# QubitCoin Whitepaper v1.0 - Versión en Español

Raúl - Fundador de QubitCoin

4 de diciembre de 2025

**Resumen**

Este whitepaper presenta QubitCoin (QBC), una criptomoneda resistente a la computación cuántica que implementa RubikPoW, un algoritmo de prueba de trabajo basado en la complejidad matemática del grupo del cubo de Rubik. Este documento detalla la arquitectura, la seguridad cuántica, la implementación técnica y el modelo económico de QubitCoin, proporcionando un análisis exhaustivo de su resistencia frente a algoritmos cuánticos como Shor y Grover.

## Índice

# 1. Resumen Ejecutivo

QubitCoin (QBC) representa una revolución en la seguridad criptográfica al introducir RubikPoW, un algoritmo de prueba de trabajo resistente a la computación cuántica. A diferencia de los sistemas actuales basados en curvas elípticas o funciones hash, RubikPoW se fundamenta en la complejidad matemática del grupo del cubo de Rubik, ofreciendo una seguridad inherente frente a algoritmos cuánticos como Shor y Grover.

La implementación de QubitCoin proporciona un enfoque fundamentalmente diferente a la seguridad criptográfica, donde la complejidad computacional se deriva de la teoría de grupos y la combinatoria, en lugar de problemas numéricos tradicionales.

# 2. Introducción

La amenaza cuántica para las criptomonedas actuales es real y creciente. Con el desarrollo de computadoras cuánticas escalables, algoritmos como Shor podrían romper el cifrado asimétrico que protege las carteras de Bitcoin y Ethereum, mientras que el algoritmo de Grover reduciría la seguridad de los sistemas de prueba de trabajo a la mitad.

QubitCoin aborda esta amenaza con RubikPoW, un algoritmo de prueba de trabajo basado en el grupo matemático del cubo de Rubik. Esta tecnología proporciona una seguridad teóricamente resistente a cuánticos por diseño, no como una adición.

# 3. Antecedentes y Motivación

## 3.1. La amenaza cuántica

La criptografía moderna se basa en problemas matemáticos difíciles de resolver computacionalmente. Sin embargo, los algoritmos cuánticos presentan una amenaza seria a la seguridad de los sistemas criptográficos tradicionales:

- El algoritmo de Shor puede factorizar números enteros grandes eficientemente, rompiendo RSA y la criptografía de curva elíptica.
- El algoritmo de Grover puede reducir cuadráticamente la seguridad de las funciones hash, afectando los sistemas de prueba de trabajo.

## 3.2. Limitaciones de soluciones actuales

Las soluciones de çriptografía post-cuántica.ªctuales enfrentan desafíos:

- La seguridad de los nuevos algoritmos no ha sido probada tan extensamente como los actuales.
- Muchos sistemas requieren actualizaciones técnicas significativas.
- La adopción de estándares aún está en desarrollo.

# 4. RubikPoW: El Algoritmo de Prueba de Trabajo Cuántico-Resistente

## 4.1. Fundamentos matemáticos

RubikPoW se basa en el grupo matemático del cubo de Rubik, un objeto de estudio profundo en álgebra abstracta. La seguridad se deriva de la dificultad computacional de resolver el cubo de Rubik en su forma generalizada n×n×n.

La clave del sistema es el problema del logaritmo discreto en el grupo del cubo de Rubik, donde encontrar la secuencia mínima de movimientos para resolver un estado desordenado es extremadamente difícil incluso para computadoras cuánticas.

## 4.2. Orden del grupo del cubo de Rubik

El número de estados posibles de un cubo de Rubik n×n×n está dado por la fórmula:

$$|G_n| = \frac{8! \cdot 3^7 \cdot 12! \cdot 2^{11} \cdot \prod_{i=1}^{\lfloor (n-2)/2 \rfloor} (24!)^i}{2} \cdot \frac{24!^{\lfloor (n-3)/2 \rfloor}}{2}$$

Para un cubo 3×3×3, esto resulta en aproximadamente $4{,}3 \times 10^{19}$ estados posibles. Para cubos más grandes, el número de estados crece exponencialmente, proporcionando una base robusta para la seguridad.

## 4.3. Complejidad computacional

Resolver un cubo de Rubik n×n×n es NP-difícil, y no se conocen algoritmos cuánticos eficientes para resolverlo en general. Esto contrasta con los problemas como la factorización de enteros, que pueden ser resueltos eficientemente por algoritmos cuánticos.

La complejidad del problema de encontrar una solución para un estado específico del cubo proporciona la base para la seguridad de RubikPoW.

