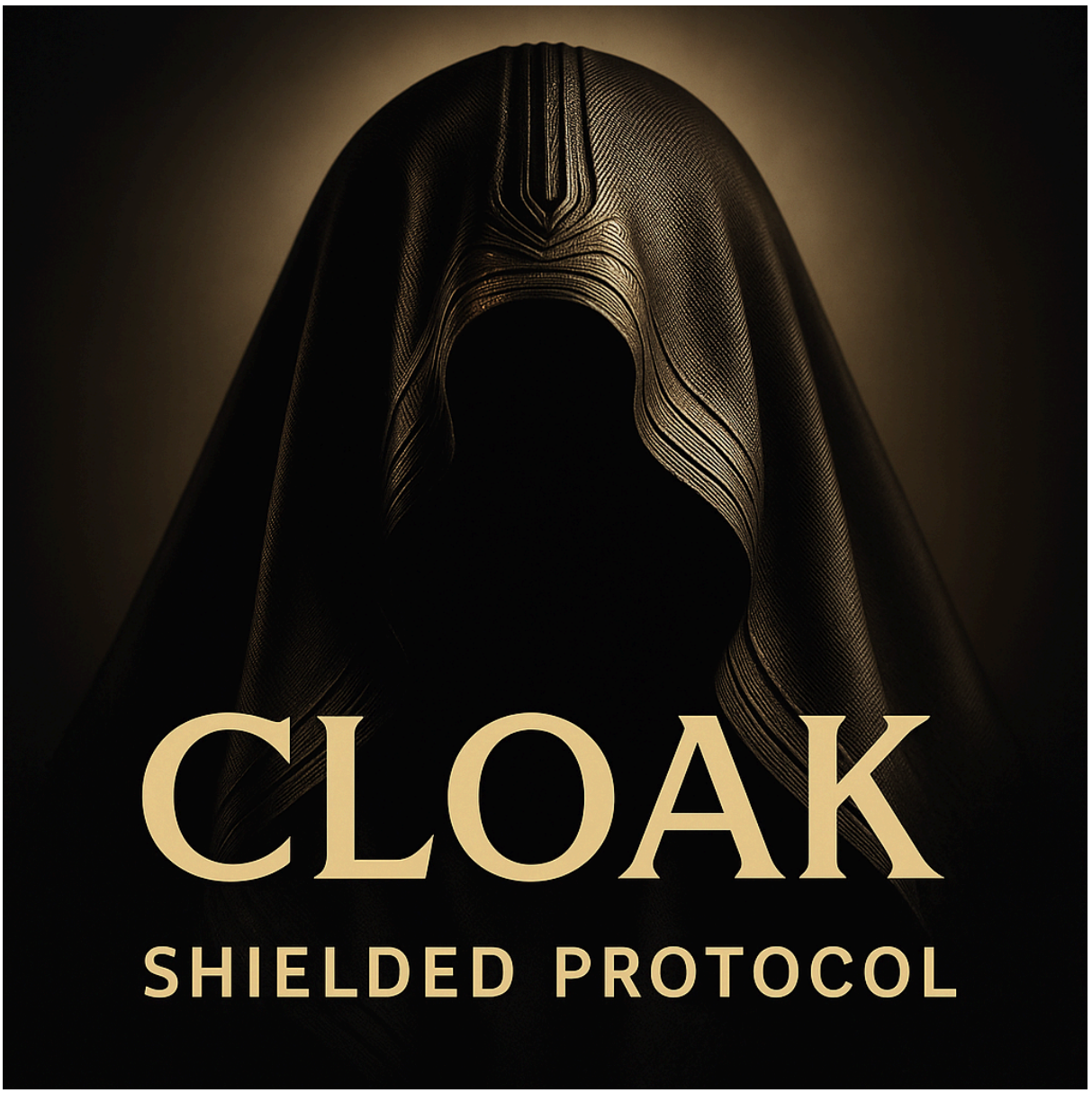
CLOAK: A Full-Fledged Privacy Protocol for Antelope-Based Blockchains

Powered by the ZEOS Caterpillar Shielded Protocol

Author: Matthias Schönebeck
Version: 1.0



1. Abstract

CLOAK is a full-fledged privacy protocol built entirely on-chain for the EOSIO/Antelope family of blockchains. It enables **Zcash-like shielded transactions** — fully private by default — while retaining full smart contract composability and supporting all token types, including NFTs.

By combining zero-knowledge proofs (Groth16) with the scalability and finality of Antelope, CLOAK introduces a **privacy-preserving financial layer** for DeFi, GameFi, and beyond — allowing users to interact with decentralized applications **without exposing transaction details**.

CLOAK is the first live instance of the **ZEOS Caterpillar Shielded Protocol**, a general-purpose, modular privacy layer that can be deployed on any Antelope chain running **leap v5.0+ with BLS intrinsics enabled**.

