# Version Notice

Document: QRB Whitepaper
Version: 1.0
Date: 29 Aug 2025
Status: Initial public release
This document reflects QRB’s current design and roadmap as of the date above. Details may be refined as audits conclude, standards evolve, and implementations harden. A changelog will accompany subsequent versions.

---

# Changelog & Document Control

Versioning Scheme: MAJOR.MINOR.PATCH (e.g., 1.1 adds features/clarifications; 1.0.1 fixes typos).

v1.0 (2025-08-29)
• Initial public release of the QRB architecture, PQ-20, QuantumBridge, security model, and developer tooling.

Document Control
• Title: QRB Whitepaper
• Version: 1.0 (2025-08-29)
• Contact: info@<your-domain> • security@<your-domain>

---

# Index


**Version Notice** 1
**Changelog & Document Control** 2
Index 3
Quantum Resilience Blockchain (QRB) Whitepaper 5
1. Abstract 5
2. Introduction 6
3. Problem Statement 8
The Quantum Threat to Existing Blockchains 8
The Consequences 8
Why Half-Measures Aren't Enough 9
QRB's Holistic Solution 9
The Bottom Line 10
4. Technical Architecture 10
4.1. Overview 10
4.2 Post-Quantum Cryptographic Framework 11
4.3. PQ-20 Token Standard 12
4.4. Consensus & Validator Layer 13
4.5. Interoperability & IBC 14
5. Migration and Bridge Infrastructure 16
5.1. QBT Token Overview 16
5.2. Secure Bridging Architecture 18
5.3. NFT & Altcoin Migration 21
6. Tokenomics 23
6.1 Supply Breakdown 23
6.2 Utility of QBT and PQ-QBT 25
6.3 Deflationary and Control Mechanics 25
Figure 3: QBT/PQ-QBT Utility & Incentive Flow 26
7. Roadmap 26
7.1 Development Milestones 26
7.2 Ecosystem Expansion 29
8. Use Cases 31
8.1 Post-Quantum Asset Security 31
8.2 NFT Preservation & Recovery 32
8.3 Quantum-Secure DAOs & Voting 32
8.4 Quantum-Safe Cross-Chain DeFi 33
Summary Table 34
Roadmap Alignment 35
9. Security Model 35
9.1 Cryptographic Risk Analysis 35

9.2 Bridge Attack Surface & Mitigations 36
9.3 Validator and VDF Centralization Risks 36
9.4 Governance and Upgrade Security 37
Summary Table 38
Figure 5: Security Layers Stack 39
10. Developer & Ecosystem Tooling 39
10.1 PQ-20 SDK 40
10.2 Wallet Integration & Ledger Support 41
10.3 Smart Contract Templates 41
10.4 L2 and ZK Compatibility 42
10.5 Developer Experience & Infra 42
Summary Table 43
11. Governance Framework 43
11.1 Principles 44
11.2 Roles 44
11.3 Processes 44
11.4 Voting & Quorum 44
11.5 Upgrades & Rollbacks 45
11.6 Risk & Audit 45
12. Conclusion 45
13. Appendices 46
A. Glossary 46
B. Cryptographic Primer 47
C. Key Contract Interfaces 48
D. Migration Flow Diagram 49
E. Technical Paper References 50
**Disclaimer & Forward-Looking Statements 51**

# Quantum Resilience Blockchain (QRB) Whitepaper

---

## 1. Abstract

**The Quantum Threat**— Quantum computing poses a critical threat to blockchain security. Algorithms like ECDSA and SHA-256—which secure Bitcoin, Ethereum, and nearly all digital assets—can be compromised by Shor's and Grover's algorithms. This endangers:

- Private keys
- Smart contracts
- Cross-chain bridges
- NFT authenticity and ownership.

**The QRB Solution**— Quantum Resilience Blockchain (QRB) is a next-generation Layer-1 blockchain engineered to withstand quantum attacks. It replaces classical cryptography with NIST-standardized, quantum-safe primitives:

- **Dilithium** for post-quantum transaction signing and authentication
- **SPHINCS+** for stateless smart contract security and governance integrity
- **Kyber** for hybrid encryption for secure messaging and storage across the network

**Key Innovations**

- **PQ-20 Token Standard**: Modular, upgradeable, and quantum-secure token layer
- **QBT Bridge Token**: Enables smooth transition from Ethereum, BSC, and Polygon to **QRB**

- **Governance-Controlled Sunset**: Legacy bridges are phased out securely via community voting

- **NFT Preservation Layer**: Uses **SPHINCS+** to anchor metadata and royalties with tamper-proof cryptographic hashes

- **PoS with Quantum Security**: Validators are authenticated using Dilithium signatures, randomness is secured by hardware-accelerated VDFs, and the network is designed to support threshold post-quantum multi-signatures for consensus governance.

- **MPC Fallbacks**: Migration contracts use **Dilithium**-based proofs and **multi-party computation (MPC)** to ensure secure, irreversible asset migration. Legacy wrapped assets are retired through time-locked, decentralized governance mechanisms.

**QRB is not a patch**—it is a purpose-built, quantum-secure infrastructure for the next generation of decentralized systems. This whitepaper presents **QRB's** architecture, tokenomics, migration strategy, cryptography, and governance—positioning it as a foundational layer for post-quantum decentralized infrastructure.

---

## 2. Introduction

**Why Quantum Security Matters Now**— Quantum computers are advancing rapidly. Within years, they will be capable of breaking today's most widely used cryptographic systems. Blockchains that rely on ECDSA, RSA, or SHA-2 are not future-proof—and billions in value may be at risk.

**How QRB Protects You**— QRB is designed from the ground up with a modular architecture, supporting scalable **PoS consensus**, secure cross-chain interoperability via **Cosmos IBC**, and **PQ-20** — a new token standard that enforces quantum-resistant asset management at the protocol level:

- **Quantum-Resistant Cryptography**
    - All signatures use Dilithium instead of ECDSA
    - Smart contracts secured with SPHINCS+, eliminating key reuse and state exhaustion
    - Encrypted communication via Kyber + AES-GCM
- **Seamless Migration**
    - Ethereum-based tokens can be migrated to PQ-20 with verifiable proofs
    - **QBT** is a bridge token initially deployed on **Ethereum, BSC, and Polygon**, and can be migrated 1:1 into the **PQ-20** format using **Dilithium**-secured bridges
    - No rushed upgrades—Time-locked bridges give users and **dApps** a safe upgrade window
- **Full Ecosystem Coverage**
    - Supports DeFi, NFTs, DAOs, and staking
    - NFT metadata and ownership is anchored with quantum-secure hashes
- **For Developers & Builders**
    - Same modularity and flexibility as Ethereum
    - Includes SDKs and wallet libraries for Dilithium, SPHINCS+, and Kyber
    - No deep cryptographic knowledge required to build quantum-resistant dApps

---

# 3. Problem Statement

## The Quantum Threat to Existing Blockchains

Most blockchain networks were built on cryptographic foundations now threatened by emerging quantum technologies. Today's chains rely heavily on:

- **ECDSA** for digital signatures → vulnerable to **Shor's algorithm**
- **SHA-2** for hashing → weakened by **Grover's algorithm**

Quantum computers capable of executing these algorithms could:

- **Recover private keys** and access user wallets
- **Forge digital signatures**, allowing malicious actors to drain bridges
- **Disrupt smart contracts** and tamper with on-chain governance mechanisms

Even before such attacks are feasible, the **"harvest now, decrypt later"** threat makes long-term blockchain data vulnerable if it isn't quantum-resistant today.

---

## The Consequences

- **Stolen Funds**: Quantum attackers could drain wallets and treasuries.
- **Bridge Hijacks**: Forged proofs could unlock wrapped tokens across chains.
- **DAO Collapse**: Quantum-compromised signatures could corrupt governance.
- **Frozen Ecosystems**: Migrating to quantum-safe models later could cause massive disruption.

---

## Why Half-Measures Aren't Enough

While some projects propose hybrid cryptography (e.g., combining ECDSA with post-quantum schemes), these remain:

- **Temporary fixes**, not foundational solutions
- Dependent on legacy infrastructure and assumptions
- Unsuitable for smart contracts, full validator consensus, and asset migration at scale

---

## QRB's Holistic Solution

The **Quantum Resilience Blockchain (QRB)** addresses the entire threat surface — not just part of it.

