself only within trusted execution environments (AWS Nitro Enclaves) and decrypt transactions only inside the enclave boundary. Execution produces encrypted state that is persisted to disk, with an encrypted state root included in each block header. The privacy layer returns transaction receipts encrypted with the transaction-sender's ephemeral public keys, provided as part of the transaction.

3.3.1 Private Wallets. The privacy layer allows developers to use their existing EVM wallets and tooling inside the trusted execution environment with full post-quantum privacy[1]. Transactions and account balances inside the privacy layer are completely private.

3.3.2 Private Tokens. The privacy layer provides post-quantum privacy for standard Solidity smart contracts. A standard ERC-20 contract deployed inside the privacy layer becomes a confidential token without modification: balances and transfers are hidden by the encrypted execution boundary. Since transactions – including the signature – are encrypted, this protects the privacy of the sender.

An ERC-20 token has some public functions such as getting others' balances `balanceOf(account)` and raises events such as `transfer(from, to, value)`. The privacy layer allows developers to apply an external access control policy to a standard Solidity smart contract. Public functions can be restricted to trusted callers or even blocked entirely. This ensures private tokens are truly private.

3.3.3 Post-Quantum Ciphersuite. The privacy layer uses only post-quantum secure encryption:

- *Communication.* The enclaves communicate using TLS 1.3 with X25519MLKEM768 [37].
- *Transaction encryption.* Senders encrypt transactions using post-quantum HPKE [12] with X-Wing [11].
- *Persistent state encryption.* Validators encrypt state to disk using AES-256-GCM-SIV with encryption keys derived from the master secret via `HKDF-SHA-256`.

[1] By *post-quantum privacy* we mean confidentiality that holds against an adversary equipped with a cryptographically relevant quantum computer—transaction content and contract state remain hidden even after a CRQC materializes, defeating harvest-now-decrypt-later attacks.

3.3.4 AWS Nitro. As of this writing, AWS KMS and Nitro [2] provide partial support for post-quantum algorithms. Most importantly, all communication is secured with post-quantum secure TLS. Circle is monitoring AWS's post-quantum roadmap for Nitro enclaves. The privacy layer will upgrade to post-quantum enclave attestation and certificates when these features become available.

3.4 Post-Quantum Account Security

Even though Arc does not currently support post-quantum transaction signatures, Arc users can still begin to migrate their accounts. We cover some of the options. Some are Arc specific, while others can be performed on other blockchains as soon as they offer at least one post-quantum signature verification function.

3.4.1 Account Abstraction. EIP-4337 account abstraction enables smart-contract wallets. The account needs to store its public key (or at least its `keccak` fingerprint on-chain), and use the `validateUserOp` interface from the EIP above to verify `SLH-DSA-SHA2-128s` signatures. Transaction calldata would contain the signature itself. If the account only stores the public key fingerprint, the transaction calldata needs to provide the full public key as well. This additional calldata is stored as part of the `UserOperation` payload.

Bundlers may continue to use ECDSA during the transition period, while the smart account validates a post-quantum signature inside `validateUserOp`.

3.4.2 Hash-and-Rotate. A hash-and-rotate key strategy can mitigate quantum exposure in ERC-4337 accounts. Instead of storing a public key on-chain, the account stores only a hash commitment to the current key. When submitting a `UserOperation`, the user includes the corresponding public key and signature. During `validateUserOp`, the contract verifies the signature against the committed hash and then updates the stored commitment to the hash of a next key, ensuring each key is used at most once.

This design minimizes the window during which a plaintext public key is exposed on the network—only between transaction broadcast and inclusion. An attacker would therefore need to break the underlying signature scheme and forge a competing `UserOperation` within that short interval, significantly raising the bar for practical quantum attacks.

Key derivation for such a scheme must be handled carefully. For example, BIP-44 and BIP-32 derive child keys from parent signing scalars, so compromise of any intermediate scalar enables derivation of all descendant keys. Moreover, intermediate public keys are often exposed in real-world usage [9], meaning a single signature can effectively reveal the entire subtree. Alternatives such as SLIP-0010, which avoid these structural weaknesses, are better suited for this setting.

3.4.3 Dual Key Derivation. Wallet providers can derive both an ECDSA key and a post-quantum key from the same BIP-39 mnemonic, making accounts "quantum-ready" by default. This option is controlled by wallet providers (e.g., Coinbase, MetaMask). It can be combined with account abstraction for a smooth migration path, or used as just-in-case recovery option.

3.4.4 Post-Quantum Public-Key Registry. An on-chain public key registry can record bindings between an account address, a post-quantum public key, and the corresponding signature scheme. Multiple schemes could coexist for a single address. As long as the registration occurs before the emergence of a cryptographically relevant quantum computer (CRQC), the security model is unchanged: only the legitimate owner can authorize the binding using their ECDSA key.

3.4.5 Frame Transactions. The EIP-8141 frame transaction proposal removes the requirement for ECDSA at the envelope layer: the transaction names the verification logic. For smart accounts, execution can invoke that verification within the contract code. Ethereum has not approved EIP-8141 yet; we hope that this or a similar solution is eventually adopted into the EVM.

4 Transition

The transition period will involve introducing post-quantum features while continuing to operate in a hybrid mode. The primary challenge is performing this migration before the full post-quantum infrastructure is in place. While Circle controls the Arc stack and the USDC smart contract, it must also account for the quantum roadmaps of its cloud service providers and other blockchain networks.