This whitepaper outlines the motivation, architecture, cryptographic design, and token model of the CLOAK ecosystem, as well as its roadmap toward becoming the universal privacy standard for Antelope.

2. Introduction and Motivation

Blockchain transparency is both a strength and a limitation. While immutability and auditability are crucial, they come at the cost of privacy — every balance, transfer, and contract call is public.

This lack of confidentiality deters serious institutional and enterprise adoption of DeFi, and exposes individuals to tracking and exploitation.

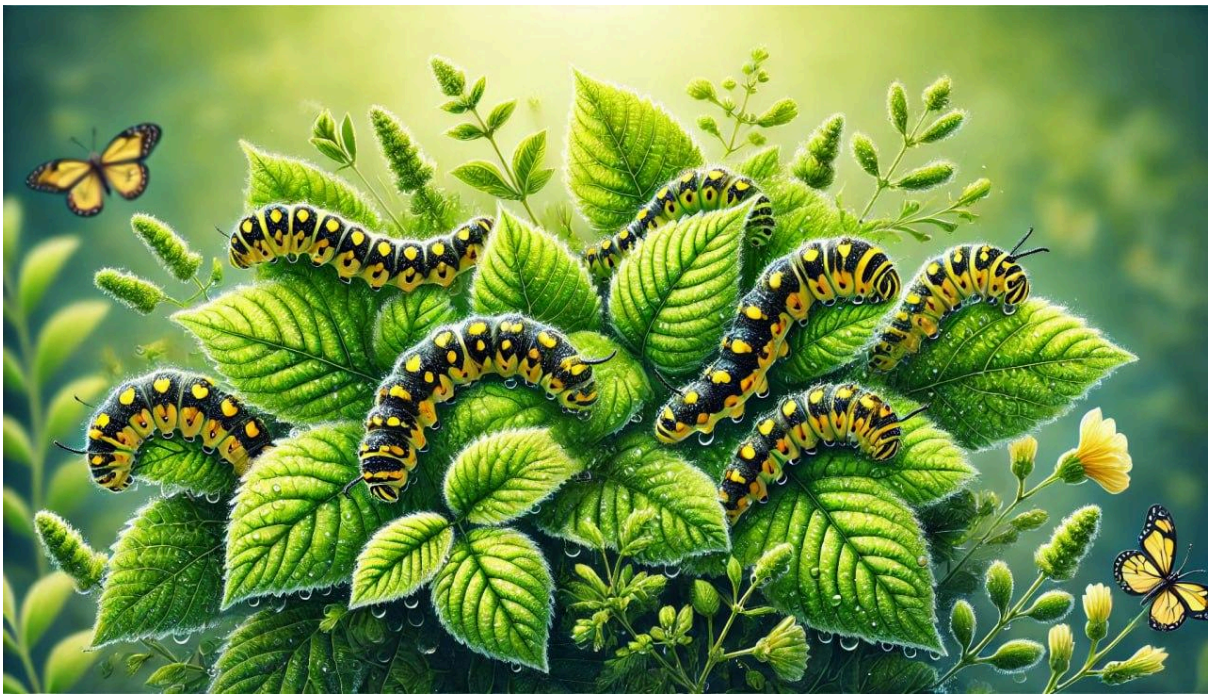
CLOAK solves this by bringing **complete transactional privacy** to general-purpose smart contract blockchains — not through an isolated privacy coin, but as a **programmable privacy layer** integrated within Antelope's architecture.

Unlike Monero or Zcash, CLOAK supports all kinds of tokens and can be used by third-party smart contracts out-of-the-box, enabling **private DeFi**, **anonymous governance**, **confidential vaults**, and **shielded game economies (GameFi)**.

Feature	Zcash	Monero	Tornado Cash	CLOAK
Privacy	✅	✅	✅	✅
Smart contracts	❌	❌	⚠️	✅

Multi-asset support	❌	❌	⚠️	✅
Cross-chain capability	❌	❌	❌	✅

3. The ZEOS Caterpillar Shielded Protocol



The **ZEOS Caterpillar Shielded Protocol** is the cryptographic and architectural foundation on which **CLOAK** is built. It represents a full-fledged, on-chain implementation of Zcash-style privacy within a smart-contract-based blockchain environment — specifically, **Antelope (formerly EOSIO)** blockchains. The protocol brings together zero-knowledge proof technology, efficient cryptographic primitives, and Antelope's high-performance architecture to enable scalable, composable privacy across all token types.

Unlike most privacy solutions that exist as separate blockchains or external mixers, the ZEOS Caterpillar Protocol operates **entirely within a single smart contract**. This design ensures that private and public assets coexist in the same blockchain ecosystem and can interact seamlessly through standardized interfaces. As a result, developers can integrate privacy directly into their decentralized applications (dApps) without requiring sidechains, custodial bridges, or external privacy layers.