- **No Legacy Code**: All cryptographic primitives are quantum-resistant from genesis.
- **NIST-Approved Algorithms**:
    - **CRYSTALS-Dilithium** for transaction authentication
    - **SPHINCS+** for smart contract and governance integrity
    - **Kyber** for post-quantum encrypted communication
- **PQ-20 Token Standard**: Native quantum-secure asset model
- **QBT Bridge Token**: Deployed on Ethereum, BSC, and Polygon — enabling verifiable, 1:1 migration into QRB via **Dilithium-signed** bridges and **MPC-backed** fallback recovery
- **Secure Governance**: Built-in protections for DAO voting and validator onboarding in a quantum world

---

### The Bottom Line

QRB is the first Layer-1 chain offering:

- **Smart contracts secured from the ground up**
- **Seamless and secure token migration**
- **Quantum-resistant consensus, governance, and interoperability**

**Without a quantum-safe foundation, no blockchain is future-proof. QRB is that foundation.**

---

# 4. Technical Architecture

---

## 4.1. Overview

The architecture of Quantum Resilience Blockchain (QRB) is designed from the ground up to operate in a quantum-adversarial environment while maintaining performance, scalability, and interoperability. It replaces vulnerable cryptographic primitives with post-quantum counterparts and introduces modular components that enable secure asset management, contract execution, and network communication.

QRB's design integrates three primary layers:

- **Post-Quantum Cryptographic Layer:** Implements NIST-selected algorithms (Dilithium, SPHINCS+, Kyber) to secure transaction signing, smart contracts, and encrypted communication.
- **Consensus & Validator Layer:** A Proof-of-Stake protocol enhanced with Dilithium-signed block validation and VDF-driven randomness.

- **Bridging & Interoperability Layer:** Cosmos SDK with IBC for scalable cross-chain migration and PQ-20 token standard enforcement.

Each component is modular and independently upgradable, allowing future improvements as quantum threats evolve and cryptographic standards mature. QRB is optimized for both native post-quantum deployments and seamless onboarding of assets and users from vulnerable legacy networks.

---

## 4.2 Post-Quantum Cryptographic Framework

At the core of QRB is a cryptographic framework designed to withstand the capabilities of quantum adversaries. It integrates multiple **NIST-approved post-quantum algorithms** to handle distinct network operations with cryptographic separation of concerns:

- **CRYSTALS-Dilithium**: A lattice-based digital signature scheme used for transaction validation, validator signatures, and bridge authentication. It offers a balance between performance and post-quantum security, with small public keys and moderate signature sizes (~2.4KB).

- **SPHINCS+**: A stateless, hash-based signature scheme used for high-security, low-frequency events such as smart contract deployment, protocol upgrades, and governance. Its deterministic and state-free nature eliminates key exhaustion risks.

- **Kyber512**: A post-quantum key encapsulation mechanism (KEM) used for secure encrypted communication between nodes, validator messaging, and P2P networking. It enables hybrid encryption by pairing with AES-GCM for efficient, quantum-safe message exchange.

This multi-layered cryptographic framework ensures that QRB can:

- Protect digital signatures and transaction authentication from Shor’s algorithm.

- Secure on-chain logic and immutable code using post-quantum auditability.

- Enable encrypted messaging and validator communication resistant to Grover-accelerated brute-force attacks.

By assigning cryptographic algorithms to roles based on their strengths, QRB establishes a robust and extensible foundation that defends against known and anticipated quantum threats.

---

## 4.3. PQ-20 Token Standard

The PQ-20 token standard is a core innovation within the Quantum Resilience Blockchain (QRB), enabling post-quantum secure, programmable, and composable digital assets. It replaces classical token standards such as ERC-20 and BEP-20, which rely on quantum-vulnerable cryptographic primitives, with a fully quantum-resistant architecture.

PQ-20 introduces the following critical features:

- **Post-Quantum Signature Validation**: Each token transfer or approval must be signed using CRYSTALS-Dilithium, ensuring that transactions remain secure even in the presence of quantum adversaries.

- **On-Chain Verifiability**: The standard includes native support for verifying Dilithium signatures directly on-chain. This avoids reliance on quantum-vulnerable oracles or wrappers, enabling smart contracts to enforce security guarantees without leaving the chain.

- **Modular Compliance**: PQ-20 is designed to be modular and extensible. It supports optional modules for governance voting, vesting schedules, cross-chain bridges, and recovery mechanisms. These modules can be composed in a permissionless and secure manner.

- **Identity Binding**: PQ-20 supports optional post-quantum identity linking (e.g., PQ DIDs), allowing advanced identity-based applications such as soulbound tokens and quantum-secure DAOs.

- **Cross-Chain Migration Metadata**: Each PQ-20 token may store metadata regarding its origin chain (e.g., Ethereum, BSC), previous address, and bridge

signature used during migration. This provides an auditable trail of token provenance, improving transparency and trust during the transition from legacy networks.

PQ-20 is designed for performance and minimal overhead despite using larger post-quantum signatures. Compression techniques and optional zero-knowledge verification pathways are supported at the VM level to ensure transaction throughput remains competitive.

This standard enables the secure evolution of tokenized assets into the post-quantum era without breaking composability, programmability, or auditability—making PQ-20 the foundation for next-generation decentralized economies.

---

## 4.4. Consensus & Validator Layer

The Quantum Resilience Blockchain (QRB) employs a highly secure, quantum-resistant consensus layer based on a modified **Proof-of-Stake (PoS)** protocol enhanced with **post-quantum validator authentication**, **Verifiable Delay Functions (VDFs)** for fair leader election, and optional support for **threshold post-quantum multisignatures**.

### Post-Quantum Validator Authentication

QRB validators are required to use **CRYSTALS-Dilithium** for block signing and authentication. This ensures that even if quantum adversaries emerge, they cannot forge validator signatures or impersonate consensus participants. Validator public keys are registered on-chain and periodically verified for authenticity using on-chain Dilithium signature proofs.

### Verifiable Delay Functions (VDFs) for Leader Randomness

To prevent quantum-speed manipulation of randomness, QRB introduces **VDF-based randomness beacons** for selecting block proposers. These delay functions are computationally expensive to solve but easy to verify, offering resistance against quantum grinding attacks. QRB allows optional hardware acceleration (e.g., FPGA-based VDFs) and supports fallback entropy sources when needed.

### Stake-Weighted Voting and Slashing

Block proposers are selected probabilistically based on stake weight, and blocks are validated through a round of stake-weighted approvals. Validators must maintain uptime and sign blocks correctly or face **slashing penalties**, including partial stake confiscation and temporary removal from the validator set.

### Multisignature Threshold Validation (future Upgrade)

As quantum-safe threshold cryptography matures, QRB is designed to support **Dilithium-based multisignature schemes** for block validation. This enables decentralized finality committees and governance-controlled quorum logic, paving the way for more robust decentralized validator governance in later phases.

### Incentives and Finality

Validators earn QBT and PQ-QBT token rewards proportional to their stake and performance. QRB supports fast finality checkpoints and optimized validator rotation for long-term scalability and decentralization.

---

This hybrid design ensures that QRB's consensus is both energy-efficient and resilient to future quantum threats. By securing every validator operation—from identity to block validation—under post-quantum cryptography, QRB eliminates the weakest links present in legacy consensus layers.

---

## 4.5. Interoperability & IBC

Interoperability is a cornerstone of the Quantum Resilience Blockchain (QRB), enabling secure interaction with legacy and future decentralized networks. QRB leverages the **Cosmos SDK** and **Inter-Blockchain Communication (IBC)** protocol, augmented with post-quantum cryptographic enhancements to ensure cross-chain operations remain secure in a quantum-capable world.

### Cosmos SDK Modularity

QRB is built on the **Cosmos SDK**, chosen for its proven modularity, developer ecosystem, and compatibility with IBC. This provides QRB with built-in capabilities such as:

- Plug-and-play consensus and staking modules
- Governance and slashing infrastructure
- Seamless upgrade paths via governance proposals

QRB extends the SDK with post-quantum cryptographic modules, including support for Dilithium-based signing and verification, SPHINCS+ governance actions, and Kyber-based encrypted state channels.

### Quantum-Hardened IBC

IBC allows QRB to exchange data and assets with other IBC-enabled chains. However, classical IBC relies on **SHA-2 Merkle proofs** and **ECDSA validator sets**, which are vulnerable to quantum attacks.

QRB hardens IBC by:

- Replacing SHA-2 with **SHA-3** or **BLAKE3** in light client proofs
- Supporting **Dilithium validator sets** for verifying consensus states
- Planning future upgrades toward **ZK-validity light clients** to replace Merkle proofs with succinct, quantum-resistant alternatives

### Cross-Chain Token and Data Bridging

QRB facilitates two main types of interoperability:

1. **Asset Migration Bridges** – Using Dilithium-signed smart contracts, assets like QBT from Ethereum, BSC, and Polygon can be securely wrapped and redeemed as PQ-20 tokens on QRB. These bridges are time-locked and include multi-party fallback mechanisms for fail-safe recovery.
2. **Data Interoperability** – Oracles and smart contracts on QRB can read cross-chain data securely using post-quantum signed relays and IBC-compatible oracle networks (e.g., Band Protocol, UMA, or QRB-native oracles).