This creates an inherent trade-off between building proprietary solutions that are viable today and waiting to integrate with more mature, native infrastructure as it emerges. At present, there is no single standard that the industry is converging toward, which further complicates the development of interoperable, cross-chain solutions.

4.1 Cold Storage

Circle plans to migrate cold storage funds to multi-signature smart contracts. Because Circle operates across many blockchains, it must adopt a strategy that works across heterogeneous environments with differing transaction signature schemes. We anticipate a multi-year transition period during which chains will adopt a variety of post-quantum signatures, rather than converging on a single approach.

Today, most blockchain systems rely on ECDSA and Ed25519, but the post-quantum transition will significantly expand the range of supported algorithms. In addition to NIST standardized schemes, developers are exploring variants that incorporate alternative hash functions (e.g., Keccak-256 or SHAKE), further increasing fragmentation. This makes it challenging to build interoperable solutions today that will function consistently across chains in the future.

Against this backdrop, on-chain multi-signature wallets provide a practical and flexible approach. They can accommodate multiple signature schemes simultaneously and compose naturally with account abstraction, enabling Circle to begin strengthening cold storage security early in the transition. Our goal is to develop a solution that works across (almost) all supported chains.

There are also alternative mechanisms using MPC and threshold signature schemes [14, 21, 25, 56]. However, given the wide range of algorithms Circle would need to support, the design, implementation, and operational complexity of these approaches is prohibitive in the near term. We will continue to monitor developments in this

area, particularly for more practical off-chain threshold signing solutions.

4.2 USDC and Smart Contracts

Almost all Circle smart contracts use the upgradeable proxy pattern, which allows contract logic to be updated to support post-quantum signatures. Initially, upgraded contracts will operate in a dual mode, supporting both classical and post-quantum signatures. These transitional contracts will include a mechanism to disable classical signatures without requiring an additional upgrade.

Circle's USDC and supporting smart contracts rely on EIP-712 signatures to verify gasless transactions, such as EIP-3009 `transferWithAuthorization` and EIP-2612 permit. These implementations currently use ecrecover to validate ECDSA signatures. Supporting post-quantum signatures will require new verification patterns and potentially new contract interfaces.

Arc mainnet will support the post-quantum signature SLH-DSA-SHA2-128s. More broadly, we expect that blockchains will begin exposing native implementations of post-quantum signature schemes to smart contracts. Circle's preference is to rely on native support, rather than implementing ad hoc verification logic in EVM assembly, which is more complex and harder to maintain.

4.3 Fixing ecrecover

Arc will need to address the widespread use of ecrecover in deployed smart contracts. Many contracts rely on ecrecover to verify ECDSA signatures over calldata, enabling users to authorize transactions off-chain while a third party submits them on-chain and pays gas fees. This pattern is widely used in EIP-2612 `permit` and EIP-3009 `transferWithAuthorization`, and has been standardized through EIP-712, reflecting its growing importance in the ecosystem.

The ecrecover function is a precompile available on all EVM chains. It takes a keccak256 hash and an ECDSA signature as input, and returns the signer's address by recovering the corresponding public key. Smart contracts then verify that the recovered address matches the expected signer.

A key challenge is that most deployed smart contracts are immutable and cannot be upgraded to support new signature schemes. Blockchains have to choose between the disruption of disabling ecrecover, allowing it to continue operating insecurely, or building a post-quantum, semantically equivalent replacement for ecrecover at the protocol level.

This transition is non-trivial. First, post-quantum signatures are significantly larger than the fixed 65-byte ECDSA signatures expected by ecrecover. Second, unlike ECDSA, most post-quantum schemes do not support public key recovery from a signature, and their public keys are too large to be passed directly as inputs.

One promising option is to deploy a modified version of ecrecover that preserves the existing ABI while extending its semantics to support both ECDSA and `SLH-DSA-SHA2-128s`. This approach allows existing contracts to continue functioning without modification, while enabling externally owned accounts to transition to post-quantum signatures in a backward-compatible manner.

A hardfork would be necessary to modify the action of ecrecover to call a new override smart contract `ECRecoverOverride`. Users will be able to register their public keys and signatures.

Algorithm 1: post-quantum `ecrecover`

Input: message digest `bytes32 h`
version `uint8 v`
public key fingerprint `bytes32 r`
signature index `bytes32 s`
Output: `signer` on success, or `address(0)` on failure
constant $V \leftarrow 29$
if $v \neq V$ **then**
  return `address(0)`
end
(signer, pk) $\leftarrow$ LookupAccount(r)
if *signer* = `address(0)` **then**
  return `address(0)`
end
$\sigma \leftarrow$ LookupSignature(s)
if σ *is empty* **then**
  return `address(0)`
end
if ***not*** VerifyPQSig(h, σ, pk) **then**
  return `address(0)`
end
return *signer*

`ECRecoverOverride` would use the `bytes32 r` and `bytes32 s` values to look-up the public key and signature. It would verify the signature, and return the registered address if verification passes and `address(0)` if it fails. The version `uint8 v` can be used to identify which verification algorithm to run: ECDSA or post-quantum.