3.1 Overview

The ZEOS Caterpillar Protocol was designed from the ground up for **Antelope-based blockchains**, leveraging their deterministic execution, fast block finality, and parallelized transaction processing. Privacy on Antelope introduces unique challenges: smart contracts

run in **WASM (WebAssembly)** environments that are single-threaded and constrained by strict CPU and RAM limits. To make zero-knowledge cryptography feasible in such an environment, the protocol uses **Groth16 zk-SNARK proofs** — a proof system known for its compact size and extremely fast verification times.

Each privacy-preserving transaction (or "shielded transaction") in the protocol contains a Groth16 proof that can be verified on-chain in a few milliseconds, consuming minimal resources while preserving strong cryptographic guarantees. This allows the protocol to function efficiently inside a single smart contract, where CPU cycles are scarce and transaction costs are closely tied to computation time.

The protocol is **asset-agnostic**. It supports all token standards available on the underlying Antelope chain — including fungible tokens such as `eosio.token` or any custom token contract, as well as non-fungible tokens (NFTs). Each asset type can be moved into the shielded pool, transacted privately, and later withdrawn back to the transparent layer without losing its identity or metadata. In essence, the ZEOS Caterpillar Shielded Protocol acts as a **universal privacy layer** that sits above the blockchain's native asset system while remaining fully interoperable with it.

By encapsulating all privacy operations within a single smart contract, the protocol achieves **composability and scalability**. Other smart contracts can interface with the shielded ledger through standardized actions, allowing developers to add private functionality to any dApp with minimal integration overhead. This modular architecture also makes it possible to deploy the protocol on multiple Antelope-based blockchains in parallel — an essential requirement for CLOAK's long-term cross-chain vision.

3.2 Protocol Components

At its core, the ZEOS Caterpillar Shielded Protocol consists of four interdependent components: the **shielded ledger**, **note commitments**, **nullifiers**, and **proof verification system**. Together, these mechanisms ensure that private transfers remain confidential, verifiable, and free from double-spending.

The Shielded Ledger serves as the protocol's internal accounting system. Instead of maintaining visible account balances like a traditional token contract, it tracks encrypted ownership records called *notes*. Each note represents a certain quantity of a specific asset owned by a particular holder — but the note itself is encrypted, hiding all sensitive information from the public blockchain. The shielded ledger organizes these notes within a **Merkle tree**, which allows efficient verification of transaction validity without revealing any underlying data.

Note Commitments are cryptographic hashes that commit to the contents of a note (owner, amount, asset type, and other parameters) without disclosing them. When a new note is created, its commitment is added to the Merkle tree, effectively registering it within the shielded ledger. Because commitments are one-way hashes, no one can derive the actual note contents from them, preserving privacy while maintaining verifiability.

Nullifiers act as cryptographic "spent markers." When a note is used as an input to a transaction, its corresponding nullifier is published on-chain to prevent the same note from

being spent twice. Each nullifier is unique to a note and its secret key, ensuring that only the rightful owner can produce it. The protocol enforces that no transaction can consume a note whose nullifier already exists in the ledger, thereby guaranteeing integrity and preventing double-spending.

Finally, **Proof Verification** ties everything together. Each shielded transaction contains a zero-knowledge proof demonstrating that:

1. The spender knows the secret keys associated with the input notes.
2. The input notes exist within the ledger's Merkle tree.
3. The sum of input and output amounts balances correctly.
4. No nullifier has been used before.

All of these conditions are verified without revealing any actual data about the transaction. The Groth16 proof system ensures that the verification step can be executed entirely on-chain using Antelope's built-in **BLS12-381 elliptic curve intrinsics**, making validation both fast and cost-efficient.

To achieve balance checking between inputs and outputs **without revealing any of the actual note values**, the protocol employs **homomorphic Pedersen Commitments**. Each note amount is encoded as a Pedersen Commitment — a cryptographic construct that hides the true value while allowing arithmetic operations to be performed in the encrypted domain. Because Pedersen Commitments are additively homomorphic, the verifier can confirm that the sum of all input commitments equals the sum of all output commitments (plus or minus the transaction fee) **without ever learning the underlying amounts**. This enables transactions to consume multiple existing notes and create multiple new notes atomically, ensuring that all value transfers remain balanced, confidential, and unlinkable. In practice, this mechanism allows CLOAK to represent arbitrarily complex asset transfers — even those involving multiple tokens and participants (shielded or unshielded) — while maintaining perfect privacy of amounts and ownership.

3.3 Integration Layer

To make the protocol usable by real applications, the ZEOS Caterpillar Shielded Protocol defines a robust **integration layer** that connects the private and public domains of the blockchain. Users and dApps can move assets from public EOSIO accounts into the shielded pool through well-defined *entry actions*, and later withdraw them back to public accounts via *exit actions*. Each of these transitions is accompanied by a zero-knowledge proof that enforces correctness and prevents misuse.

This mechanism makes it possible to create **hybrid workflows**, where some interactions remain public while others are executed privately, all within the same blockchain environment. For instance, a decentralized exchange can accept **private deposits** from users' shielded wallets, execute the **actual trade transparently** on the public layer for verifiable price discovery, and then **return the proceeds back into the privacy layer**, all within a single composable transaction flow. This hybrid model preserves full auditability where needed, while maintaining end-to-end privacy for user balances and positions.