### Future-Proofing for Web3 Interconnectivity

QRB's IBC layer is designed to integrate with upcoming Layer-1 and Layer-2 networks as they adopt post-quantum standards. This ensures that QRB remains a secure hub for cross-chain liquidity, messaging, and computation even as the broader ecosystem evolves.

---

By combining the flexibility of Cosmos IBC with the cryptographic strength of post-quantum primitives, QRB builds a truly secure interoperability layer. This ensures not only seamless migration from legacy blockchains but also future participation in the decentralized quantum-ready Web3 landscape.

---

# 5. Migration and Bridge Infrastructure

## 5.1. QBT Token Overview

The **QBT token** is the gateway to the Quantum Resilience Blockchain (QRB), serving as both an early-access pass and a secure migration utility. Initially deployed on Ethereum as an ERC-20 token, QBT enables seamless onboarding into QRB's quantum-resistant ecosystem.

With a fixed supply of **10 billion tokens**, QBT is allocated as follows:

- **40%** Treasury — for long-term development and ecosystem growth
- **20%** Team — vested over 3 years
- **15%** Community & Grants
- **15%** Presale — offered at tiered pricing
- **10%** Liquidity — for decentralized exchange (DEX) stability

---

### Pre-Mainnet Utility

Before QRB mainnet launch, QBT empowers participants to:

- **Join the Presale** – Acquire QBT at discounted tiers, with early-bird bonuses
- **Shape Governance** – Propose and vote on validator rules, testnet parameters, and grant initiatives
- **Reserve Access** – Whitelist for validator onboarding, dev bounties, and early dApp deployment
- **Prepare for Migration** – Lock QBT and associate a post-quantum wallet for seamless redemption of PQ-QBT

---

### Post-Mainnet Utility

Following the mainnet launch, QBT evolves into:

- **Burn-to-Migrate** – Burn QBT on Ethereum to mint PQ-QBT on QRB at a 1:1 ratio
- **Cross-Chain Bridging** – Continue using QBT as a migration and bridging utility for EVM-based users
- **Governance Rights** – PQ-QBT holders gain full voting power over upgrades, staking economics, and protocol changes
- **Exclusive Features** – Stake PQ-QBT to unlock ZK-proof services, validator roles, and premium network functionality

---

### Secure Migration via Quantumbridge

All migrations from QBT to PQ-QBT are safeguarded by:

- **Dilithium-Signed Proofs** – Quantum-resistant address binding
- **One-Time Minting Guarantees** – Prevents replay or double-redemption
- **MPC-Based Fallbacks** – Emergency recovery via multi-party validation

- **Governance-Controlled Sunset** – Community votes determine when legacy bridges are closed

---

**In summary**, QBT is more than a token — it's your launchpad into a quantum-secure future. From early participation and governance to post-quantum interoperability, QBT ensures a trusted and forward-compatible transition into the QRB network.

---

## 5.2. Secure Bridging Architecture

The **QuantumBridge** is QRB's two-phase post-quantum bridging protocol, designed to securely migrate QBT tokens and NFTs from Ethereum, BSC, and Polygon into the post-quantum realm of the QRB mainnet.

It guarantees:

- **1:1 token preservation** — no inflation or wrapped derivatives
- **Quantum-safe authentication** — using *Dilithium* and *SPHINCS+*
- **Decentralized security** — powered by *validators + MPC + governance*

---

### Phase 1: Lock-and-Prove (on Legacy Chains)

This phase occurs on Ethereum and other EVM chains where QBT or NFTs are held.

#### For QBT (erc-20)

- **Burn-to-Migrate**: QBT tokens are permanently burned

- **Quantum Address Binding**: Users submit a Dilithium-based post-quantum address
- **Event Emission**: Bridge emits migration event for validator processing

For Nfts (erc-721 / Erc-1155)

- **Escrow Locking**: NFTs are transferred to a smart contract escrow
- **Metadata Anchoring**: SPHINCS+ signatures bind off-chain data (IPFS hash, royalties)

Shared Security Protections

| Feature | Protection |
|---|---|
| Minimum PQ Address Length | Prevents malformed or malicious payloads |
| Reentrancy Guards | Ensures atomic operations (lock + emit) |
| Fraud-Proof Window | 48-hour challenge period for suspicious ops |

---

Phase 2: Redeem-and-Mint (on QRB Mainnet)

Once verified, assets are redeemed as quantum-native tokens on the QRB mainnet.

For PQ-QBT

- **Threshold Validation**: 5-of-9 validators verify burn TX and address match
- **Minting with Provenance**: PQ-QBT includes origin metadata for auditability

For PQ-Nfts

- **Dual Verification**: NFT escrow confirmed and metadata certificate validated
- **Immutable Lineage**: Mints PQ-NFT with proof of legacy ownership and metadata

---

Cross-Asset Security Matrix

| Feature | QBT (ERC-20) | NFTs (ERC-721/1155) |
|---|---|---|
| On-Chain Proof | Burn TX + Dilithium signature | SPHINCS+-signed metadata escrow |
| Minting on QRB | 1:1 mint after validator quorum | Replica mint with full provenance |
| Recovery Mechanism | MPC fallback (after 90 days) | Governance vote to reclaim escrow |
| Bridge Expiry | Optional sunset via governance | Indefinite reversibility by design |

---

Why This Architecture Works

For Nfts

- **SPHINCS+ metadata anchoring** → tamper-proof digital art provenance
- **Escrow reversibility** → collectors retain ownership until redemption
- **Royalty preservation** → creator fees remain intact across migration

For All Assets

- Unified logic and validator quorum
- Modular and upgradeable bridge contracts
- Compatibility with light clients and ZK-proofs

---

Example: Migrating Cryptopunk #9999

1. **Lock** — Escrow the NFT on Ethereum, signed with SPHINCS+
2. **Verify** — Validators confirm escrow and metadata authenticity
3. **Mint** — PQ-Punk #9999 is minted with full provenance on QRB

---

## 5.3. NFT & Altcoin Migration

Migrating NFTs and altcoins from legacy chains to QRB requires maintaining their identity, metadata, and utility within a quantum-safe environment. The QuantumBridge infrastructure provides a robust, flexible mechanism tailored to the unique characteristics of each asset class.

### NFT Migration Strategy

- **Preservation of Authenticity**: Each NFT migration includes a SPHINCS+ signature over its metadata (e.g., IPFS hash, royalties, collection ID).
- **Escrow & Proof**: Original NFT is locked in an escrow contract; metadata proof is submitted to QRB validators.
- **Minting with Lineage**: Validators confirm metadata and provenance before minting the PQ-NFT on QRB.
- **Royalty Inheritance**: Smart contracts auto-port creator fees (e.g., OpenSea royalties) to the new chain.
- **Reversibility**: Governance can unlock NFTs back to legacy chains under controlled conditions.

---

### Altcoin (erc-20/Bep-20) Migration Strategy

- **Whitelist Migration Only**: QRB does not support arbitrary wrapped tokens. Only audited and whitelisted assets can migrate.
- **Token Lock & Audit Trail**: Original tokens are locked or burned; event logs are submitted for validation.
- **Quantum-Aware Minting**: PQ-Tokens are minted with metadata referencing origin chain, hash of the burn/lock TX, and cross-chain identifiers.
- **DeFi Reinstatement**: Supported altcoins can regain utility in QRB DeFi protocols via PQ versions (PQ-USDC, PQ-DAI, etc.)

