KUBU PROTOCOL

WHITEPAPER

Native Nickname Layer for Proof-of-Work Payments

Technical and economic specification of the KUBU network, nickname protocol, payment memo flow, and operational model.

Version 1.0.0

Status: Final

Contents

- Abstract
- 1. Design Goals
- 2. Core Network Specification
  - 2.1 Identity
  - 2.2 Consensus
  - 2.3 Capacity Limits
  - 2.4 Mainnet Parameters
  - 2.5 Address Encoding (Mainnet)
- 3. Monetary Policy
  - 3.1 Block Subsidy
  - 3.2 Emission Behavior
  - 3.3 Fees and Relay Policy
- 4. Native Nickname Protocol
  - 4.1 Nickname Rules
  - 4.2 Lifecycle Durations
  - 4.3 State Machine
  - 4.4 On-Chain Nickname Operations
- 5. Nickname Economics
  - 5.1 Base Registration Price and Bond
  - 5.2 Dynamic Pricing Multiplier
  - 5.3 Renewals
  - 5.4 Who Gets What
- 6. Nickname Payments and Memo
  - 6.1 Memo Format
  - 6.2 Wallet/RPC Support
  - 6.3 Memo Requirement Semantics
- 7. Security Model
  - 7.1 Ownership Control
  - 7.2 Anti-Squatting Mechanisms
  - 7.3 Mempool and Conflict Safety
  - 7.4 Checkpoints
- 8. User Flows
  - 8.1 Basic User
  - 8.2 Exchange / Merchant
- 9. Operational Notes
  - 9.1 Network Environments

  - 9.2 Node Discovery
- 10. Limitations and Future Work
- 11. Conclusion

Version: 1.0.0
Status: Final

Abstract

KUBU is a Proof-of-Work cryptocurrency with native, protocol-level nicknames. Instead of sharing long wallet addresses, users can register short human-readable names (for example, `@shop_kubu`) and receive funds through them.

KUBU is designed around three priorities: - practical payments for normal users, - sustainable miner incentives, - anti-squatting protection for nickname space.

The nickname layer is not a centralized service: it is validated by full nodes and stored in chain state.

1. Design Goals

1. Human-friendly payments: allow sending to `@nickname` instead of a long address.
2. Open participation: keep entry-level nickname prices affordable, especially for long names.
3. Anti-squatting: make mass reservation expensive through locked bonds and dynamic pricing.
4. Miner alignment: registration fees are miner fees; network usage directly rewards block producers.
5. Compatibility: preserve standard UTXO behavior and RPC wallet workflows.
6. Exchange readiness: support nickname + memo payment flows for merchant and exchange integrations.

2. Core Network Specification

2.1 Identity

- Name: `KUBU`
- Ticker / unit: `KUBU`
- Client version: `1.0.0.0`

2.2 Consensus

- Base model: Nakamoto consensus (longest/heaviest valid chain)
- PoW hash: `scrypt_1024_1_1_256`
- Difficulty retarget style: Digishield-enabled

- Block target spacing: `54 seconds`
- Mainnet block target timespan: `54 seconds` (rapid adjustment profile)
- Coinbase maturity:
- early branch: `30 blocks`
- Digishield branch (effective from height 1000): `240 blocks`
- AuxPoW / merged mining:
- Chain ID: `63` (`0x003f`)
- Activation height: `39000`
- Strict chain ID enforcement: enabled

2.3 Capacity Limits

- Max block serialized size: `4,000,000 bytes`
- Max block weight: `4,000,000`
- Max base block size: `1,000,000 bytes`
- Max sigops cost per block: `80,000`

2.4 Mainnet Parameters

- Network magic bytes: `d3 c4 b5 a6`
- P2P port: `45874`
- RPC port: `45873`
- Genesis hash: `0xb995d8d81cfb8c8cff5829b23cbff8ff4b34347f1c48c7e098ea7fb416a56951`
- Genesis timestamp: `2026-03-02 17:54:56 UTC`
- DNS seed: `seeds.kubu.supply`
- Fixed seeds:
- `81.91.177.17:45874`
- `5.8.248.76:45874`
- `5.8.248.229:45874`

2.5 Address Encoding (Mainnet)

- P2PKH prefix byte: `45` (addresses start with `K`)
- P2SH prefix byte: `22`
- WIF prefix byte: `158`

3. Monetary Policy

3.1 Block Subsidy

Halving interval: `1,168,000 blocks` (about 730 days at 54s/block).

Reward schedule: - Era 0: `48 KUBU` - Era 1: `24 KUBU` - Era 2: `12 KUBU` - Era 3: `6 KUBU` - Era 4+: `2 KUBU` tail emission

3.2 Emission Behavior

- Pre-tail issuance (first four eras): `(48 + 24 + 12 + 6) * 1,168,000 = 105,120,000 KUBU`
- Tail emission continues at `2 KUBU/block` after era 3.

This means KUBU uses a perpetual security budget model (finite high-subsidy eras, then stable low inflation).

Note: `MAX_MONEY = 800,000,000 KUBU` in code is a value-range safety constant, not a hard total-supply cap.

3.3 Fees and Relay Policy

- Recommended min tx fee: `0.01 KUBU`
- Default min relay fee: `0.001 KUBU/kB`
- Incremental relay fee: `0.0001 KUBU/kB`

4. Native Nickname Protocol

4.1 Nickname Rules

A valid nickname must: - be 4 to 16 characters, - use only lowercase ASCII letters, digits, _, - not start or end with _, - not contain __ consecutively, - include at least one letter.

Nicknames are normalized to lowercase before validation and storage.