Pre-registering the signature in a separate transaction is inconvenient. Arc will allow EOAs to post multiple transactions in one single call to the `batch(calls[])` precompile. Signature registration can happen in the same transaction as the actual usage. Even more promising is the proposed EIP-8141 which would allow ecrecover read frame call data to get the signature.

We hope that the industry agrees upon one solution for all EVM chains.

4.4 Encrypted Mempool

Circle is investigating options for offering an encrypted mempool. An encrypted mempool would hide the transaction payloads even from validators until ordering. This would eliminate the residual on-spend risk from a malicious validator and prevent MEV-driven manipulation. An encrypted mempool would also reduce the attack window for accounts that use hash-and-rotate to protect their ECDSA public keys.

4.5 Key management and Inventory

On-chain keys are only one layer within a broader security architecture. Like other large enterprises, Circle relies on extensive backend controls to ensure safe and reliable operations. Upgrading cryptographic keys to post-quantum algorithms is therefore insufficient if the surrounding infrastructure continues to depend on classical security assumptions.

During the transition, Circle will need to:

(1) Inventory and upgrade its entire cryptographic stack;

(2) Assess the post-quantum readiness of cloud and infrastructure providers, and migrate where necessary;
(3) Systematically rotate keys and secrets.

The order of rotation is critical. If key A protects key B, which in turn protects key C, then A must be rotated before B, and B before C. Failing to do so could expose the system to adversaries employing *harvest now, decrypt later* attacks, where previously captured encrypted material becomes vulnerable once a CRQC materializes.

4.6 Libp2p

The `libp2p` [55] networking library is a quantum-risk dependency across many blockchain systems, including Arc. It provides a modular suite of protocols, including GossipSub for publish/subscribe messaging, Noise for secure transport, and request/response mechanisms. Together, these components enable peer discovery, connection management, gossip-based data dissemination, synchronization, and node identity.

At present, `libp2p` relies on classical cryptographic primitives, including X25519 for key agreement and Ed25519 for identity and message authentication. The `libp2p` 2025 annual report [39] indicates that support for post-quantum algorithms is under active exploration.

4.7 Chain History Integrity

Proof-of-stake and proof-of-authority blockchains are vulnerable to chain history forgery. A CRQC can derive *every* historical validator's private key from on-chain public keys. All past keys break simultaneously and the attacker can forge a history.

The solution is two-fold: upgrade the validators to use post-quantum signatures and create a post-quantum secured checkpoint of block state. It may also be helpful to revalidate the block history inside an archival node with post-quantum signatures. This can be done optimistically: one validator signs the entire history and the rest check and cosign the final result.

5 Switch

At some point, Circle will need to execute a full transition away from classical cryptography. Circle will provide our partners, customers, and developers with ample notice. We will coordinate breaking changes with the broader crypto ecosystem.

5.1 Hard Switch

Some changes will inevitably result in disabled accounts and stranded funds. Circle expects to provide partners, customers, developers, and the broader ecosystem with appropriate notice where feasible.

Delisting USDC. Circle may need to establish timelines for host blockchains to adopt post-quantum security measures, informed by evolving quantum-risk assessments, ecosystem readiness, and regulatory expectations. If a host blockchain does not achieve adequate post-quantum protections within those timelines, Circle may need to consider withdrawing support or pausing certain contract functionality to protect users from quantum-enabled forgery.

USDC and Other Circle Smart Contracts. Circle will update its smart contracts to remove support for ECDSA-based `ecrecover`. See Sections 4.2 and 4.3.

Transaction Signatures. Arc and USDC smart contracts will reject transactions signed with ECDSA. As a result, assets held in accounts that have not migrated will be frozen for user protection. This freeze should be understood as a protective control against quantum-enabled theft, not as a determination that the underlying asset holder has abandoned or forfeited any economic interest. See Section 5.2 for account recovery options.

Native `ecrecover`. Arc will transition `ecrecover` to a post-quantum-only mode (see Section 4.3). This change may break existing smart contracts and lead to stranded funds, with primary responsibility for remediation resting with contract owners.

Consensus Signatures. Arc will upgrade validator signatures to a post-quantum scheme. A final candidate has not yet been selected, pending the emergence of a suitable aggregate signature alternative to BLS. Recent work on hash-based multi-signatures [26] is promising but not yet production-ready. This upgrade is expected to be transparent to users, affecting only validators.

Infrastructure Upgrade. Circle will complete the migration of its internal infrastructure and rotate any remaining at-risk cryptographic keys.

5.2 Technical Approaches for Account Recovery

This section describes the technical approaches that may be used to rescue funds. Pre-registering keys or preparing dual-key seed phrases can help with recovery efforts §3.4.

Move to Arc. Circle may move a copy of frozen USDC contracts from non-compliant blockchains to Arc to manually manage fund recovery. Prior to the emergence of a CRQC, this contract can even be configured to allow automatic withdrawal to a post-quantum account with a verified signature.

Seed Phrases Recovery. Most wallets derive keys from a BIP-39 mnemonic, which is converted to a 512-bit seed. From this seed, two key hierarchy derivation paths are common: BIP-32 (used by ECDSA chains like Arc) and SLIP-0010 (used by EdDSA chains like Sui, Solana, and Near). For SLIP-0010, Baldimtsi et al. [9] show a practical ZK recovery circuit. For BIP-32, Osuntokun [49] recently proposed a zk-STARK recovery circuit that uses the seed (not the signing scalar) as witness.