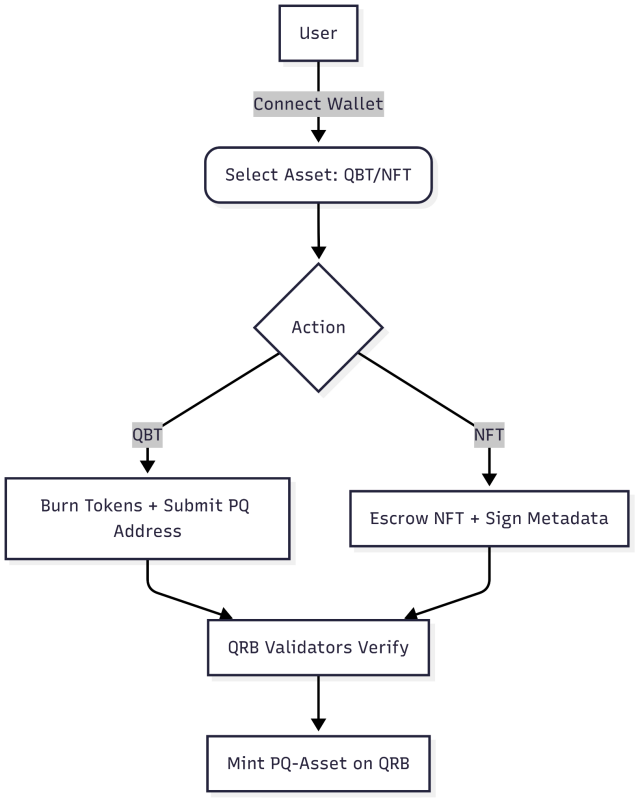
---

### User Flow Summary

1. **Connect Wallet** on Ethereum/BSC/Polygon
2. **Burn or Escrow** assets via QuantumBridge UI
3. **Register PQ Address** (Dilithium/Sphincs+) on bridge
4. **Validators Confirm** on-chain proof and metadata
5. **PQ Token/NFT Minted** on QRB
6. **Track Origin & Metadata** in blockchain index

---

Figure 1: Quantumbridge Asset Migration Process




This architecture ensures trustless interoperability and long-term security for the digital assets most vulnerable to quantum threats.

---

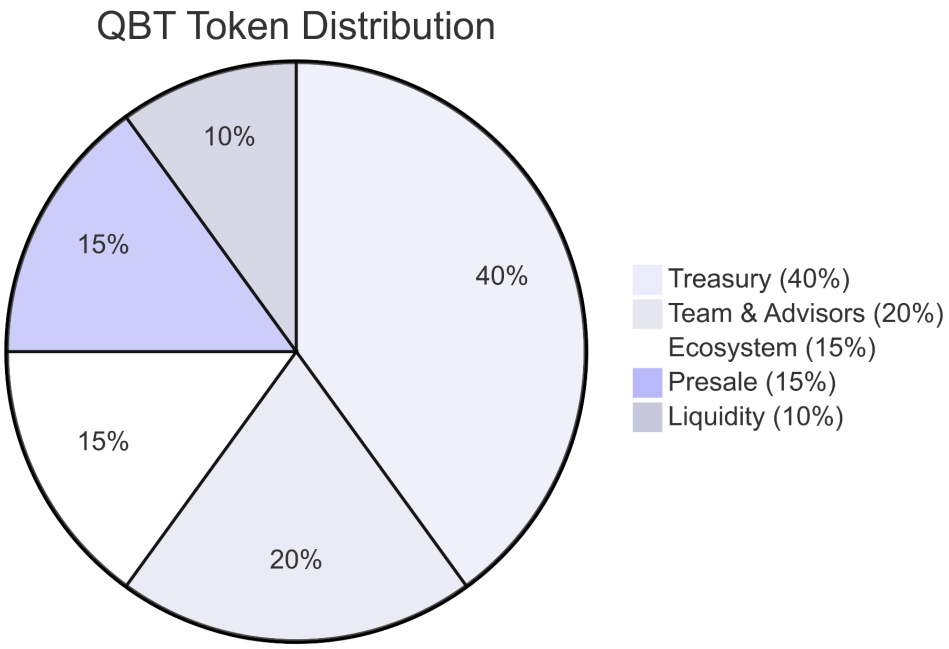
# 6. Tokenomics

---

## 6.1 Supply Breakdown

The total supply of QBT tokens is capped at **10,000,000,000 (10 billion)** units. The allocation is strategically designed to support long-term sustainability, governance participation, and ecosystem growth.

Figure 2: QBT Token Distribution




| Allocation | % | Purpose | Release Schedule |
|---|---|---|---|
| Treasury | 40% | Long-term development,Protocol upgrades, grants, ecosystem incentives | Governance-controlled |
| Team & Advisors | 20% | Core development and strategic guidance | 6-mo cliff, 36-mo linear vest |
| Ecosystem | 15% | Hackathons, partnerships, and public goods funding | Milestone-based (transparent votes) |
| Presale | 15% | Early backers, tiered pricing | 1-mo cliff, 6-mo linear vest |
| Liquidity | 10% | DEX pools (Uniswap/PancakeSwap) and market stability | over 12 months |

## 6.2 Utility of QBT and PQ-QBT

### Pre-Mainnet Utility (QBT)

- **Presale Participation** — Early supporters receive QBT tokens at discounted tiers
- **Governance Rights** — Voting on early testnet parameters and grant funding
- **Validator Access** — Reserve slots for validator onboarding and staking pools
- **Migration Preparation** — Register PQ-address and lock QBT for guaranteed swap

### Post-Mainnet Utility (PQ-QBT)

- **Governance** — PQ-QBT is the primary voting token on the mainnet
- **Staking** — Used to participate in PoS consensus and earn validator rewards
- **Gas Discount** — Transaction fee reductions for dApps and smart contracts using PQ-QBT
- **Bridge Role** — Used as a medium for legacy-to-quantum migration

---

## 6.3 Deflationary and Control Mechanics

QRB introduces long-term sustainability through controlled emission and deflationary elements:

**Mechanisms:**

- **Burn-to-Migrate** — Every migrated QBT is burned on Ethereum, reducing legacy supply
- **Governance Voting Burn** — Small burn fee applied to proposal submissions

- **Validator Penalties** — Slashed tokens are partially burned to maintain economic integrity

**Supply Security**

- **Treasury Locks**: Funds released via governance vote.
- **Team Vesting**: Smart contract enforces 6-month cliff + 3-year linear release.
- **Liquidity**: **10%.** Dedicated to DEX pools and market stability.

Figure 3: QBT/PQ-QBT Utility & Incentive Flow


---

# 7. Roadmap

## 7.1 Development Milestones

QRB follows a multi-phase development timeline designed to ensure robust testing, developer onboarding, and adoption milestones before full mainnet launch.

---

### Phase 0: Research & Protocol Design (Completed)

- Selection of NIST-approved post-quantum algorithms: Dilithium, SPHINCS+, Kyber.
- Design of the PQ-20 token standard and QBT bridge mechanism.
- Validator and consensus architecture with Dilithium-based authentication and VDF randomness.
- Initial threat modeling and MPC fallback structure.
- Drafting whitepaper, early community involvement, and ecosystem analysis.

### Phase 1: Presale & Community Formation (Ongoing)

- ERC-20 QBT token launch on Ethereum, BSC and Polygon.
- Treasury allocation for development, audits, and validator incentives.
- Community formation, governance preview, and early whitelisting for testnet validators and builders.
- Presale rounds with tiered pricing.
- Token utility campaign: access to validator roles, bridge trials, and PQ registration onboarding.
- Presale concludes Q4 2025.
- Bridge contract audits completed by Q1 2026.

---

### Phase 2: Testnet Alpha (Q1 2026)

- Launch of validator testnet with Dilithium-signed PoS consensus.
- Initial governance simulation using SPHINCS+ on-chain voting.
- Bridge prototype for burn-and-mint QBT migration (with MPC fallback).
- NFT escrow + metadata anchoring via SPHINCS+
- Testnet faucet, explorer, validator dashboard, and developer SDKs.

---

### Phase 2.5: Alpha Scalability (Q2 2026)

- Bridge UI refinements for user feedback loops.
- Stress-testing to simulate 10,000 TPS load.
- Public bug bounty campaign targeting bridge and validator modules.

---

### Phase 3: Testnet Beta (Q3 2026)

- Validator incentives and staking UX rollout.
- NFT + altcoin migration simulation with audit trail and SPHINCS+ proofing.
- Cosmos IBC integration for interoperability testing.
- Initial ZK rollup compatibility for privacy (ZK-STARK-based testing for PQ-light clients).
- Ecosystem bounties for PQ-20 token tooling, NFT porting tools and governance dApps.

---

### Phase 3.5: Beta Scaling & Final Audits (Q4 2026)

- **Stress-testing validator throughput** to validate network performance under load, targeting 10,000 transactions per second (TPS).
- **Bridge UI and UX refinement** based on feedback and internal review from alpha/beta participants.
- **Public bug bounty program** launched to detect critical vulnerabilities before mainnet.
- **Formal audit reports** finalized for core modules: consensus, bridge, staking, and governance.

