# CAMO: Private Digital Cash

Oct 30, 2026 · @Anthony Maniscalco



## Abstract

CAMO is private by default. Every transaction hides who sent it, who received it and how much moved, and nobody holds a special allocation of coins. CAMO is a fork of Monero v0.18, the most battle-tested private cryptocurrency. It inherits Monero's audited cryptography unchanged, launches on a fresh chain with no premine, and is mined on ordinary CPUs with RandomX.

CAMO does not invent new cryptography. New cryptography is where privacy coins break, so CAMO deliberately reuses primitives that have protected billions of dollars in Monero transactions since 2014. What CAMO adds is a clean start: a new genesis block whose reward nobody can spend, a fair launch, and a community that can evolve the chain independently.

## Motivation

On a public ledger like Bitcoin's, every payment is visible to everyone forever. That makes money surveillable, and surveillable money stops being fully fungible.

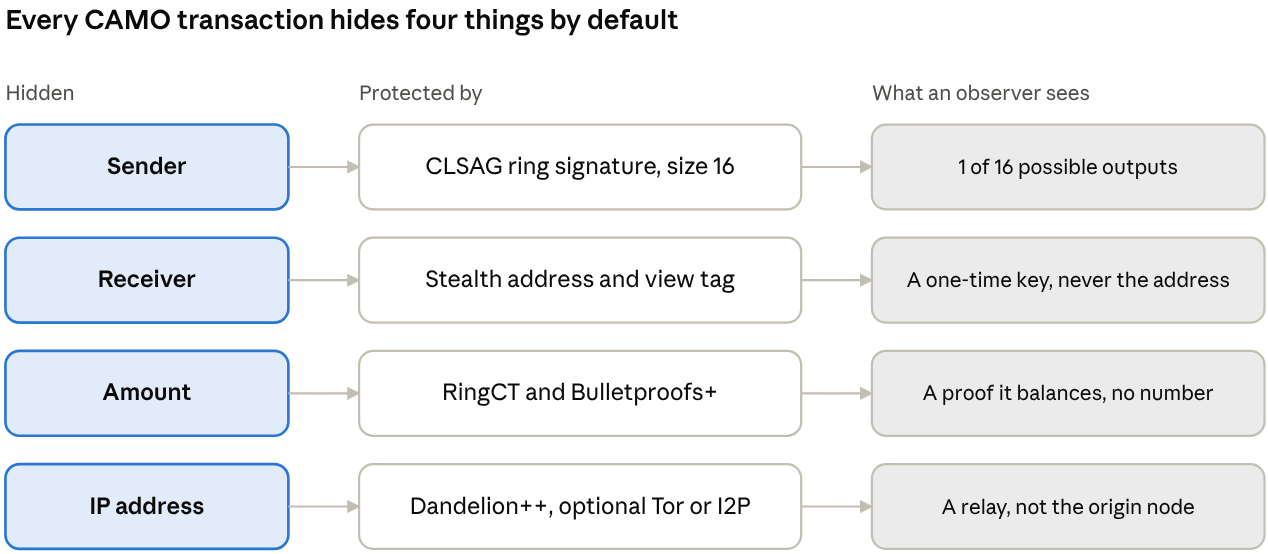
- **Surveillance.** Pay someone once and they can see your balance and your full payment history. Chain-analysis firms link addresses to identities as a business.

- **Lost fungibility.** When coins carry a visible history, exchanges and merchants can refuse "tainted" ones. One coin is then no longer equal to another, and cash is supposed to be interchangeable.
- **Safety.** A visible balance is a target. It exposes individuals to theft and extortion, and it exposes businesses' supplier and payroll data to competitors.

Cash solves all three by default: a banknote carries no history and reveals nothing about who held it before. CAMO aims to be digital cash in that sense. Privacy is mandatory for every transaction, because optional privacy makes the private transactions stand out.

## Privacy design

CAMO hides four things in every transaction: the sender, the receiver, the amount and the sender's IP address. Each is protected by a separate, peer-reviewed mechanism, and all of them are switched on from block 2 with no opt-out.




CAMO privacy layers · 4 protections, on for every transaction

In testing on the CAMO chain, a real transfer showed exactly this: both inputs hidden among 16 ring members, every amount recorded as 0, and the receiver visible only as one-time keys.

## Receiver: stealth addresses

A CAMO address (95 characters, starting with **C**) is never written to the blockchain. For each payment, the sender's wallet uses Diffie-Hellman key exchange to derive a fresh one-time public key that only the receiver can recognise and spend. Two payments to the same address share nothing an observer can link.