TEE-Attested Recovery. A TEE can automate recovery without the need for STARKs. Arc already provides uses TEEs to implement the privacy layer §3.3. For example, a token issuer can deploy a private contract that verifies BIP-39 seeds, letting users securely prove ownership of their accounts and secure the funds with a post-quantum key. The privacy layer would execute the transaction within the TEE and issue an attested claim authorizing migration. The seed would never leave the enclave boundary and the attestation would be verifiable on-chain.

Off-chain recovery. Circle will assess whether an off-chain process could support recovery in limited circumstances. Depending

on the facts, such a process could take into account Circle-held customer records, custodial or exchange attestations, legal process, estate or corporate authority documents, or other independently verifiable evidence of entitlement. For self-custodied EOAs, KYC alone may not be sufficient to establish ownership, particularly where Circle does not have an independent basis to associate a claimant with the pseudonymous address.

5.3 Policy Implications.

The blockchain industry will need to address the challenge of funds stranded in EOA accounts and smart contracts. The central difficulty is identifying and verifying the rightful owner of such funds once a CRQC can forge signatures. We encourage regulators and lawmakers to provide clear guidance on how to treat assets that remain unclaimed after migration deadlines.

Circle believes that migration policies must protect users and emphasize account recovery. Some blockchains are considering issuing a flag-day beyond which accounts will be disabled. Bitcoin proposal BIP-361 [40] "restricts ECDSA/Schnorr spends by encumbering them with a quantum-safe rescue protocol." Under BIP-361, wallets that cannot use the rescue protocol (e.g. hardware wallets) will lose funds. Optimism has announced a hard January 2036 deadline for accounts to transfer, but has not described any rescue/recovery plans [48]. The Coinbase Quantum Computing & Blockchain position paper also contemplates a flag day [22]. Regulators should address asset recovery lest asset lock-up becomes the industry norm.

Circle's view is that a post-quantum flag day should distinguish between disabling unsafe cryptographic control and extinguishing an asset holder's economic interest. For Circle-issued assets, such as USDC, a cutoff for vulnerable signatures may be necessary to prevent quantum-enabled forgery. However, where technically feasible and legally supportable, assets locked after such a cutoff should remain subject to recovery if entitlement can be established through reliable evidence.

This distinction is especially important for self-custodied EOAs. Once a CRQC can forge classical signatures, ECDSA or EdDSA signatures may no longer provide reliable evidence of ownership. At the same time, KYC alone will generally be insufficient to establish ownership of a pseudonymous EOA, because Circle may not know who controls the address. Recovery standards may therefore need to differ across categories of assets and claimants, including Circle-hosted customers, institutional accounts, custodial or exchange-controlled addresses, self-custodied EOAs, smart-contract-held balances, estates, and legally restricted or sanctioned addresses.

Several existing doctrines provide useful reference points:

- **Escheat / Unclaimed Property.** Where property with an identifiable owner remains unclaimed, it is typically transferred to the state as custodian of last resort. This provides the closest analogue for handling unmigrated funds where ownership can still be established.
- **Bailment / Custody.** Where an issuer, custodian, or intermediary holds assets on behalf of users, it may owe duties of care to safeguard those assets and take reasonable steps to facilitate their return.
- **Abandonment.** Property may be claimed by another party only where the owner has clearly relinquished it. In practice, such intent is difficult to establish, particularly in cases involving lost keys, death, or incapacity.
- **Adverse Possession.** This doctrine has limited applicability, as blockchain "possession" is cryptographic rather than physical, but it may serve as a loose analogy in systems that allow durable reassignment of control over time.
- **Lost / Mislaid Property / Embedded Property / Treasure Trove.** Doctrines tied to physical location are unlikely to apply in blockchain contexts.

For fiat-backed stablecoins, token balances are best understood as claims against the issuer. Issuers or affiliated entities may also act in a custodial or quasi-custodial capacity, and are therefore subject to duties that emphasize safeguarding and recovery. When assets remain unclaimed after a reasonable migration period or become effectively unclaimable due to the inability to prove ownership, the doctrine of **escheat** provides the closest legal analogue.

Overall, these constraints suggest a policy approach focused on preservation, recovery, and clear ex ante rules. Regulatory guidance would be helpful on what notice issuers should provide before a post-quantum cutoff, what evidence should be sufficient to recover assets from vulnerable EOAs or smart contracts, how long locked assets should remain recoverable before being treated as abandoned or unclaimed, and how unclaimed-property, escheat, custody, redemption, sanctions, AML/CFT, estate, and court-order frameworks should apply. There is a meaningful window—potentially on the order of 5–10 years—to develop such regulatory guidance. In the long run, greater clarity will not only support quantum migration, but also improve outcomes for users who have lost access to their keys, are deceased, or are otherwise incapacitated. More broadly, it will help align crypto with mainstream financial frameworks.

6 Conclusion

Post-quantum migration for blockchains is not a single event, but a structured, multi-year effort whose urgency varies across the stack. Harvest-now-decrypt-later attacks require immediate protection for privacy-sensitive data, while data integrity and consensus-layer risks can be addressed incrementally as post-quantum infrastructure matures. This asymmetry in urgency motivates the three-phase architecture described in this paper.