CLOAK extends this integration layer with additional abstractions: **vaults** and **authentication tokens (auth tokens)**. Vaults act as persistent privacy wallets within the protocol, while auth tokens serve as anonymous capability tokens that represent private user identities in dApps. Together, they enable seamless, privacy-preserving smart contract interactions — allowing developers to build decentralized applications that respect user confidentiality by default.

4. Architecture of CLOAK

CLOAK is a full implementation of the **ZEOS Caterpillar Shielded Protocol**—a self-contained privacy layer that brings Zcash-like zero-knowledge functionality directly into the Antelope smart-contract environment. Whereas the Caterpillar protocol defines the theoretical and cryptographic framework, CLOAK realizes it as a working on-chain system composed of interoperable smart contracts, an off-chain cryptographic library, and multiple front-end applications. Together, these components form a complete privacy infrastructure for Antelope-based blockchains, enabling seamless movement between transparent and shielded assets, private dApp interaction, and eventually cross-chain privacy.

4.1 System Overview

At a high level, CLOAK consists of four layers that interact through standardized interfaces:

1. **On-Chain Core (Smart Contracts)** – Implements the shielded ledger, note commitment trees, proof verification, and all user-facing actions for deposit, withdrawal, and private transfers.

2. **Cryptographic Engine (`zeos-caterpillar` Library)** – A Rust-based library that performs the heavy cryptographic operations required for proof generation and verification, compiled both natively (for the desktop wallet) and to WebAssembly (for browser-based apps).

3. **Application Layer (Vaults, Auction, DEX, Bridge, etc.)** – Smart contracts and web components built on top of the shielded core that provide user-level functionality such as token distribution, private staking, vault management, and DeFi interactions.

4. **User Interfaces (CLOAK Wallet & Web Apps)** – Cross-platform Qt wallet and companion web applications providing graphical access to all on-chain features, including shielded transfers, vault creation, and auction participation.

These layers together provide a coherent privacy ecosystem, where each component is replaceable yet interoperable through strict on-chain interfaces and cryptographic standards defined by the underlying protocol.

4.2 On-Chain Core

The on-chain core of CLOAK is implemented as a single, large-scale smart contract that embodies the entire ZEOS Caterpillar Shielded Protocol. This contract maintains the

shielded ledger, verifies Groth16 proofs on-chain using **BLS12-381 elliptic-curve intrinsics**, and enforces the correct creation and consumption of commitments and nullifiers. Every shielded transaction is represented as an EOSIO action whose payload contains the zero-knowledge proof, encrypted notes, and auxiliary data necessary for Merkle-tree updates.

A key design principle is **self-containment**: all private state, including the Merkle tree, note commitments, and nullifiers, resides within the contract's tables, ensuring that the privacy layer remains completely deterministic and verifiable by any node. No off-chain state or external storage is required. This makes CLOAK both **trustless** and **composable**, as other smart contracts can call its actions directly without relying on third-party servers or relayers.

All shielded actions are handled by specialized entry points that correspond to high-level operations such as:

- `mint` – deposit from public to private;
- `spend` – transfer within the shielded domain;
- `burn` – withdraw from private back to public.

Each action is backed by a corresponding Groth16 circuit in the cryptographic layer, guaranteeing that all invariants—ownership, balance, and double-spend prevention—are mathematically enforced on-chain.

4.3 Cryptographic Engine (`zeos-caterpillar` Library)

The heavy cryptographic operations of CLOAK—such as zero-knowledge proof generation, Pedersen commitment computation, and Merkle-path hashing—are performed by the **`zeos-caterpillar`** Rust library. This library implements all circuits and arithmetic logic used by the on-chain contract and can be compiled to both native and WebAssembly targets.

For the **desktop wallet**, it runs natively, taking full advantage of multithreading and modern CPU instructions to generate proofs quickly.
For **web applications**, it is compiled to **WebAssembly (wasm)**, allowing users to generate proofs directly in their browsers without sharing private keys or note secrets with any external service. A multi-threaded wasm build is available as well and leverages `SharedArrayBuffer` and web-worker parallelism when the browser operates in a cross-origin-isolated context.

This design ensures that proof generation remains **client-side, non-custodial, and trustless**, while still providing acceptable performance on consumer hardware.

4.4 Application Layer

Above the shielded core, CLOAK introduces a set of specialized on-chain applications that utilize its privacy primitives to deliver real-world functionality. The most notable of these include:

- **Token Auction** – A bootstrapping mechanism for the CLOAK token itself. Users can contribute the **native system token of the underlying EOSIO/Antelope blockchain** in periodic rounds. A fixed percentage (10 %) of every contribution is automatically staked to allocate CPU and NET resources required to operate the shielded protocol. After each round, participants can claim newly issued CLOAK tokens. This process ensures that the protocol provisions its own on-chain resources organically as adoption grows.