View tags (one byte per output) let a wallet skip about 99.6% of outputs without the full key derivation, so syncing stays fast without revealing anything.

## Sender: ring signatures (CLSAG)

Each input is signed with a CLSAG ring signature over 16 outputs: the real one plus 15 decoys drawn from the chain. An observer can verify that one of the 16 is being spent, but not which one. A key image, a value unique to each output, stops the same output being spent twice without revealing which output it came from.

## Amount: RingCT and Bulletproofs+

Amounts are replaced by Pedersen commitments, which hide the value while still letting every node check that inputs equal outputs plus fee, so no coins are created from nothing. Bulletproofs+ range proofs prove each hidden amount is positive and in range. They are about 600 bytes for two outputs, with no trusted setup.

## Network: Dandelion++ and Tor/I2P

A new transaction is first passed along a random chain of single peers (the stem) before being broadcast widely (the fluff). That makes it hard to trace to the node that created it. Nodes can also run over Tor or I2P.

## Selective disclosure: view keys

Every wallet has a private view key. Sharing it lets an auditor, accountant or tax authority see incoming payments without being able to spend. Per-transaction proofs (`get_tx_proof`) can prove a single payment without revealing anything else. The owner chooses what to disclose.

## Consensus and mining

CAMO is secured by proof of work using RandomX, which is designed so that an ordinary desktop CPU mines competitively. That keeps mining spread across many people instead of a few ASIC farms.

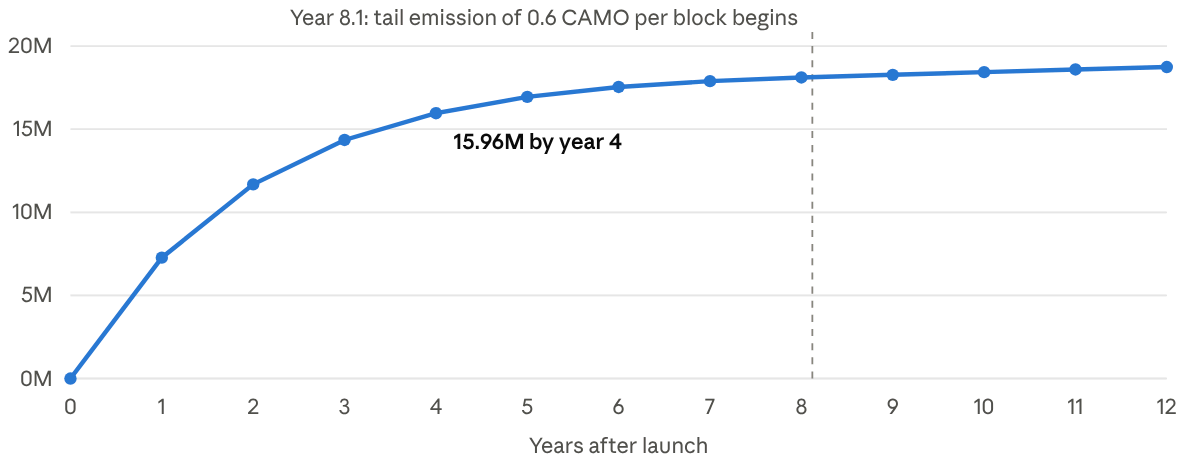
- **RandomX.** Each block hash requires executing randomly generated programs in a 2 GB dataset. This favours general-purpose CPUs and makes custom mining chips (ASICs) and GPUs uneconomical.
- **Block time.** One block every 2 minutes on average (720 blocks a day).
- **Difficulty.** Retargets every block from the last 720 blocks' timestamps, trimming outliers, so block time stays near 2 minutes as hashrate changes.
- **Dynamic block size.** There is no fixed size cap. Blocks can grow above the recent median size, but the miner then forfeits part of the reward. A long-term median limits how fast the chain can grow.
- **Dynamic fees.** The minimum fee per byte adjusts to block-size pressure. In testing on the CAMO chain, a normal 2-input, 2-output transfer paid 0.0026 CAMO.

## Monetary policy

CAMO has no premine, no developer fund and no coins for sale: every coin in circulation was mined. Emission follows Monero's curve. About 18.13 million CAMO are mined over roughly 8 years, then a permanent tail of 0.6 CAMO per block continues forever.

**Most of the main CAMO supply is mined in the first four years**

Circulating supply, millions of CAMO. 88% of the 18.13M main emission is mined by year 4.