Arc launches with native post-quantum protections at genesis, including an SLH-DSA precompile for quantum-safe smart-account wallets and the private execution environment, which removes public-key exposure for contracts and assets within its boundary. Together with encrypted memos and quantum-safe confidential tokens, these features mitigate privacy leakage and allow developers to independently begin their migration.

The transition phase addresses the more complex system-wide challenges. The goal is upgrade piece-meal, avoid disrupting existing applications, and operate in hybrid mode until the full quantum migration. All the pieces, including policies and procedures for account recovery, must be in place before the final switch to fully quantum-secure operation.

Table 4: Attacks and the phase(s) that address each in Arc's roadmap. Section labels in parentheses.

Attack	Now (§3)	Transition (§4)	Switch (§5)
(A1) At-rest forgery	SLH-DSA precompile (§3.1); Arc Privacy (§3.3); ERC-4337 smart accounts (§3.4.1); frame txs (§3.4.5); PQ public-key registry (§3.4.4)	Cold storage / multisig custody (§4.1); USDC upgrades (§4.2); `ecrecover` override (§4.3); key mgmt (§4.5)	Hard switch (§5.1); account rescue (§5.2)
(A2) Privacy loss	Encrypted memos (§3.2); Arc Privacy (§3.3)	Key management & rotation order (§4.5)	—
(A3) Consensus disruption	—	—	PQ consensus signatures (§5.1)
(A4) Long-range attack	—	PoW / social checkpointing (§4.7)	PQ consensus signatures (§5.1)
(A5) P2P compromise	—	libp2p / Noise PQ upgrade (§4.6)	—
(A6) RPC intercept	Arc Privacy RPC (§3.3)	TLS PQ upgrade (§4.6); encrypted mempool (§4.4)	—
(A7) On-spend	Hash-and-rotate (§3.4.2); Arc Privacy (§3.3)	Encrypted mempool (§4.4)	PQ transaction signatures (§5.1)

Post-quantum security is not achieved through a single upgrade, but through disciplined coordination across cryptography, infrastructure, and policy. Organizations that succeed will not be those that move fastest, but those that transition deliberately—preserving security, minimizing disruption, and maintaining user trust at every stage.

References

[1] Advanced Micro Devices. 2024. *AMD Secure Encrypted Virtualization (SEV).* https://www.amd.com/en/developer/sev.html

[2] Amazon Web Services. 2024. *AWS Nitro System.* https://aws.amazon.com/ec2/nitro/

[3] Aptos Foundation. 2024. *AIP-137: Post-Quantum Aptos Accounts via SLH-DSA-SHA2-128s.* https://github.com/aptos-foundation/AIPs/blob/main/aips/aip-137-post-quantum-aptos-accounts-via-slh-dsa-sha2-128s.md

[4] Arm Limited. 2024. *TrustZone for Cortex-A.* https://www.arm.com/technologies/trustzone-for-cortex-a

[5] Gal Arnon, Alessandro Chiesa, Giacomo Fenzi, and Eylon Yogev. 2024. STIR: Reed-Solomon Proximity Testing with Fewer Queries. In *Advances in Cryptology – CRYPTO 2024*, Leonid Reyzin and Douglas Stebila (Eds.). Springer Nature Switzerland, Cham, 380–413.

[6] Gal Arnon, Alessandro Chiesa, Giacomo Fenzi, and Eylon Yogev. 2025. WHIR: Reed–Solomon Proximity Testing with Super-Fast Verification. In *Advances in Cryptology – EUROCRYPT 2025*, Serge Fehr and Pierre-Alain Fouque (Eds.). Springer Nature Switzerland, Cham, 214–243.

[7] Sarah Azouvi, Christian Cachin, Duc V. Le, Marko Vukolić, and Luca Zanolini. 2023. Modeling Resources in Permissionless Longest-Chain Total-Order Broadcast. In *26th International Conference on Principles of Distributed Systems (OPODIS 2022) (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 253)*, Eshcar Hillel, Roberto Palmieri, and Etienne Rivière (Eds.). Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 19:1–19:23. doi:10.4230/LIPIcs.OPODIS.2022.19

[8] Ryan Babbush, Adam Zalcman, Craig Gidney, Michael Broughton, Tanuj Khattar, Hartmut Neven, Thiago Bergamaschi, Justin Drake, and Dan Boneh. 2026. Securing Elliptic Curve Cryptocurrencies against Quantum Vulnerabilities: Resource Estimates and Mitigations. (2026). Google Quantum AI, Ethereum Foundation, Stanford University. Dated March 30, 2026.

[9] Foteini Baldimtsi, Konstantinos Chalkias, Arnab Roy, and Mahdi Sedaghat. 2025. Post-Quantum Readiness in EdDSA Chains. Cryptology ePrint Archive, Paper 2025/1368. https://eprint.iacr.org/2025/1368

[10] Manuel Barbosa, Deirdre Connolly, João Diogo Duarte, Aaron Kaiser, Peter Schwabe, Karolin Varner, and Bas Westerbaan. 2024. X-Wing: The Hybrid KEM You've Been Looking For. *IACR Communications in Cryptology* 1, 1 (2024). doi:10.62056/a3qj89n4e

[11] Manuel Barbosa, Deirdre Connolly, João Diogo Duarte, Aaron Kaiser, Peter Schwabe, Karoline Varner, and Bas Westerbaan. 2024. X-Wing: The Hybrid KEM You've Been Looking For. IETF Internet-Draft, draft-irtf-cfrg-xwing. https://datatracker.ietf.org/doc/draft-irtf-cfrg-xwing/.