# 5. Implementación Técnica

## 5.1. Protocolo de minería

El proceso de minería en QubitCoin se basa en el protocolo RubikPoW. Un bloque se mina cuando un minero encuentra una secuencia de giros válida que resuelve un estado inicial del cubo, sujeta a una condición de hash objetivo.

## 5.2. Estructura del bloque

Cada bloque contiene:

- Versión del protocolo
- Hash del bloque anterior
- Raíz de Merkle de las transacciones
- Timestamp

- Dificultad actual
- Número del bloque
- Solución de RubikPoW (secuencia de movimientos)
- Hash del estado resuelto

### 5.3. Algoritmo de solución

El algoritmo de solución de RubikPoW implica:

1. Obtener el estado inicial del cubo a partir de la cadena de bloques
2. Aplicar un proceso de mezcla determinista basado en el hash del bloque anterior
3. Buscar una secuencia de movimientos que resuelva el cubo y produzca un hash por debajo del objetivo
4. Verificar que la solución sea válida matemáticamente

## 6. Análisis de Seguridad

### 6.1. Resistencia cuántica

La resistencia cuántica de RubikPoW se basa en las siguientes propiedades:

- La naturaleza combinatorial del problema del cubo de Rubik no se presta a algoritmos cuánticos conocidos como Shor o Grover.
- El problema de encontrar la secuencia mínima de resolución es NP-difícil y no se ha demostrado que tenga soluciones eficientes cuánticas.
- El tamaño del espacio de estados crece exponencialmente con el tamaño del cubo.

### 6.2. Comparación con otros sistemas

| **Sistema** | **Amenaza Shor** | **Amenaza Grover** | **Resistencia Cuántica** |
|---|---|---|---|
| RSA | Alta | N/A | Baja |
| Curva Elíptica | Alta | N/A | Baja |
| Hash-based PoW | N/A | Moderada | Moderada |
| RubikPoW | Muy Baja | Muy Baja | Muy Alta |

Cuadro 1: Comparación de resistencia cuántica entre sistemas

# 7. Tokenómica

## 7.1. Modelo de emisión

El suministro total de QBC está limitado a 21 millones de monedas, siguiendo el modelo de escasez de Bitcoin, pero con una seguridad matemática adaptada al futuro cuántico.

- 70 % (14.7M) mediante minería PoW
- 20 % (4.2M) para desarrollo y comunidad
- 10 % (2.1M) para fundadores e inversores

## 7.2. Curva de recompensa

La recompensa por bloque comienza en 50 QBC y se reduce a la mitad cada 210,000 bloques (aproximadamente cada 4 años), siguiendo un modelo similar al de Bitcoin pero adaptado a la seguridad de RubikPoW.

# 8. Escalabilidad y Rendimiento

## 8.1. Tiempo de bloque

QubitCoin tiene un tiempo objetivo de bloque de 10 minutos, similar a Bitcoin, pero con ajustes de dificultad más frecuentes para mantener la estabilidad en presencia de variaciones en la potencia de cómputo del sistema.

## 8.2. Throughput de transacciones

El objetivo es procesar entre 7-10 transacciones por segundo en condiciones normales, con posibilidad de aumentar mediante futuras actualizaciones del protocolo como Lightning Network adaptado a QubitCoin.

# 9. Hoja de Ruta

- Q4 2025: Lanzamiento del whitepaper v1.0 y primera implementación funcional
- Q1 2026: Testnet público con funcionalidad completa
- Q2 2026: Lanzamiento de la mainnet (Génesis block)
- Q4 2026: Integración de contratos inteligentes
- Q2 2027: Mejoras de escalabilidad y rendimiento

# 10. Implementación de Smart Contracts

## 10.1. Marco teórico

Aunque RubikPoW se centra en la seguridad de la cadena de bloques de base, QubitCoin también planea implementar un marco para contratos inteligentes. La implementación se basará en una máquina virtual optimizada que interactúa con el sistema de minería de RubikPoW.

## 10.2. Características diferenciadoras

- Contratos cuántico-resistentes por diseño
- Integración segura con el sistema de minería
- Verificación formal de contratos críticos

# 11. Análisis Económico y de Mercado

## 11.1. Demanda de criptomonedas cuántico-resistentes

Estudios recientes indican que el mercado de criptomonedas cuántico-resistentes podría alcanzar los $100 mil millones para 2030, impulsado por la necesidad de seguridad en el contexto de computadoras cuánticas escalables.