Computed from the CAMO emission formula in the source code; 262,800 blocks a year

- **Block reward.** Each block pays (M − A) ÷ 2^19 atomic units, where M = 2^64 − 1 and A is the amount already issued. The reward starts at about 35.18 CAMO and shrinks a little with every block.
- **Tail emission.** Once the reward would fall below 0.6 CAMO (around block 2.13 million), it stays at 0.6. That adds about 157,680 CAMO a year, roughly 0.87% inflation at first and falling every year after. Miners are always paid, so the chain's security never depends on fees alone, and coins lost to forgotten keys are slowly replaced.
- **Burned genesis.** The genesis block's 17.59 CAMO reward pays a key made by hashing the phrase "CAMO genesis output key mainnet" onto the curve. Nobody knows its private key, so those coins can never move, and anyone can recompute the key to check.
- **Supply audit.** Amounts are hidden, but block rewards are public and every node checks that each transaction's inputs and outputs balance. Total supply is therefore the sum of all block rewards, which anyone can add up from the chain.

The front-loaded curve means early miners earn most of the supply: about 40% is mined in year one. A launch date announced well in advance, with binaries released beforehand, gives everyone the same chance to mine from block 1.

## Network parameters

These values are fixed in the CAMO source code. They ensure CAMO nodes and wallets can never be confused with Monero's.

| Parameter | Mainnet | Testnet | Stagenet |
|---|---|---|---|
| Ticker | CAMO | | |
| Smallest unit | 10^-12 CAMO | | |
| Address prefix (base58) | C… (prefix 69) | prefix 112 | prefix 140 |
| Subaddress / integrated prefix | 70 / 71 | 114 / 113 | 142 / 141 |
| P2P port | 19740 | 29740 | 39740 |
| RPC port | 19741 | 29741 | 39741 |
| ZMQ port | 19742 | 29742 | 39742 |

| Parameter | Mainnet | Testnet | Stagenet |
|---|---|---|---|
| Genesis nonce | 7171 | 7172 | 7173 |
| Genesis block hash | 9b1db372e5494d8a… | | |
| Ring size | 16 | 16 | 16 |
| Block time | 120 s | 120 s | 120 s |
| Binaries | camod, camo-wallet-cli, camo-wallet-rpc | | |
| Data folder | ~/.camo | | |

Each network has its own randomly generated network ID, so peers from different networks reject each other at the handshake.

## Security model and known limitations

CAMO's cryptography is as strong as Monero's because it is the same code. The biggest risk is that CAMO is a young chain with little hashrate. No privacy technology is perfect, and users deserve to know where the limits are.

| Risk | What could happen | How CAMO responds |
|---|---|---|
| 51% attack | Anyone controlling most of the hashrate could reverse recent payments (double-spend). RandomX hashpower can be rented from the much larger Monero network. | Checkpoints compiled into each release, exchanges waiting for more confirmations early on (60+ blocks), and growing the miner base before listing anywhere |
| Ring heuristics | With ring size 16, analysis of timing or decoy selection can sometimes narrow down the real input. | Default ring size of 16 with a decoy selection algorithm that mirrors real spending patterns. Upgrade path to FCMP++ (see roadmap) |
| Network metadata | Running a node on your home IP can reveal that you use CAMO. | Dandelion++ by default; Tor/I2P support; wallets can use remote nodes |