4.2 Lifecycle Durations

At 54-second blocks: - Active period: `144,000 blocks` (about 90 days) - Grace period: `14,400 blocks` (about 9 days)

4.3 State Machine

States: - `ACTIVE` - `EXPIRED_GRACE` - `BOND_CLAIMABLE` - `EXPIRED_AVAILABLE` - `RELEASED`

Behavior summary: - During `ACTIVE`, nickname resolves normally. - During `EXPIRED_GRACE`, owner can still manage/renew. - After grace ends, bond can be claimed (`BOND_CLAIMABLE`). - Once bond is claimed after expiry, name becomes `EXPIRED_AVAILABLE` (re-registrable). - If owner manually frees name and then claims bond, historical terminal status is `RELEASED`.

4.4 On-Chain Nickname Operations

Operation payload marker: `KNA1` (stored in `OP_RETURN`)

Supported operations: - `REGISTER` - `UPDATE` (payout address) - `TRANSFER` (new owner pubkey) - `RENEW` - `RELEASE` (free nickname) - `CLAIM_BOND`

Protocol protections include: - max one nickname operation per transaction, - coinbase cannot include nickname operations, - owner authorization input required for mutable operations, - bond outpoint constraints (must be preserved/spent as required by op type), - mempool rule: one pending operation per nickname at a time.

5. Nickname Economics

5.1 Base Registration Price and Bond

By normalized length: - length 4: fee `24`, bond `48` KUBU - length 5: fee `12`, bond `24` KUBU - length 6: fee `6`, bond `12` KUBU - length 7: fee `3`, bond `6` KUBU - length 8+: fee `1`, bond `3` KUBU

5.2 Dynamic Pricing Multiplier

KUBU scales nickname prices by a rolling demand multiplier.

Parameters: - default multiplier: `1000 permille` (1.00x) - bounds: `500..3000 permille` (0.50x..3.00x) - rolling window: `11,200 blocks` (about 7 days) - adjustment epoch: `1,600 blocks` (about 1 day) - target demand: `280 registrations / 7 days` - smoothing alpha: `20%` - per-epoch max step: `10%`

Intuition: - if registrations exceed target, multiplier trends upward; - if registrations drop below target, multiplier trends downward; - step limit avoids sudden price shocks.

5.3 Renewals

- Renewal fee basis: `25%` of current registration fee.
- In current implementation, renewal cost is added to locked bond (`bond increase`), not treated as required protocol fee.

5.4 Who Gets What

- Registration fee requirement is enforced as transaction fee, so it is paid to miners.
- Bond is not miner revenue; it is locked collateral controlled by protocol rules and later claimable by owner.
- Transfer/update/release/claim still pay ordinary network tx fees set by sender (like any normal transaction).

6. Nickname Payments and Memo

KUBU supports sending directly to nicknames and optionally attaching a structured memo payload.

6.1 Memo Format

Memo payload marker: `KMEM1`

Limits and types: - max memo data: `48 bytes` - `memo_type`: - `numeric` - `alnum` - `utf8`

6.2 Wallet/RPC Support

Relevant RPC methods: - `sendtonickname` - `encodenicknamememo` - `decodenicknamememo` - `resolvenickname`

Nickname lifecycle RPC: - `checknickname` - `getnicknameinfo` - `listnicknames` - `listwalletnicknames` - `registernickname` - `updatenickname` - `transfernickname` - `renewnickname` - `releasenickname` - `claimnicknamebond`

6.3 Memo Requirement Semantics

A URI can signal `req-memo=1` at wallet/application level.
This is an integration rule for UX and processors; base consensus does not globally reject a nickname payment that omits memo.

7. Security Model

7.1 Ownership Control

Nickname ownership is tied to public key authorization and validated through spending inputs controlled by owner key scripts.

7.2 Anti-Squatting Mechanisms

- Length-tier pricing (short names cost more).
- Lockup bond for each active registration.
- Dynamic multiplier reacts to registration pressure.

7.3 Mempool and Conflict Safety

- Parallel pending operations on same nickname are blocked.
- Bond outpoint misuse is rejected.
- Unauthorized owner changes are rejected by pubkey and spend checks.

7.4 Checkpoints

Current network configuration keeps only genesis checkpoint entries in chain parameters.

8. User Flows

8.1 Basic User

1. Register nickname.
2. Use nickname for incoming payments.
3. Renew before active/grace window expires if needed.
4. Update payout address when rotating receiving wallet path.
5. Free nickname or claim bond when no longer needed.

8.2 Exchange / Merchant

1. Keep platform nickname (for example `@exchange`).
2. Generate per-payment memo and required confirmations.
3. Show payment page with nickname + memo + amount.
4. Detect tx, monitor confirmations, credit account after threshold.

9. Operational Notes

9.1 Network Environments

Mainnet: - P2P `45874`, RPC `45873`

Testnet: - P2P `46874`, RPC `46873`

Regtest: - P2P `47874`, RPC `47873`

9.2 Node Discovery

Recommended bootstrap stack: - DNS seed: `seeds.kubu.supply` - fixed IP seeds from chainparams for fallback.

10. Limitations and Future Work

1. Global protocol-level mandatory memo enforcement is not active; required memo is currently integration-level behavior.

2. Additional merchant/exchange SDK layers can improve reconciliation, callback reliability, and invoice lifecycle management.
3. Governance and upgrade policy should be published before public mainnet launch (BIP-style KIPs, release signaling, mandatory window policy).

11. Conclusion

KUBU combines fast scrypt PoW settlement with a native nickname identity layer that is actually validated by full nodes.
The system is designed to be usable for everyday payments while preserving miner incentives and limiting nickname squatting through economic constraints.

With nickname + memo flows, KUBU can function as both a retail payment rail and an exchange-friendly deposit network.