- **Vault System** – Implements persistent shielded wallets identified by *auth tokens*, enabling private balances and long-term storage of assets inside the privacy layer. Vaults act as the private counterpart to public EOSIO accounts, providing a flexible foundation for private deposits, withdrawals, and smart-contract interactions.

- **DEX and Bridge Modules** – Extend the core by enabling private DeFi operations and cross-chain transfers between different Antelope networks. These components interact with the shielded core through defined entry and exit actions, ensuring that all private value transfers remain consistent with the zero-knowledge framework.

- **Independent Third-Party dApps** – Any decentralized application deployed on the same EOSIO/Antelope chain can integrate directly with CLOAK through its well-defined on-chain interface. By linking their contract logic to the shielded protocol, developers can make existing or new dApps privacy-aware—allowing users to interact privately without modifying the broader blockchain infrastructure.

All modules share the same cryptographic foundation, meaning that a vault deposit, a DEX trade, a bridge transfer, or a third-party dApp integration are all ultimately **shielded transactions verified by the same Groth16 proof verifier**. This unified architecture allows the privacy layer to scale horizontally across applications while maintaining a single, auditable cryptographic base.

4.5 User Interfaces

To make this infrastructure accessible to end users, CLOAK provides a suite of user-facing applications built around two main entry points:

- **CLOAK Wallet (QT Desktop App)** – A full node-capable, cross-platform wallet written in C++/Qt that embeds the Rust cryptographic engine via FFI. It supports Windows 10/11, Ubuntu 22 & 24, and macOS (Big Sur and later, x86 build for now). The wallet handles key management, shielded transactions, vault interactions, and direct connection to web apps through a secure local WebSocket bridge.

- **CLOAK Web Apps** – A bundle of browser applications (Dashboard, Auction, Vaults, DEX, and Bridge) that connect to either an Antelope/EOSIO account or a CLOAK privacy wallet. These apps offer a seamless interface for sending tokens between public accounts, private addresses, and vault auth token hashes, effectively demonstrating the interoperability between transparent and shielded domains.

Through this architecture, CLOAK achieves a unified privacy experience: from core cryptography to wallet integration, everything operates within a consistent, permissionless, and developer-friendly framework.

5. Token Economy and Bootstrapping Model

The **CLOAK token** serves as the **native gas and utility token** of the shielded protocol. Every private transaction executed through the protocol—on any supported EOSIO/Antelope-based blockchain—requires a small fee denominated in CLOAK. These fees are used to sustain the protocol's on-chain execution costs and to prevent spam or denial-of-service attacks within the privacy layer.

5.1 Deflationary Token Model

CLOAK is inherently **deflationary by design**. A fixed percentage of each transaction fee is permanently **burned** at the protocol level. This applies to all shielded actions—private transfers, deposits, withdrawals, dApp interactions, and any operation invoking zero-knowledge proof verification. Because these fees are algorithmically deducted and destroyed as part of the transaction flow, CLOAK's circulating supply continuously decreases over time, directly tying network usage to deflationary pressure.

Although the **maximum supply** of CLOAK is capped at **1 billion tokens**, it will **never reach this limit** in practice. From the very moment the protocol goes live, token burns begin to occur naturally as users interact privately. In other words, CLOAK's deflationary property is not governed by discretionary decisions or governance votes—it is an immutable economic law embedded in the protocol itself. This ensures long-term scarcity and value alignment between token holders and protocol activity.

5.2 Resource Coupling and Protocol Sustainability

CLOAK's launch mechanism is designed to bootstrap both **token distribution** and **resource allocation** simultaneously.
During the token auction phases, users contribute the **native system token** of the underlying Antelope blockchain (e.g., EOS, UOS, or TLOS). A portion of each contribution (currently 10%) is **automatically staked** to allocate CPU and NET resources to the protocol contract. This ensures that the shielded system remains self-sufficient, with its computational and bandwidth needs covered as adoption grows.

Once a round concludes, participants can claim their CLOAK tokens proportionally to their contribution. These distributions not only establish the initial token supply but also directly link resource provisioning to token ownership—creating a **sustainable feedback loop** between user participation, protocol capacity, and economic growth.

5.3 Cross-Chain Utility

Because CLOAK is the **gas token** for all shielded operations, it retains its utility across every Antelope-based blockchain that integrates the protocol. Through the planned **CLOAK**

Bridge, users will be able to move their shielded assets privately between networks while maintaining a unified CLOAK balance across chains.
Every transaction—no matter which chain it occurs on—contributes to the same global burn logic, ensuring that **all protocol usage everywhere contributes to CLOAK's deflationary effect**. This cross-chain cohesion allows the token economy to scale with adoption while maintaining consistent monetary policy across all participating blockchains.