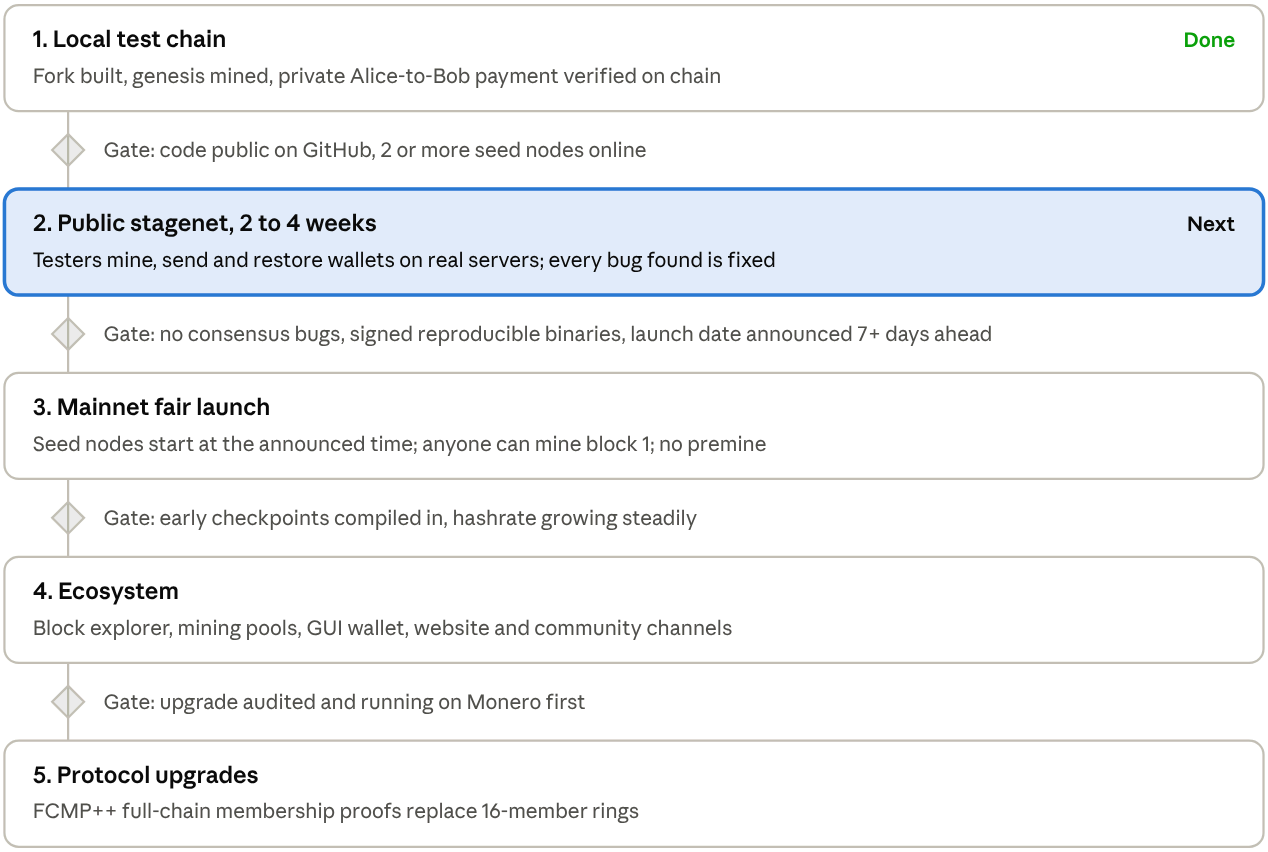
| Risk | What could happen | How CAMO responds |
|---|---|---|
| Endpoint security | Malware, a stolen seed phrase or a screenshot defeats any cryptography. | Signed, reproducible binaries; 25-word seed backup; hardware wallet support inherited from Monero |
| Small anonymity set | Early on, few transactions exist to serve as decoys, so privacy is weaker than on a busy chain. | Privacy improves as usage grows; users can wait before making sensitive payments |
| Fork code risk | A bug in Monero is also a bug in CAMO. | Track Monero security releases and merge fixes promptly |

CAMO adds no new cryptography. Every change from Monero is a parameter (names, ports, prefixes, genesis, schedule), which keeps the audit surface small.

## Launch plan and roadmap

CAMO goes to mainnet only after a public stagenet proves it in the open. Each phase starts only when its gate is met, not on a fixed date.

**CAMO reaches mainnet through a public stagenet, with a gate before each phase**




CAMO launch roadmap · 5 phases, 4 gates

The largest planned upgrade is FCMP++ (full-chain membership proofs). Instead of hiding a spend among 16 outputs, it hides it among every output on the chain, which removes ring-analysis heuristics entirely. CAMO will adopt it only after it is audited and live on Monero.

## Governance, legal notice and credits

CAMO is open-source software with no company, no foundation and no token sale behind it. Changes are proposed publicly as GitHub pull requests. A protocol upgrade takes effect only if node operators and miners choose to run the new release.

**Legal notice.** This paper describes software. It is not an offer to sell anything or investment advice, and CAMO is not promised to have any value. Privacy coins are restricted or delisted on some exchanges and in some jurisdictions. Users are responsible for following the laws where they live. Nobody behind CAMO sells coins or promises returns.

**Credits.** CAMO is built on Monero, which builds on CryptoNote. Copyright for that code stays with the Monero Project and the Cryptonote developers under the BSD 3-Clause license, which CAMO keeps. All cryptographic credit belongs to their researchers.

## References


- Monero source code, release v0.18
- CryptoNote v2.0 whitepaper, Nicolas van Saberhagen, 2013
- Zero to Monero, 2nd edition, koe, Alonso and Noether, 2020
- CLSAG: Concise Linkable Spontaneous Anonymous Group signatures, Goodell, Noether and RandomRun, 2019
- Bulletproofs+, Chung, Han, Ju, Kim and Seo, 2020
- Dandelion++, Fanti et al., 2018
- RandomX specification