- **Upgrade candidate frozen** for deployment on mainnet chain.
- **Enable NFT migration** for BSC and Polygon via QuantumBridge
- **Cross-chain porting logic tested** and validated for multi-chain provenance

---

### Phase 4: Mainnet Launch (Q1 2027)

- 1:1 PQ-QBT redemption enabled via Dilithium bridge proofs.
- PQ-governance activation using SPHINCS+-verified voting.
- Launch of PQ-20 smart contract templates and SDKs.
- Release of the QuantumBridge NFT migration portal.

---

### Phase 5: Post-Mainnet Hardening (2027)

- Privacy integrations via ZK-rollups and encrypted metadata (ZK-STARK).
- Rollup support extended with lattice-compatible verifiers.
- Threshold signature modules and MPC upgrades for validator nodes.
- Formal audit reports and community validation of PQ-20 token adoption.
- Governance-led deprecation and final retirement of ERC-20 legacy bridges and QBT wrapping on EVM.

---

## 7.2 Ecosystem Expansion

QRB is not just a blockchain—it is a migration-ready quantum-resilient ecosystem. Post-mainnet efforts will focus on adoption, toolkits, and strategic growth:

**Quantum Grants**

- Funding dApps, PQ-wallets, NFT migration tools, and ZK circuits.
- Priority support for projects using the QBT → PQ-QBT bridge.

**NFT Porting Initiative**

- Dedicated flow for migrating legacy NFTs with SPHINCS+-anchored provenance.
- Governance-based approval for verified creators (DAO-based curation).
- Royalty metadata preservation and OpenSea mapping modules.

**DeFi Incubator**

- Infrastructure for DEXs, stablecoins, and yield protocols built on PQ-20.
- Oracle feeds integrated with post-quantum price signature chains.
- Pilot project: Quantum-secure lending pool.

**Validator Academy**

- Educational content on Dilithium signing, VDF manipulation protection, and staking simulation.
- Incentivized staking trials and security simulations.

**ZK Builders Program**

- PQ-compatible ZK circuit library for smart contracts.
- ZK-based privacy layers with Cosmos IBC light client compatibility.

**Additional Notes**

- **Contingency Phase (optional)**: A buffer window post-Testnet Beta for additional security audits.
- **Governance Adaptivity**: PQ-QBT holders will steer roadmap changes through protocol voting.
- **Post-Quantum Threat Forecasts**: QRB roadmap urgency is backed by forecasts from IBM, NIST, and the European PQC initiative.

---

# 8. Use Cases

The transition to a quantum-resistant ecosystem is not just a technical upgrade—it is a necessary evolution to protect real-world value. QRB delivers both the **cryptographic depth** and the **practical functionality** needed to safeguard assets, governance, and interoperability in the post-quantum era.

---

## 8.1 Post-Quantum Asset Security

**Risk:** Classical cryptography (ECDSA, RSA) leaves treasuries, vaults, and long-term holdings exposed to Shor-enabled quantum attacks. Even before real-time quantum computers arrive, adversaries can record encrypted data today and decrypt it later (“harvest now, decrypt later”).

**QRB’s Solution:**

- **Dilithium** signatures secure all transactions at the protocol level.
- **SPHINCS+** protects smart contracts from signature forgery.
- **PQ-20 token standard** enforces quantum-resistant token lifecycle management.

- **MPC fallback** ensures resilience in bridge redemption.

**Key Benefit:** Long-term preservation of digital assets, including treasuries and tokenized real-world assets, without fear of retroactive decryption.

---

## 8.2 NFT Preservation & Recovery

**Risk:** NFTs are especially vulnerable to provenance and metadata attacks. A quantum attacker could forge a signature to illegally transfer high-value NFTs like CryptoPunks or Bored Apes.

**QRB's Solution:**

- **SPHINCS+ metadata anchoring** ensures permanent provenance and tamper-proof ownership.
- **Escrow-based migration** avoids destructive burns when porting NFTs.
- **Royalty inheritance logic** preserves creator economics across chains.
- **Governance-controlled fallback** enables recovery of orphaned or lost NFTs.

**Key Benefit:** Collectors, creators, and marketplaces gain a **tamper-proof, quantum-safe ownership model** with optional reversibility, something classical chains cannot provide.

---

## 8.3 Quantum-Secure DAOs & Voting

**Risk:** DAO governance can collapse if attackers forge private keys of council members or token holders, hijacking proposals or draining treasuries.

**QRB's Solution:**

- **Dilithium-verified ballots** secure on-chain proposal signing and voting.

- **SPHINCS+-anchored governance contracts** prevent forgery and maintain integrity.

- **Key rotation and stake access** enforced by post-quantum logic.

- **PQ-QBT tokens** serve as the basis for weighted voting and proposal sponsorship.

**Key Benefit:** DAOs on QRB remain **verifiable, tamper-resistant, and corruption-proof** under post-quantum adversaries.

---

## 8.4 Quantum-Safe Cross-Chain DeFi

**Risk:** Bridges, DEXs, and lending protocols are frequent exploit targets. Classical exploits (Ronin, Wormhole) already caused billions in losses—quantum attacks would multiply this risk.

**QRB's Solution:**

- **Quantum-hardened bridges** authenticated with Dilithium signatures.
- **SPHINCS+-secured contracts** for privacy-preserving swaps and lending.
- **MPC fallback** in bridge redemption ensures safety against incomplete proofs.
- **PQ-20 token standard** enforces post-quantum secure asset flows and composability across chains.
- **Cosmos IBC integration** enables post-quantum interoperability with Ethereum, BSC, Polygon, and beyond.

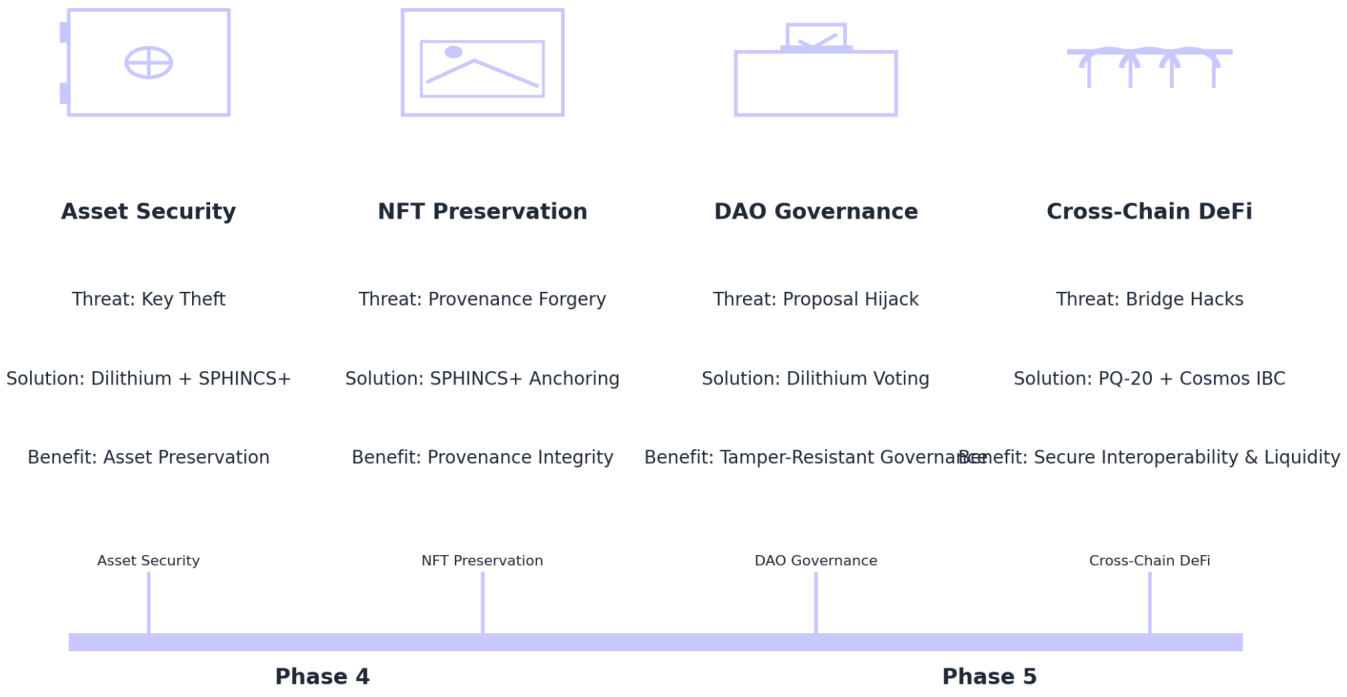
**Key Benefit:** Secure, future-proof DeFi and cross-chain liquidity without exposure to signature forgery or legacy vulnerabilities.