5.4 Long-Term Vision

While CLOAK currently focuses on its role as a **gas and utility token**, future iterations of the ecosystem may expand its function to include **staking, incentive, and DAO-based governance mechanisms**. However, at launch, the design remains deliberately minimalistic: CLOAK powers transactions, enforces scarcity through automated burns, and fuels the ongoing operation of the privacy layer itself.

In essence, the CLOAK token is **both the fuel and the heartbeat of the protocol**. Every private transaction consumes it; every network interaction reinforces its scarcity.
This creates an elegantly simple and self-regulating economy—one where usage itself ensures sustainability, and privacy intrinsically generates value.

6. Token Distribution and Auction Mechanics

The distribution of the CLOAK token is designed to be **transparent, fair, and self-sustaining**, without reliance on pre-mines, venture allocations, or centralized control. Instead of issuing tokens arbitrarily, CLOAK employs an **on-chain token auction mechanism** that mirrors the original EOS token sale—adapted for the next generation of Antelope-based networks and extended with privacy support.

6.1 Overview of the Auction Process

The token auction is managed by the protocol's smart contract itself and is executed entirely on-chain, without intermediaries. The auction operates in a series of **rounds**, each lasting a fixed duration (e.g., 12 or 24 hours).
During each round, participants can contribute the **native system token** of the host blockchain (such as EOS, UOS, or TLOS). At the end of every round, a predetermined number of CLOAK tokens is allocated proportionally among all contributors of that round.

This creates a dynamic, market-driven distribution process: participants effectively determine the implicit token price by how much they collectively contribute during a given period. Each round thus represents a snapshot of network demand, providing both fair access and organic price discovery.

6.2 Contribution and Staking Mechanism

Every contribution to the auction serves two purposes:

1. **Token Acquisition** – The majority of the contributed system tokens are recorded as part of the user's participation and determine their claimable CLOAK amount for that

round.

2. **Protocol Resource Allocation** – A fixed percentage (currently **10%**) of each contribution is automatically **staked** by the contract. This staking provides the CPU and NET bandwidth required for the operation of the CLOAK shielded protocol on the blockchain itself.

Through this mechanism, CLOAK **self-provisions** the computational resources it needs to scale. The more people participate, the more resources the protocol accumulates—creating a direct correlation between adoption and throughput capacity.

This architecture ensures that the privacy layer remains sustainable, decentralized, and permanently operational without depending on external funding or privileged accounts.

6.3 Claiming and Token Issuance

After each auction round concludes, participants can **claim** their CLOAK tokens through one of two methods:

- **Public Claim** – Performed using a standard EOSIO account.

- **Private Claim** – Executed through the **CLOAK Shielded Protocol** itself, using an **auth token** for authentication and privacy preservation.

This dual-mode claiming system ensures seamless interoperability between transparent and shielded domains. Users who wish to remain anonymous can participate and claim entirely within the privacy layer, while others can do so transparently.

Token issuance itself is **deferred until claim time**—meaning CLOAK tokens are only minted when claimed, not pre-generated. This design ensures that **unclaimed allocations never exist on-chain**, reducing potential supply overhead and eliminating unused inflation.

6.4 Fairness and Anti-Whale Properties

Because each round distributes a fixed amount of CLOAK regardless of the total contributed volume, every participant within the same round competes on equal terms. Large contributors do not inherently gain an advantage, as their proportional share only depends on their contribution relative to others during that same period.
This structure naturally smooths token distribution over time, preventing early concentration of supply and ensuring that community participation remains meaningful even in later rounds.

Moreover, the auction contract enforces **uniform contribution conditions** and fully deterministic logic. All rules—such as minimum contribution size, staking ratio, and round timing—are stored and publicly verifiable on-chain.

6.5 Summary

In summary, the auction serves as both a **launchpad and bootstrap mechanism** for the CLOAK protocol:

- It distributes the token in a transparent, fair, and decentralized manner.
- It allocates blockchain resources automatically through the staking of native system tokens.
- It enables both **public** and **private** participation.
- It kickstarts the **deflationary cycle** by making CLOAK immediately usable—and burnable—as the gas token for private transactions.

Through this model, CLOAK's initial distribution and infrastructure provisioning are elegantly unified: **each contribution funds the very resources that make privacy possible**, ensuring that from the very first auction round, the system sustains itself.

7. Vaults, Auth Tokens, and dApp Integration

While the CLOAK Shielded Protocol provides the cryptographic foundation for privacy, **Vaults** and **Auth Tokens** define how users and decentralized applications (dApps) actually interact with the system. Together, they form the **integration layer** that bridges private and transparent domains and enables fully private DeFi, GameFi, and other smart-contract use cases.

7.1 Vaults — Private Accounts for the Shielded Layer

A **Vault** is the shielded equivalent of a blockchain account.
Each vault represents a private wallet within the CLOAK privacy layer, capable of holding any supported asset — whether a fungible token, NFT, or other on-chain representation. Unlike public EOSIO accounts, vaults are **not identified by names**; instead, they are bound to **auth token hashes** that act as one-way pseudonymous identifiers.