[12] Richard Barnes, Karthikeyan Bhargavan, Benjamin Lipp, and Christopher A. Wood. 2022. *Hybrid Public Key Encryption.* RFC 9180. IETF. doi:10.17487/RFC9180

[13] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. 2018. Fast Reed-Solomon Interactive Oracle Proofs of Proximity. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018) (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 107)*, Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella (Eds.). Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 14:1–14:17. doi:10.4230/LIPIcs.ICALP.2018.14

[14] Alexander Bienstock, Leo de Castro, Daniel Escudero, Antigoni Polychroniadou, and Akira Takahashi. 2025. Quorus: Efficient, Scalable Threshold ML-DSA Signatures from MPC. Cryptology ePrint Archive, Paper 2025/1163. https://eprint.iacr.org/2025/1163

[15] Nina Bindel and Britta Hale. 2023. A Note on Hybrid Signature Schemes. https://eprint.iacr.org/2023/423. Cryptology ePrint Archive, Paper 2023/423.

[16] Nina Bindel, Britta Hale, Deirdre Connolly, and Florence Driscoll. 2025. *Hybrid Signature Spectrums.* Internet-Draft draft-ietf-pquip-hybrid-signature-spectrums-07. IETF. https://datatracker.ietf.org/doc/draft-ietf-pquip-hybrid-signature-spectrums/07/

[17] Nina Bindel, Udyani Herath, Matthew McKague, and Douglas Stebila. 2017. Transitioning to a Quantum-Resistant Public Key Infrastructure. https://eprint.iacr.org/2017/460. Cryptology ePrint Archive, Paper 2017/460.

[18] Cecilia Boschini, Dario Fiore, and Elena Pagnin. 2022. Progressive and Efficient Verification for Digital Signatures. In *Applied Cryptography and Network Security (ACNS 2022)*. Springer-Verlag, Berlin, Heidelberg, 440–458. doi:10.1007/978-3-031-09234-3_22

[19] Bundesamt für Sicherheit in der Informationstechnik (BSI). 2025. *Quantum-Safe Cryptography — Fundamentals, Current Developments and Recommendations.* Technical Report. BSI. https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Brochure/quantum-safe-cryptography.pdf.

[20] Canadian Centre for Cyber Security. 2025. *Roadmap for the Migration to Post-Quantum Cryptography for the Government of Canada (ITSM.40.001).* Technical Report. CCCS. https://www.cyber.gc.ca/en/guidance/roadmap-migration-post-quantum-cryptography-government-canada-itsm40001.

[21] Sofía Celi, Rafaël del Pino, Thomas Espitau, Guilhem Niot, and Thomas Prest. 2026. Efficient Threshold ML-DSA. Cryptology ePrint Archive, Paper 2026/013. https://eprint.iacr.org/2026/013

[22] Coinbase Quantum Advisory Council. 2026. Quantum Computing and Blockchain: A Comprehensive Assessment of Risks and Migration Strategies. Technical report. https://assets.ctfassets.net/sygt3q11s4a9/6EjYavuGdtJDYCqaJrASj9/9f464a8bf26f44bd6c85710fe7e4a29f/Quantum_Computing_and_Blockchain_v10.3_15April2026.pdf Version 10.3, April 15, 2026.

[23] CSIRO Data61. 2022. *Post-Quantum Cryptography Whitepaper.* Technical Report. Commonwealth Scientific and Industrial Research Organisation. https://www.math.auckland.ac.nz/~sgal018/CSIRO-PQC-whitepaper.pdf.

[24] CWI and TNO and AIVD. 2023. *PQC Migration Handbook.* Technical Report. CWI/TNO/AIVD. https://www.marc-stevens.nl/research/papers/2023_PQC_Migration_Handbook.pdf.

[25] Rafael del Pino and Guilhem Niot. 2025. Finally! A Compact Lattice-Based Threshold Signature. Cryptology ePrint Archive, Paper 2025/872. doi:10.1007/978-3-031-91826-1_6

[26] Justin Drake, Dmitry Khovratovich, Mikhail Kudinov, and Benedikt Wagner. 2025. Hash-Based Multi-Signatures for Post-Quantum Ethereum. *IACR Communications in Cryptology* 2, 1 (2025). doi:10.62056/aey7qjp10

[27] European Commission. 2024. Coordinated Implementation Roadmap for the Transition to Post-Quantum Cryptography. https://digital-strategy.ec.europa.eu/en/library/coordinated-implementation-roadmap-transition-post-quantum-cryptography.

[28] European Telecommunications Standards Institute. 2023. *Quantum Safe Cryptography and Security.* Technical Report. ETSI. White Paper No. 8. https:

//www.etsi.org/images/files/ETSIWhitePapers/QuantumSafeWhitepaper.pdf.

[29] European Union Agency for Cybersecurity (ENISA). 2024. *Post-Quantum Cryptography: Current State and Quantum Mitigation.* Technical Report. ENISA. https://www.enisa.europa.eu/publications/post-quantum-cryptography-current-state-and-quantum-mitigation.

[30] Fraunhofer Institute for Secure Information Technology. 2022. *Practical Post-Quantum Cryptography.* Technical Report. Fraunhofer SIT. https://www.sit.fraunhofer.de/fileadmin/dokumente/studien_und_technical_reports/Practical.PostQuantum.Cryptography_WP_FraunhoferSIT.pdf.