---

## Summary Table

| Use Case | Quantum Threat | QRB's Solution | Key Benefit |
|---|---|---|---|
| Asset Security | Private Key Theft / Retroactive Decrypt | Dilithium + SPHINCS+ + PQ-20 + MPC fallback | Unbreachable Asset Preservation |
| NFT Preservation | Provenance Forgery / Metadata Hijack | SPHINCS+ Anchoring + Escrow + Royalty Logic + DAO fallback | Provable, Future-Proof Ownership |
| DAO Governance | Malicious Proposal Passing / Key Forgery | Dilithium Voting + SPHINCS+ Governance + PQ-QBT | Corruption-resistant governance |
| Cross-Chain DeFi | Bridge Hacks via Forged Signatures | Quantum-Hardened Bridges + PQ-20 + Cosmos IBC + MPC | Secure interoperability & liquidity |

Figure 4: QRB Use Cases – Threats, Solutions, Benefits, and Roadmap Alignment



### Roadmap Alignment

- **Asset Security (8.1):** Delivered at **Mainnet Launch (Phase 4)**
- **NFT Preservation (8.2):** Enabled via QuantumBridge at **Mainnet (Phase 4)**
- **DAO Governance (8.3):** PQ-secured voting and governance at **Phase 4**
- **Cross-Chain DeFi (8.4):** Full interoperability and liquidity protocols roll out in **Post-Mainnet Hardening (Phase 5)**

---

## 9. Security Model

QRB's architecture is **security-first**. Every protocol layer—from transaction signing to cross-chain migration—is built with quantum-resistant cryptographic primitives and **defense-in-depth practices** to counter both classical and post-quantum adversaries.

---

### 9.1 Cryptographic Risk Analysis

QRB's stack replaces vulnerable classical algorithms with **post-quantum cryptography validated by NIST**.

| Threat | Classical Weakness | QRB Replacement |
|---|---|---|
| Private Key Extraction | ECDSA (vulnerable to Shor's algorithm) | CRYSTALS-Dilithium |
| Signature Forgery | ECDSA / RSA | SPHINCS+ (hash-based stateless signatures) |
| Encrypted Message Decryption | AES/SHA-2 weakened by Grover's algorithm | Kyber for PQ encryption |
| Contract Logic Spoofing | Forged tx metadata | SPHINCS+-anchored contracts |
| Replay Attacks | Nonce re-use + low entropy | PQ-entropy validators + VDFs |
| Randomness Manipulation | RNG from block hashes or predictable entropy | VDF-based Randomness Beacons |

Mitigations:

- PQ-20 Tokens Enforce Signature Validation At Every Lifecycle Point (mint, Burn, Transfer)
- Bridge Migration Proofchains Rely on Multi-Party Signature Fallback
- Contracts Are Anchored via Sphincs+ Metadata Binding to Prevent Tampering

---

## 9.2 Bridge Attack Surface & Mitigations

Bridges are the #1 exploit vector in DeFi today, with billions lost to signature forgery and multi-sig manipulation.

| Attack Vector | QRB Defense |
|---|---|
| Signature Spoofing | Dilithium-based proof signing |
| Validator Key Collusion | Distributed bridge signers + MPC fallback |
| Replay/Malleability | Per-bridge nonces + deterministic hashes |
| Oracle Manipulation | On-chain voting on state transitions |
| Chain Reorgs / Forking | Finality delays + VDF randomness synchronization |
| Economic / DoS Attacks | Fee markets, relayer incentives, and guaranteed time-locked withdrawals |

**Outcome:** QRB bridges are quantum-hardened, resilient against both **cryptographic exploits** and **economic manipulation**.

---

## 9.3 Validator and VDF Centralization Risks

**Risk 1: Validator Concentration**
If validator power concentrates, consensus can be hijacked.

- **Mitigations:**

    - Stake caps + slashing penalties.
    - Validator Academy to encourage diverse participation.
    - Dilithium signer rotation for cryptographic diversity.
    - **Delegation mechanisms** to distribute voting power while maintaining accessibility.

**Risk 2: VDF (Verifiable Delay Function) Centralization**
VDFs can create bottlenecks if controlled by few parties.

- **Mitigations:**
    - Open-source VDF implementations with community benchmarking.
    - Multi-supplier VDF runtime options.
    - Randomness attestation chains.
    - **Future goal:** Decentralized VDF networks where multiple nodes compute and cross-verify outputs.

---

## 9.4 Governance and Upgrade Security

**Risk: DAO Takeover or Malicious Proposal Execution**

**QRB's Protections:**

- Proposals and votes signed with **Dilithium**, anchored with **SPHINCS+**.
- Stateless governance contracts eliminate reliance on vulnerable on-chain keys.
- Emergency rollback provisions allow governance to **pause upgrades**.
- Multi-phase pipeline: draft → vote → timelock → execute.

**Upgrade Integrity:**

- Community-verified upgrade packages.
- On-chain hash commit-reveal for upgrade code.
- Validator quorum with Dilithium signatures required for activation.
- **Rollback Trigger Clarification:** Emergency rollback may be initiated via quorum-based validator signatures or expedited community vote, ensuring transparency and preventing unilateral control.

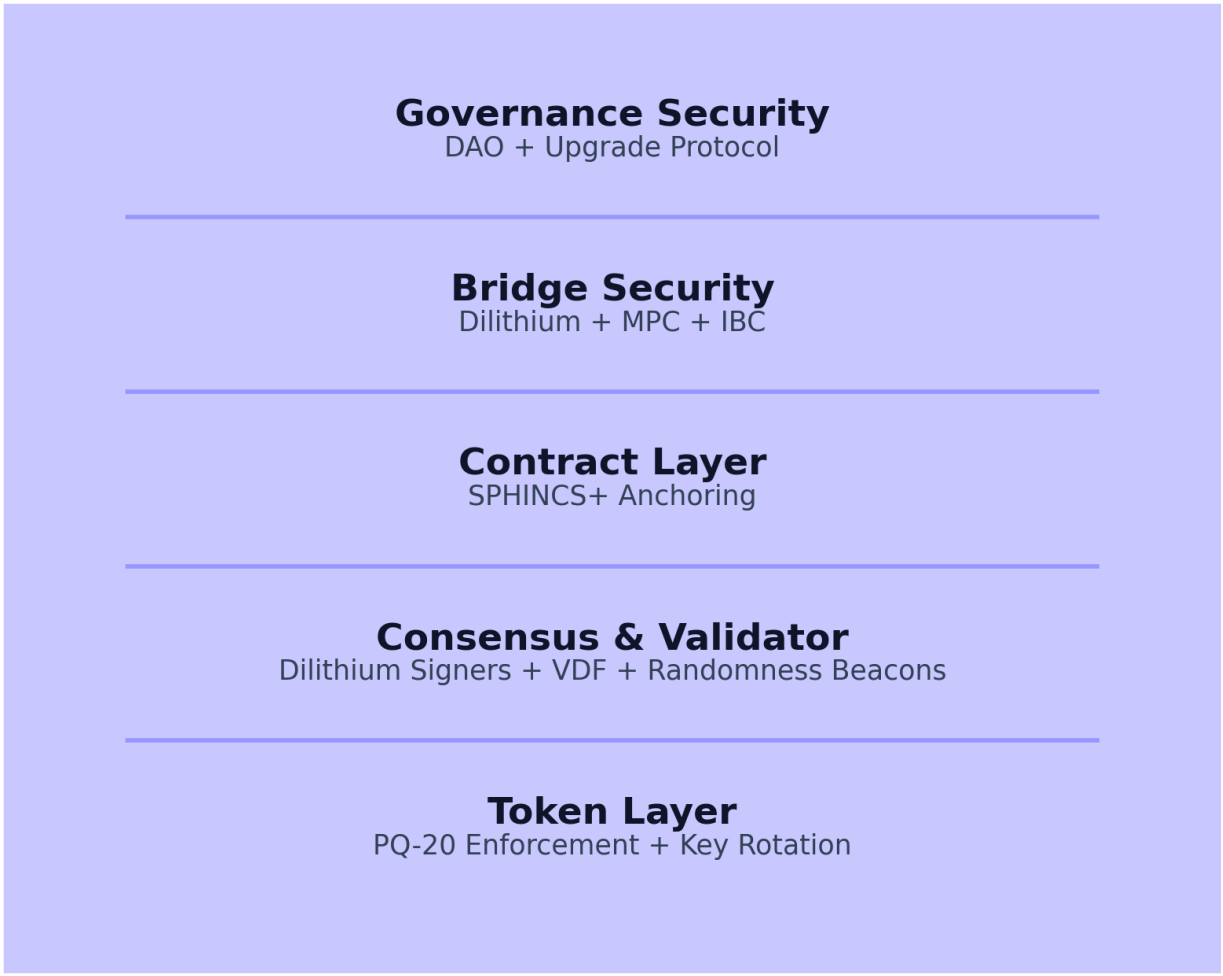
---

## Summary Table

| Domain | Threat | QRB Defense | Key Benefit |
|---|---|---|---|
| Cryptography | Shor/Grover attacks, randomness bias | Dilithium, SPHINCS+, Kyber, VDF randomness | PQ-resistant core primitives |
| Bridges | Forgery, collusion, DoS | Dilithium proofs, MPC fallback, fee markets | Secure cross-chain settlement |
| Validators / VDFs | Centralization bottlenecks | Stake caps, delegation, decentralized VDFs | Resilient consensus |
| Governance | DAO takeover, upgrade attacks | Dilithium votes, multi-phase pipeline, rollback | Trustworthy upgrade process |

---

Figure 5: Security Layers Stack




---

## 10. Developer & Ecosystem Tooling

The long-term success of any blockchain protocol depends on the strength of its builder community. QRB is engineered as a **developer-first** platform: production-ready SDKs, audited smart-contract templates, wallet integrations, and rollup/IBC tooling so teams can ship **quantum-resistant** apps without a steep learning curve.