Vaults can:

- Receive deposits directly from other shielded wallets.
- Accept public or transparent transfers from EOSIO accounts via the `thezeosvault` bridge contract.
- Send private withdrawals back to public accounts or to other vaults.
- Interact with dApps privately through authentication and scoped permissions.

Functionally, vaults behave like privacy wallets but are implemented as **on-chain tables within the shielded ledger**. Each vault's contents are represented as encrypted UTXOs that can only be accessed by their owner's private viewing and spending keys. This design maintains the same security and verifiability guarantees as Zcash, while remaining fully composable with smart contracts on the same blockchain.

7.2 Auth Tokens — One-Way Identifiers for Private Interaction

Auth Tokens are cryptographic authentication primitives that serve as the interface between the shielded protocol and dApps.

Each auth token is derived from a **seed phrase**, in the same way wallets are created themselves. The hash of this token becomes a **unique identifier** that can be used by other smart contracts as a reference to a vault — without ever revealing the vault's "private key" or link to its owner.

There are two primary types of auth tokens:

1. **User-Generated Tokens:**
   Users can manually generate auth tokens inside the CLOAK wallet application. This is especially useful when preparing to interact with a dApp privately. After creating an auth token, the user can transfer tokens to its hash value (used as a recipient), and a corresponding vault will automatically appear on the **Vaults** page.

2. **dApp-Generated Tokens:**
   When interacting with privacy-enabled dApps (e.g., the CLOAK DEX or Auction), new auth tokens are automatically created as part of the transaction flow. These tokens act as temporary identities representing private positions, orders, or balances held within the dApp's logic.

Auth tokens are **one-way constructs** — they can be verified, but not reversed. This guarantees **privacy and unlinkability**, even when multiple tokens originate from the same privacy wallet. In essence, they provide a scalable mechanism for **"ephemeral privacy identities"** that can represent user state across DeFi, gaming, or DAO environments without ever exposing wallet information.

Importantly, when a dApp integrates with CLOAK, this entire **auth token lifecycle is handled transparently in the background**. From the user's perspective, interacting with a privacy-enabled application feels identical to using a standard EOSIO/Antelope account: the wallet authorizes actions, assets move, and balances update — yet under the hood, **auth tokens are silently created, linked, and managed** to preserve privacy. This design ensures that adopting privacy does not come at the cost of usability; developers can add shielded functionality without altering the familiar user experience.

7.3 Integration Model for dApps

The CLOAK protocol is intentionally designed to make **third-party integration straightforward**. Any dApp deployed on the same EOSIO/Antelope chain can add privacy support by integrating with CLOAK's public action interface.

When a dApp on the same EOSIO/Antelope chain needs to handle user funds privately, **the assets move out of the Shielded Protocol and into the dApp's transparent contract account**, while privacy is preserved through **auth tokens** used for internal bookkeeping:

- **Private Deposit → Transparent Execution.**
  The user initiates a **private deposit** from their CLOAK wallet to the dApp. Under the hood, the Shielded Protocol **nullifies the user's shielded note** and transfers the corresponding asset **to the dApp's transparent EOSIO account**, tagged with the user's **auth token** (or a derivative). From this point, the dApp can execute its logic (trades, orders, positions, game state) **transparently and audibly**, while internally

associating state with the **auth token** rather than a public EOSIO/Antelope account.

- **Private Withdrawal (Synchronous).**
  When the user withdraws **privately** from the dApp, the dApp returns the asset to the Shielded Protocol, where a **new UTXO (note commitment) is minted** to the user's privacy wallet. To the user, balances "reappear" privately because on-chain, the dApp's transfer back is followed by a shielded **mint** tied to the withdrawn asset.

- **Private Withdrawal (Asynchronous via Vault).**
  If the dApp returns assets **asynchronously** (e.g., settlement, order fill) and includes the **auth token in the memo**, the transfer is automatically routed to the **Vault** contract. There, funds are stored **under the corresponding auth token hash**. The user can later **mint** these assets back into their privacy wallet at any time, without ever exposing a public address.

This model preserves a **hybrid workflow**: **transparent, fully auditable execution** inside the dApp where it makes sense (price discovery, matching, settlement), combined with **private entry and exit** anchored by the Shielded Protocol. Throughout, **auth tokens** provide the linkage needed for dApp state management—**without revealing the user's identity**.

7.4 Use Cases and Extensibility

The vault and auth token model opens up a broad range of possibilities beyond traditional private transfers:

- **Private DeFi:** Users can participate in liquidity pools, auctions, or lending markets while keeping their balances, positions, and counterparties private.
- **GameFi and NFT Privacy:** Auth tokens can represent in-game assets or NFT identities that remain hidden until revealed by the player.
- **Private DAO Participation:** Members can vote or contribute funds privately, maintaining both accountability and anonymity.
- **Cross-Chain Privacy:** When connected to the CLOAK Bridge, vaults will serve as unified privacy accounts spanning multiple Antelope networks.