[31] Lov K. Grover. 1996. A Fast Quantum Mechanical Algorithm for Database Search. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC)*. ACM, 212–219. doi:10.1145/237814.237866

[32] GSM Association. 2023. *Post-Quantum Telco Network Impact Assessment.* Technical Report. GSMA. https://www.gsma.com/newsroom/wp-content/uploads/PQ.1-Post-Quantum-Telco-Network-Impact-Assessment-Whitepaper-Version1.0.pdf.

[33] Intel Corporation. 2024. *Intel Software Guard Extensions (SGX).* https://www.intel.com/content/www/us/en/products/docs/accelerator-engines/software-guard-extensions.html

[34] Intel Corporation. 2024. *Intel Trust Domain Extensions (TDX).* https://www.intel.com/content/www/us/en/developer/tools/trust-domain-extensions/overview.html

[35] Samuel Jaques. 2024. Quantum Attacks on AES: When Do We Need to Worry about a Structureless, Quantum, Known-Plaintext Attack?. In *Conference on Cryptographic Hardware and Embedded Systems (CHES) — invited talk.* University of Waterloo. Presentation on concrete costs of Grover-based key recovery for AES..

[36] David Joseph, Rafael Misoczki, Marc Manzano, Joe Tricot, Fernando Dominguez Pinuaga, Olivier Lacombe, Stefan Leichenauer, Jack Hidary, Phil Venables, and Royal Hansen. 2022. Transitioning Organizations to Post-Quantum Cryptography. *Nature* 605 (2022), 237–243. doi:10.1038/s41586-022-04623-2

[37] Kris Kwiatkowski, Panos Kampanakis, Bas Westerbaan, and Douglas Stebila. 2026. Post-quantum Hybrid ECDHE-MLKEM Key Agreement for TLSv1.3. IETF Internet-Draft. https://datatracker.ietf.org/doc/draft-ietf-tls-ecdhe-mlkem/ Work in progress. Expires August 2026.

[38] Duc V. Le, Mahimna Kelkar, and Aniket Kate. 2019. Flexible Signatures: Making Authentication Suitable for Real-Time Environments. In *Computer Security – ESORICS 2019*, Kazue Sako, Steve Schneider, and Peter Y. A. Ryan (Eds.). Springer International Publishing, Cham, 173–193. doi:10.1007/978-3-030-29959-0_9

[39] libp2p Project. 2025. *libp2p 2025 Annual Report: Direction and Scope.* https://libp2p.io/reports/annual-reports/2025/

[40] Jameson Lopp, Christian Papathanasiou, Ian Smith, Joe Ross, Steve Vaile, and Pierre-Luc Dallaire-Demers. 2026. Post Quantum Migration and Legacy Signature Sunset. Bitcoin Improvement Proposal (BIP-361). https://github.com/bitcoin/bips/blob/master/bip-0361.mediawiki Draft proposal addressing post-quantum migration of Bitcoin signatures.

[41] National Cyber Security Centre (UK). 2024. *Next Steps in Preparing for Post-Quantum Cryptography.* Technical Report. NCSC. https://www.ncsc.gov.uk/whitepaper/next-steps-preparing-for-post-quantum-cryptography.

[42] National Institute of Standards and Technology. 2021. *Getting Ready for Post-Quantum Cryptography.* Technical Report CSWP 04282021. NIST. https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04282021.pdf.

[43] National Institute of Standards and Technology. 2024. *FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard.* Federal Information Processing Standard 203. NIST. doi:10.6028/NIST.FIPS.203 ML-KEM.

[44] National Institute of Standards and Technology. 2024. *FIPS 204: Module-Lattice-Based Digital Signature Standard.* Federal Information Processing Standard 204. NIST. doi:10.6028/NIST.FIPS.204 ML-DSA.

[45] National Institute of Standards and Technology. 2024. *FIPS 205: Stateless Hash-Based Digital Signature Standard.* Federal Information Processing Standard 205. NIST. doi:10.6028/NIST.FIPS.205 SLH-DSA.

[46] National Institute of Standards and Technology. 2024. Post-Quantum Cryptography FAQ. https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/faqs. Accessed March 2026.

[47] Office of Management and Budget. 2022. *Memorandum M-23-02: Migrating to Post-Quantum Cryptography.* Technical Report. Executive Office of the President. https://www.whitehouse.gov/wp-content/uploads/2022/11/M-23-02-M-Memo-on-Migrating-to-Post-Quantum-Cryptography.pdf.

[48] Optimism. 2026. Optimism post on X regarding protocol update / announcement. https://x.com/Optimism/status/2015423032613855380 Social media post, accessed 2026-04-29.

[49] Olaoluwa Osuntokun. 2026. Post-Quantum BIP-86 Recovery via zk-STARK Proof of BIP-32 Seed Knowledge. Bitcoin Development Mailing List. https://groups.google.com/g/bitcoindev/c/Q06piCEJhkI/m/5QOIiIozAgAJ

[50] PQShield. 2022. Quantum Threat 1: NIST PQC Standards Are Here — How Can You Keep Ahead? https://8530430.fs1.hubspotusercontent-na1.net/hubfs/8530430/Assets/PQShield%20Quantum%20Threat%201%20-%20NIST%20PQC%20Standards%20are%20here%20-%20How%20can%20you%20keep%20ahead%20-%20July%202022%20(1).pdf.