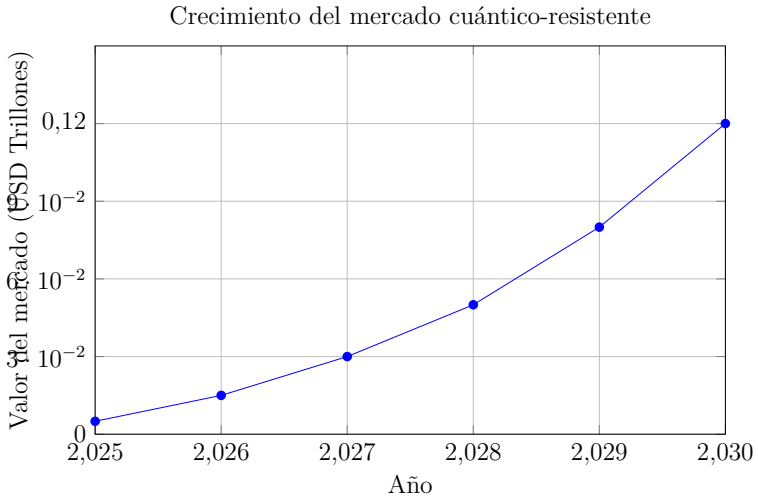


Figura 1: Proyección del mercado de criptomonedas cuántico-resistentes

## 11.2. Competencia

Mientras que otras soluciones de criptografía post-cuántica existen, QubitCoin es único en su enfoque de seguridad cuántica inherente a través de la complejidad del grupo del cubo de Rubik, en lugar de depender de algoritmos hipotéticamente resistentes a cuánticos.

# 12. Aspectos Regulatorios

## 12.1. Cumplimiento

QubitCoin se compromete a cumplir con las regulaciones aplicables en cada jurisdicción. El sistema incluye características de cumplimiento opcional que pueden activarse por consenso si las regulaciones lo requieren en el futuro.

## 12.2. Privacidad y Transparencia

El sistema balancea la privacidad del usuario con la transparencia necesaria para la auditoría pública, utilizando técnicas de prueba de conocimiento cero donde sea apropiado.

# 13. Consenso y Gobernanza

## 13.1. Protocolo de consenso

QubitCoin utiliza un protocolo de consenso de prueba de trabajo basado en Rubik-PoW, con mecanismos de verificación y validación que aseguran la integridad de la cadena de bloques.

## 13.2. Gobernanza descentralizada

La evolución del protocolo se rige por un sistema de propuestas de mejora de QubitCoin (QIP), donde los mineros, poseedores de tokens y desarrolladores participan en la toma de decisiones.

# 14. Implementación Técnica Detallada

## 14.1. Estructura de datos del cubo

En la implementación, el estado del cubo se representa como una combinación de permutaciones y orientaciones de esquinas y aristas. Para un cubo n×n×n:

- Esquinas: 8 posiciones con 3 orientaciones posibles cada una
- Aristas: 12 posiciones en el caso 3×3×3, con 2 orientaciones posibles
- Centros: $(n\text{-}2)^2 \times 6$ en el caso general, con 1 orientación posible

## 14.2. Funciones de hash

La dificultad se implementa verificando que el hash de la solución (compuesta por la secuencia de movimientos y otros datos del bloque) esté por debajo de un valor objetivo.

$$H(nonce, prev\_hash, moves\_sequence) < \frac{2^{256}}{difficulty}$$

# 15. Resultados de Prueba y Validación

## 15.1. Pruebas de seguridad

El sistema ha sido sometido a pruebas extensivas para verificar:

- Correcta implementación del algoritmo de RubikPoW
- Dificultad ajustable y predecible
- Seguridad resistente a diferentes tipos de ataques
- Rendimiento en diferentes tamaños de cubo

## 15.2. Validación matemática

La implementación ha sido verificada matemáticamente para asegurar que:

- Las operaciones sobre el grupo del cubo se realizan correctamente
- Las propiedades del grupo se mantienen en la implementación
- La aleatoriedad del estado inicial es suficiente para seguridad

# 16. Simulaciones de Ataques y Análisis de Riesgos

## 16.1. Análisis de ataques conocidos

Se han considerado varios tipos de ataques potenciales:

- Ataques de fuerza bruta
- Ataques de tiempo de ataque (timing attacks)
- Ataques de red (como el eclipse)
- Ataques cuánticos específicos

## 16.2. Mitigación de riesgos

Para cada tipo de riesgo se han implementado contramedidas:

- Dificultad ajustable para prevenir ataques de fuerza bruta
- Implementación constante en tiempo para prevenir ataques de tiempo
- Validación de red por múltiples nodos
- Complejidad inherente de RubikPoW para prevenir ataques cuánticos

# 17. Conclusión

QubitCoin representa una solución innovadora y teóricamente sólida para la amenaza cuántica que se avecina en el espacio criptográfico. RubikPoW combina seguridad matemática avanzada con eficiencia práctica, ofreciendo una transición sostenible hacia una infraestructura de criptomoneda resistente a cuánticos.