This modular architecture ensures that privacy is not a closed system but rather a **permissionless extension layer** available to every smart contract on the network.

7.5 Summary

Vaults and auth tokens transform the abstract cryptography of the shielded protocol into a **developer-friendly and user-accessible privacy layer**. They provide the building blocks for seamless integration between private and public smart-contract logic, enabling dApps to adopt privacy as easily as they would add a new API call.

In practice, this means that **any dApp can go private**—without requiring special infrastructure or custom cryptographic code. CLOAK handles the complexity, while developers and users benefit from privacy that is composable, scalable, and truly native to the Antelope ecosystem.

8. Cross-Chain Architecture and the CLOAK Bridge

The long-term vision of CLOAK extends beyond a single blockchain. Privacy must not end at chain boundaries — it must travel with the user. To achieve this, CLOAK introduces a **cross-chain architecture** built upon existing trustless interoperability standards within the Antelope ecosystem.

8.1 Foundation: Trustless IBC for Antelope

The Antelope (formerly EOSIO) ecosystem already possesses a mature, **trustless inter-blockchain communication (IBC)** protocol, originally developed by **UX Network**.
This implementation — documented at ibc-docs.uxnetwork.io — enables **light-client-based cross-chain transfers** between compatible Antelope blockchains, without the need for custodial bridges or third-party relayers. Each participating chain verifies the other's consensus proofs directly on-chain, achieving **native interoperability** with cryptographic finality.

Although the UX Network itself is no longer active, its IBC technology remains open-source and fully compatible with all modern **Leap-based Antelope chains (v5.0 and above)**. CLOAK is built to leverage this infrastructure — effectively re-purposing it to enable **trustless, private IBC** between privacy-enabled blockchains.

8.2 CLOAK Bridge Architecture

The **CLOAK Bridge** builds on top of Antelope IBC by embedding privacy at the transport layer.
When a user transfers a shielded asset across chains, the transaction flow is as follows:

1. **Burn on Source Chain** — The user's private UTXO (representing an asset on chain A) is nullified within the CLOAK protocol. A zero-knowledge proof attests to the burn event without revealing the amount or the sender's vault identity.

2. **IBC Relay and Verification** — The burn proof and event data are transmitted through the Antelope IBC channel. The target chain verifies the Merkle-proofed IBC packet natively, just as it would for a transparent token transfer.

3. **Mint on Destination Chain** — A new shielded UTXO is created on chain B, representing the same asset in the destination privacy layer. From the user's perspective, the transfer is seamless — the asset simply "appears" on the new chain, still under the same private key.

This system introduces the world's first **trustless private IBC**, allowing fully private value transfers across independent blockchains without ever exposing the transaction contents to intermediaries or relayers.

8.3 Unified Multi-Chain Wallets

Because the CLOAK Shielded Protocol uses deterministic key derivation across all supported networks, a user's **single private key** controls their privacy wallets on every chain simultaneously.
Internally, the wallet tracks multiple ledgers — one per chain — each synchronized via IBC proofs. Tokens may be distributed across networks, but they remain accessible through one unified **CLOAK wallet interface**.

This design ensures that:

- Privacy is **portable** — users can move their assets without re-establishing new shielded identities.

- The **CLOAK token** retains cross-chain gas functionality. Every chain enforces the same burn logic, so all shielded transactions everywhere contribute to global deflation.

- Each network can **scale independently**, while remaining part of a shared privacy ecosystem.

8.4 Summary

By leveraging the existing Antelope IBC standard and extending it with zero-knowledge proofs, CLOAK establishes the foundation for a **trustless, cross-chain privacy layer**. The CLOAK Bridge ensures that privacy is not confined to a single blockchain but becomes a **portable, verifiable, and composable primitive** across the entire Antelope ecosystem.

9. Implementation and Repositories

- **Core Protocol:** github.com/mschoenebeck/zeos-caterpillar
- **MPC Setup:** github.com/mschoenebeck/zeos-caterpillar-mpc
- **CLOAK Contracts:** (private repo, soon to be open-sourced)
- **CLOAK App Bundle:** app.cloak.today

Code Stack

- C++ / EOSIO Smart Contracts
- Rust (ZK core)
- WebAssembly (Browser proof generation)
- Qt (Desktop wallet)
- React (Web apps)

10. Team and Acknowledgements

CLOAK is developed and maintained by **Matthias Schönebeck**, the creator of the ZEOS Caterpillar Shielded Protocol. The project builds upon pioneering open-source work from **Electric Coin Company (Zcash)** and the broader **zk-SNARK community**.

Special thanks to the Antelope engineering community for continued innovation in blockchain scalability and performance.

11. Disclaimer

This document is for informational purposes only and does not constitute financial advice or a solicitation for investment. CLOAK is open-source software provided “as-is.” Use at your own risk.
Privacy does not imply illegality — users are expected to comply with their jurisdiction’s laws.