---

## 10.1 PQ-20 SDK

**Purpose:** Make quantum-secure assets and cross-chain migration feel familiar to Web3 developers.

**Languages:** TypeScript/JavaScript, **Python**, Rust, Go.

**Core Features**

- **PQ-20 token interface:** mint, burn, transfer, approvals, metadata; chain-agnostic helpers.
- **Integrated signatures:** Dilithium and SPHINCS+ modules with safe defaults (domain separation, deterministic nonces).
- **Bridge clients:** high-level wrappers for burn-and-mint flows between Ethereum/BSC/Polygon ↔ QRB (with retry, idempotency, and finality polling).
- **Audit-ready utilities:** typed message builders, canonical hashing, replay protection, and **fallback paths** for degraded conditions.
- **Tooling:**
    - **CLI** for keygen, signing, and transaction simulation.
    - **Test vectors & simulators** for unit/integration testing.
    - Example apps and reference architectures.

**Availability:** Open-source packages (NPM, PyPI, Cargo, crates.io, Go modules) with versioned APIs, changelogs, and reproducible builds (SBOM + signature on releases).

---

## 10.2 Wallet Integration & Ledger Support

**Goal:** End-user accessibility without compromising cryptography.

**Planned Integrations**

- **MetaMask Snap (PQ-Snap):** custom Dilithium/SPHINCS+ signing workflows.
- **Mobile Wallet SDK:** native keygen & signing for React Native and Flutter.
- **Hardware Wallets:** Ledger Nano/X (firmware extension pending); HSM profiles for institutional custody.

**Functionality**

- Quantum-safe derivation paths and secure key storage.
- Full **PQ-20 support** (balances, transfers, bridging, governance).
- Multi-account / multi-wallet support, watch-only mode, and transaction previews with PQ verification.

---

## 10.3 Smart Contract Templates

**Purpose:** Accelerate secure dApp development with safe-by-default modules.

**Core Modules**

- **PQ-Governance:** stateless DAO voting; Dilithium-verified ballots; timelocks and proposal pipelines.
- **PQ-NFT:** SPHINCS+ metadata anchoring, royalty preservation, non-destructive migration.
- **PQ-Vault:** withdrawal policies with Dilithium auth, quorum logic, and rate limits.

- **Bridge-Compatible Contracts:** upgrade-safe patterns with cross-chain message verification.

**Roadmap Templates:** PQ-DEX liquidity pool skeleton, PQ-Lending primitives (oracle adapter, collateral manager).

**Availability:** Fully open-source and audited; shipped with usage guides, migration notes for Solidity/EVM teams, and formal-verification handoffs.

---

## 10.4 L2 and ZK Compatibility

**Objective:** Combine scalability and privacy with **post-quantum integrity**.

**Roadmap Integrations**

- ZK-rollup compatibility via PQ signature anchoring (SNARK/STARK wrappers).
- Zero-knowledge **governance**: private voting with public verifiability.
- IBC-compatible light clients for rollups and bridge finality.
- PQ-compatible execution environments for modular L2s.

**Concrete Example — Private Voting with ZK + PQ**
A voter holds a PQ-identity; they cast a vote signed with **SPHINCS+**. A **ZK-STARK** proves "eligible and not double-voting" and proves the signature's validity **without revealing identity or choice**. The rollup posts the proof; QRB verifies it and records the tally — **privacy plus quantum-safe authenticity**.

---

## 10.5 Developer Experience & Infra

- **Dev Portal:** docs, tutorials, API references, and copy-paste snippets.
- **Localnet/Devnet:** Docker compose + Helm charts, faucets, block explorers.

- **Testing:** official fixtures, deterministic harness, example CI templates (GitHub Actions) with security checks.
- **Security Toolchain:** package signing, SBOMs, lint rules, dependency allowlists, and break-glass procedures for critical fixes.
- **Observability:** sample dashboards/logging for bridge clients and validators.

---

### Developer Ecosystem Commitment

Builders **will have access to**:

- **Grants program** for core tooling, audits, and ecosystem apps.
- **Technical Ambassador network** (office hours, code reviews, workshops).
- **Continuous security audit partnerships** and a **public bug bounty**.
- Regular hackathons and migration support for projects moving from EVM chains.

---

## Summary Table

| Deliverable | Scope / Notes | Phase |
|---|---|---|
| PQ-20 SDK (TS, Python, Rust, Go) | Tokens, signatures, bridge clients, CLI, test vectors | 4 |
| Wallet Integrations | PQ-Snap, Mobile SDK, Ledger/HSM profiles | 4 |
| Contract Templates | PQ-Governance, PQ-NFT, PQ-Vault, Bridge-Compatible | 4 |
| ZK + PQ Toolkit | Rollup wrappers, light clients, private voting example | 5 |
| PQ-DEX / PQ-Lending Seeds | Reference modules for DeFi builders | 5 |

---

# 11. Governance Framework

QRB governance balances security, decentralization, and delivery speed. It combines on-chain Dilithium-signed voting with stateless proposal execution, timelocks, and emergency safeguards to minimize governance risk.

---

## 11.1 Principles

- Security-first and transparent by default.
- Minimize persistent on-chain keys; favor stateless verification.
- Progressive decentralization with clear milestones.
- Community legitimacy via open audits and public artifacts.

---

## 11.2 Roles

- Token holders: vote and delegate; proposal sponsorship.
- Validators: enforce finality, sign upgrades, execute rollbacks under quorum.
- Maintainers: coordinate audits, operate dev tooling, publish releases.
- Ambassadors: education, integrations, and ecosystem support.

---

## 11.3 Processes

- Proposal lifecycle: draft → review → vote → timelock → execute.
- Emergency pause/rollback: quorum-based validator signatures or expedited community vote.
- Upgrade transparency: commit–reveal of code hash; signed SBOM and release notes.

---

## 11.4 Voting & Quorum

Stake-weighted Dilithium ballots; quorum and supermajority thresholds configured per proposal type.

---

### 11.5 Upgrades & Rollbacks

Multi-stage upgrades with reproducible builds; rollbacks require explicit quorum and public incident report.

---

### 11.6 Risk & Audit

Independent audits, bug bounty, and post-incident reviews; all artifacts retained for public verification.

---

## 12. Conclusion

QRB delivers end-to-end quantum resilience without abandoning Web3 composability. By aligning post-quantum primitives (Dilithium, SPHINCS+, Kyber) with secure bridges, governance, and developer tooling, QRB offers a credible path to protect assets, NFTs, and DeFi through and beyond the quantum transition.