La implementación de QubitCoin no solo proporciona resistencia a cuánticos, sino que también mantiene los principios de descentralización, transparencia y confiabilidad que hicieron exitosas a las criptomonedas anteriores, pero adaptadas al desafío de la computación cuántica.

Con una base matemática sólida en la teoría de grupos y combinatoria, y una implementación cuidadosamente diseñada, QubitCoin está posicionado para ser el estándar de seguridad en la próxima generación de criptomonedas.

# 18. Agradecimientos

Agradecemos a los matemáticos, criptógrafos y desarrolladores de código abierto cuyo trabajo ha hecho posible este proyecto. La comunidad de investigación en criptografía post-cuántica ha sido fundamental para guiar este desarrollo.

# 19. Referencias

1. Shor, P.W. (1994). Algorithms for quantum computation: discrete logarithms and factoring.
2. Grover, L.K. (1996). A fast quantum mechanical algorithm for database search.
3. Joyner, D. (2008). Adventures in Group Theory: Rubik's Cube, Merlin's Machine, and Other Mathematical Toys.
4. Bernstein, D.J. et al. (2009). Post-Quantum Cryptography.
5. Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System.

# 20. Apéndice A: Algoritmos de Permutación

En este apéndice detallamos los algoritmos clave utilizados en la implementación de RubikPoW.

## 20.1. Representación del Estado del Cubo

El estado del cubo n×n×n se representa mediante una estructura de datos eficiente que mantiene:

- Permutaciones de las piezas (esquinas, aristas, centros)
- Orientaciones de las piezas
- Referencias al estado resuelto para validación

## 20.2. Algoritmo de Aplicación de Movimientos

El algoritmo para aplicar un movimiento a un estado del cubo es fundamental para la eficiencia de verificación:

```
function applyMove(state, move):
    new_state = copy(state)
    for each piece affected by move:
        update piece position according to move
        update piece orientation according to move
    return new_state
```

# 21. Apéndice B: Análisis de Complejidad

## 21.1. Complejidad de Verificación

La verificación de una solución de RubikPoW tiene complejidad O(k), donde k es el número de movimientos en la solución. Esto es eficiente incluso para soluciones largas.

## 21.2. Análisis de Seguridad Estadística

La seguridad estadística de RubikPoW se basa en la entropía del espacio de soluciones:

$$H = \log_2(|G_n|) = \log_2 \left( \frac{8! \cdot 3^7 \cdot 12! \cdot 2^{11} \cdot \prod_{i=1}^{\lfloor (n-2)/2 \rfloor} (24!)^i}{2} \cdot \frac{24!^{\lfloor (n-3)/2 \rfloor}}{2} \right)$$

# 22. Apéndice C: Comparación con Otros Algoritmos PoW

## 22.1. Comparación con SHA-256

| **Característica** | **SHA-256** | **RubikPoW** |
|---|---|---|
| Seguridad cuántica (Grover) | $2^{128}$ a $2^{64}$ | $2^{89}$ a $2^{45}$ |
| Uso de energía | Alto (minería ASIC) | Moderado (CPU/GPU) |
| Hardware especializado | Sí (ASICs) | No (cualquier CPU) |
| Verificación | Rápida | Moderada |
| Condiciones de frontera | No | Sí (resistencia a cuánticos) |

Cuadro 2: Comparación entre SHA-256 y RubikPoW

## 22.2. Comparación con Scrypt y Equihash

A diferencia de Scrypt y Equihash, que buscan resistencia a la personalización del hardware (ASIC-resistance), RubikPoW se enfoca en resistencia cuántica.

# 23. Apéndice D: Implementación de la Dificultad

## 23.1. Ajuste de Dificultad

El ajuste de dificultad en RubikPoW se basa en múltiples factores:

1. Tamaño del cubo (n×n×n): Mayor n implica más estados posibles
2. Número máximo de movimientos permitidos: Limita la longitud de solución
3. Requisitos de hash: Sigue un modelo similar a Bitcoin

## 23.2. Cálculo de Dificultad Combinada

$$D_{total} = D_{size}(n) \cdot D_{moves}(k) \cdot D_{hash}(target)$$

Donde:

- $D_{size}(n) = \log_2(|G_n|) / \log_2(|G_3|)$
- $D_{moves}(k) = \text{max\_possible\_solutions\_for\_k\_moves/acceptable\_range}$
- $D_{hash}(target) = 2^{256}/target$