[51] PQShield. 2022. Quantum Threat 2: The First NIST Post-Quantum Cryptographic Standards. https://8530430.fs1.hubspotusercontent-na1.net/hubfs/8530430/Assets/PQShield%20Quantum%20Threat%202%20-%20The%20First%20NIST%20Post-Quantum%20Cryptographic%20Standards%20-%20July%202022%20(1).pdf.

[52] PQShield. 2022. Quantum Threat 3: An Overview of Post-Quantum Cryptography. https://8530430.fs1.hubspotusercontent-na1.net/hubfs/8530430/Assets/PQShield%20Quantum%20Threat%203%20-%20An%20Overview%20of%20Post-Quantum%20Cryptography%20-%20August%202022%20(1).pdf.

[53] Thomas Prest, Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Ricosset, Gregor Seiler, William Whyte, and Zhenfei Zhang. 2017. FALCON: Fast-Fourier Lattice-Based Compact Signatures over NTRU. https://falcon-sign.info/. NIST Post-Quantum Cryptography Standardization Project submission.

[54] Project Eleven. 2025. Bitcoin RisQ Metrics: Quantifying Quantum Risk to Bitcoin Addresses. Online. https://www.projecteleven.com/bitcoin-risq-list/metrics.

[55] Protocol Labs. 2019. libp2p: A modular network stack. https://libp2p.io

[56] Kiarash Sedghighadikolaei and Attila Altay Yavuz. 2025. A Survey of Threshold Signatures: NIST Standards, Post-Quantum Cryptography, Exotic Techniques, and Real-World Applications. *ACM Comput. Surv.* 58, 6, Article 143 (Dec. 2025), 39 pages. doi:10.1145/3772274

[57] István András Seres, Noemi Glaeser, and Joseph Bonneau. 2025. Short Paper: Naysayer Proofs. In *Financial Cryptography and Data Security*, Jeremy Clark and Elaine Shi (Eds.). Springer Nature Switzerland, Cham, 22–32. doi:10.1007/978-3-031-78679-2_2

[58] Peter W. Shor. 1994. Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 124–134. doi:10.1109/SFCS.1994.365700

[59] The White House. 2022. *Memorandum on Promoting United States Leadership in Quantum Computing While Mitigating Risks to Vulnerable Cryptographic Systems (NSM-10).* Technical Report NSM-10. Executive Office of the President. https://www.presidency.ucsb.edu/documents/memorandum-promoting-united-states-leadership-quantum-computing-while-mitigating-risks.

[60] World Economic Forum. 2022. *Transitioning to a Quantum-Secure Economy.* Technical Report. WEF. https://www3.weforum.org/docs/WEF_Transitioning%20to_a_Quantum_Secure_Economy_2022.pdf.

A List of Abbreviations

AA account abstraction
ABI Application Binary Interface
AES Advanced Encryption Standard
AML anti-money laundering
BFT Byzantine fault tolerant
BIP Bitcoin Improvement Proposal
BLS Boneh–Lynn–Shacham
BSI Bundesamt für Sicherheit in der Informationstechnik
CCCS Canadian Centre for Cyber Security
CFRG Crypto Forum Research Group
CFT combating the financing of terrorism
CRQC cryptographically relevant quantum computer
CSIRO Commonwealth Scientific and Industrial Research Organisation
CSWP Cybersecurity White Paper
dApp decentralized application
DeFi decentralized finance
ECC elliptic-curve cryptography
ECDH Elliptic Curve Diffie–Hellman
ECDSA Elliptic Curve Digital Signature Algorithm
EdDSA Edwards-curve Digital Signature Algorithm
EIP Ethereum Improvement Proposal
ENISA European Union Agency for Cybersecurity
EOA externally owned account
ERC Ethereum Request for Comment
ETSI European Telecommunications Standards Institute
EVM Ethereum Virtual Machine
FIPS Federal Information Processing Standards
FN-DSA FFT-based Lattice Digital Signature Algorithm (standardized Falcon)
HD hierarchical deterministic
HKDF HMAC-based Key Derivation Function
HNDL harvest-now-decrypt-later
HPKE Hybrid Public Key Encryption
IETF Internet Engineering Task Force
KEM key encapsulation mechanism
KMS key management system
KYC know-your-customer
MEV maximal extractable value
ML-DSA Module-Lattice Digital Signature Algorithm
ML-KEM Module-Lattice Key Encapsulation Mechanism
MPC multi-party computation
NCSC National Cyber Security Centre
NIST National Institute of Standards and Technology
NSM National Security Memorandum
P2P peer-to-peer
PoS proof-of-stake
PoW proof-of-work
PQ post-quantum
PQC post-quantum cryptography
RFC Request for Comments
RPC remote procedure call
RSA Rivest–Shamir–Adleman
SDK software development kit
SHA Secure Hash Algorithm
SLH-DSA Stateless Hash-based Digital Signature Algorithm
SLIP Satoshi Labs Improvement Proposal
SNARK succinct non-interactive argument of knowledge
STARK scalable transparent argument of knowledge
TEE trusted execution environment
TLS Transport Layer Security
USDC USD Coin
VRF verifiable random function
ZKP zero-knowledge proof