---

# 13. Appendices

## A. Glossary

| Term | Definition |
|---|---|
| Post-Quantum Cryptography (PQC) | Cryptographic schemes designed to resist quantum attacks such as Shor's and Grover's. |
| CRYSTALS-Dilithium | NIST-selected lattice-based digital signature used for wallets, validators, and bridge proofs. |
| SPHINCS+ | NIST-selected stateless hash-based signature used for high-assurance operations (e.g., governance, contract anchoring). |
| Kyber (KEM) | Post-quantum key-encapsulation mechanism for establishing shared secrets and secure channels. |
| PQ-20 | QRB's quantum-secure token standard enforcing signature validation, lifecycle hooks, and provenance metadata. |
| QBT / PQ-QBT | QBT: ERC-20 bridge token on legacy chains; PQ-QBT: 1:1 redeemed token on QRB after migration. |
| QuantumBridge | Two-phase migration (lock/burn → redeem/mint) using Dilithium proofs with MPC fallback in worst-case events. |
| IBC (Inter-Blockchain Communication) | Cross-chain messaging protocol; QRB uses PQ-hardened and planned ZK-validity light clients. |
| VDF (Verifiable Delay Function) | Function costly to compute but easy to verify; used for unbiased randomness and replay resistance. |
| Randomness Beacon | Publicly verifiable source of randomness resistant to grinding via VDFs. |
| MPC (Multi-Party Computation) | Securely computes signatures or recoveries across multiple participants without sharing private keys. |
| Threshold Signatures | t-of-n signatures enabling shared control; targeted for validator committees and custody policies. |
| Stateless Governance | Pattern avoiding persistent on-chain keys; proposals/ballots signed off-chain and verified on-chain. |
| Harvest-Now, Decrypt-Later | Adversaries record ciphertexts now to decrypt when quantum capability matures. |
| Light Client | Compact verifier of another chain's state and proofs. |
| Finality | Point after which blocks are economically or |

| | |
|---|---|
| | cryptographically irreversible. |
| Replay/Malleability (Bridge) | Reusing/altering proofs across contexts; mitigated by nonces and canonical hashes. |
| NFT Provenance | Cryptographic history of ownership and metadata; anchored via SPHINCS+ on QRB. |
| Escrow-Based Migration | Non-destructive NFT migration preserving royalties and lineage. |
| Commit-Reveal | Two-step integrity: publish hash (commit), then reveal value for verification. |
| ZK-SNARK / ZK-STARK | Zero-knowledge proof systems for private verification without revealing inputs. |
| DAO | Decentralized Autonomous Organization; votes Dilithium-signed, contracts SPHINCS+-anchored. |
| Emergency Rollback | Governance-controlled pause/revert with quorum and transparency rules. |
| SBOM | Software Bill of Materials; signed manifests for SDK/tooling releases. |
| HSM | Hardware Security Module used for secure key storage and PQ signing. |

---

## B. Cryptographic Primer

- Dilithium (Signatures): lattice-based, efficient, NIST-selected.
- SPHINCS+ (Signatures): hash-based, stateless; larger sizes but strongest assumptions.
- Kyber (KEM): key exchange/encryption for hybrid channels.
- VDFs & Beacons: verifiable delay prevents randomness grinding.
- ZK Proofs: SNARK/STARK wrappers provide privacy with PQ signatures.

---

## C. Key Contract Interfaces

Interface highlights (illustrative):

• IPQ20: mint(address to, uint256 amt), burn(uint256 amt), transfer(address to, uint256 amt)

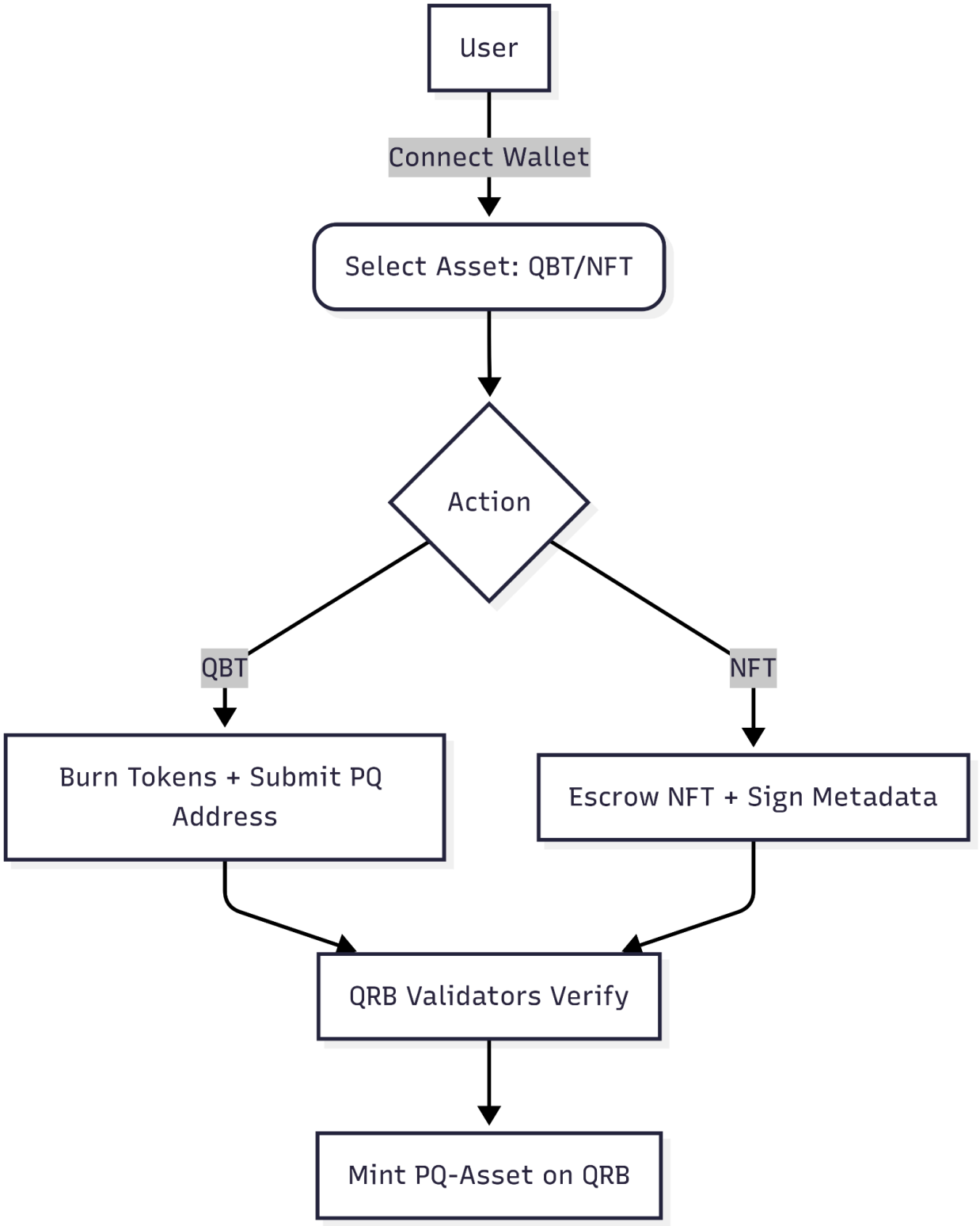
• IPQGovernance: propose(bytes32 hash), vote(uint256 id, bool support), queue(uint256 id), execute(uint256 id)

• IPQNFT: mintWithAnchor(bytes metaHash, address to), setRoyalty(address artist, uint16 bps)

• IPQBridgeClient: lock(bytes proof), redeem(bytes proof), setFinality(uint32 slots)

---

D. Migration Flow Diagram



## E. Technical Paper References

- NIST PQC Standards: CRYSTALS-Dilithium, SPHINCS+, Kyber.
- IBC: Inter-Blockchain Communication Protocol (core spec).
- VDFs: Verifiable Delay Functions for consensus randomness.
- ZK-SNARKs/STARKs: private verification frameworks.
- Bridge Security: multisig risks, proof malleability, economic DoS.

---

# Disclaimer & Forward-Looking Statements

This whitepaper is provided for informational purposes only and does not constitute investment, legal, accounting, or tax advice, nor an offer to sell or a solicitation to buy any asset. References to tokens (e.g., QBT, PQ-QBT) describe technical functionality, not financial characteristics.

Forward-Looking Statements. This document may contain statements including words such as “anticipate,” “believe,” “estimate,” “expect,” “intend,” “may,” “plan,” “project,” “should,” and similar expressions. These are subject to risks and uncertainties. Actual results may differ materially. QRB undertakes no obligation to update such statements.

Security & Cryptography. Post-quantum cryptography and its implementations evolve quickly. If best practices, NIST profiles, or vetted parameter sets change, QRB may adopt revisions (e.g., signature/KEM parameters, serialization, key-rotation procedures). Where changes impact users or developers, migration paths and deprecation timelines will be published.

Bridges & Economic Risks. Cross-chain activity involves technical and market risks (e.g., relayer incentives, fee markets, finality delays). QRB’s defense-in-depth approach reduces, but cannot eliminate, such risks.

Regulatory. Regulatory treatment of digital assets varies by jurisdiction and may change. Builders and users are responsible for compliance with local laws.

No Warranties. Materials are provided “as is,” without warranties of any kind, express or implied.

Responsible Disclosure. Security issues can be reported to security@<your-domain>. A public bug-bounty program may apply to eligible findings.

---