GALACTICA

NETWORK

zkKYC Tech Specification

Version 1.0

Jun 27, 2024

Table of contents

What is zkKYC

Core Concept

Zero-Knowledge Know Your Customer (zkKYC) is an advanced KYC mechanism that leverages zero-knowledge proofs to verify user identities without exposing personal data. Allows for privacy-preserving on-chain verification of user identities without exposing personal information. This technology aims to balance the regulatory requirements for Anti-Money Laundering (AML) and Counter-Terrorist Financing (CTF) with the privacy and security needs of users in decentralized environments.

How does it work?

1. **KYC Guardians** (centralized parties) ranked by reputation issue KYC Records for users. Guardian may ask users to pass KYC to issue one or use existing KYC data (SwissBorg Example). For each KYC, the guardian issues 2 entities:
   - A hash of the user's KYC is issued as a **Merkle leaf** in a **Merkle tree** structure on the blockchain.
   - KYC data itself is being sent directly to the user's machine in an encrypted manner, ensuring maximum privacy and security. It is then stored in a non-custodial wallet.
2. **Users can prove statements on their KYC (ZK Proofs)** to specific entities or protocols when asked. Example statements are:
   - Is the user KYC verified?
   - Is the user a resident of the US.
   - Does the user meet the specific criteria for participating in a token sale?
   - Is the user eligible for accessing age-restricted content based on jurisdiction?

Key Features

1. **Existing KYC brokers as KYC Providers:** Galactica leverages existing KYC data by allowing users to import their verified KYCs from other entities through seamless integrations. For example, users can generate zkKYC on Galactica using their Swissborg account.
2. **Off-chain proving:** enables any service, including non-crypto-related ones, to request proof of certain conditions, such as age verification or eligibility, without accessing or revealing the user's underlying data.
3. **User address obfuscation:** the Guardian issuing the KYC Record doesn't know the address that will be using the record. Even if there is a data leak, the KYC data will not be associated with on-chain activity.
4. **Fraud investigation process:** authorized entities decrypt user’s data through a consensus mechanism if KYC Guardians verify the legitimacy of suspicious activity.

Purpose and Vision

1. **Decentralized Identity:** the KYC is passed once and then can be used with various parties. All the KYC issuers (Guardians) are equal and are ranked by reputation.
2. **Trust and Compliance:** bridges the gap between the need for regulatory compliance and the privacy-focused ethos of the cryptocurrency community.

What zkKYC can be used for?

Decentralized Finance (DeFi) and Beyond

- **Identity Verification:** zkKYC enables on-chain identity verification, allowing users to interact with DeFi protocols, and DEXs, and participate in token sales and DAO administration while maintaining privacy and complying with AML/CTF regulations. This ensures secure and anonymous participation together with considering member contributions and trustworthiness across multiple Web3 platforms.

- **Reduced Compliance Costs:** automating the KYC process, reducing manual verification needs, and lowering regulatory compliance costs for platforms.

- **Fraud Prevention:** zero-knowledge proofs to prevent fraudulent activities, ensuring only legitimate participants trade on platforms.

- **Privacy-preserving on-chain scoring:** KYCed users may be scored to evaluate creditworthiness without revealing personal information which allows building under-collateralised lending protocols

Off-Chain Use Cases

- **Age Verification:** confirms users are over 18 for accessing certain websites or services, such as social media platforms or age-restricted content, without revealing their actual birthdate.

- **Service Access:** verifies conditions for using services like online gambling or bookmaking, ensuring compliance with legal and regulatory requirements while maintaining user privacy.

- **Gaming and Metaverse:** facilitates secure in-game transactions and the exchange of virtual assets through verified identities ensuring compliance with age restrictions and other regulations while protecting privacy.

By adopting zkKYC, businesses can streamline their KYC processes, enhance privacy and security for users, and ensure compliance with regulatory standards, making it a crucial innovation for the future of decentralized finance and beyond.

Tech Design Properties & Features

The zkKYC framework is designed to ensure privacy, security, and compliance through the use of zero-knowledge proofs and blockchain technology. Its properties include:

- **Privacy:** Users' personally identifiable information (PII) remains confidential and secure. KYC Provider that issued the record cannot associate it with the on-chain address that is using it.

- **Interoperability:** Integrates seamlessly with decentralized applications (dApps), smart contracts, and other Web3 technologies.

- **Open source**

- **Decentralised** – no single party, no centralized kyc data storage - cannot be hacked.

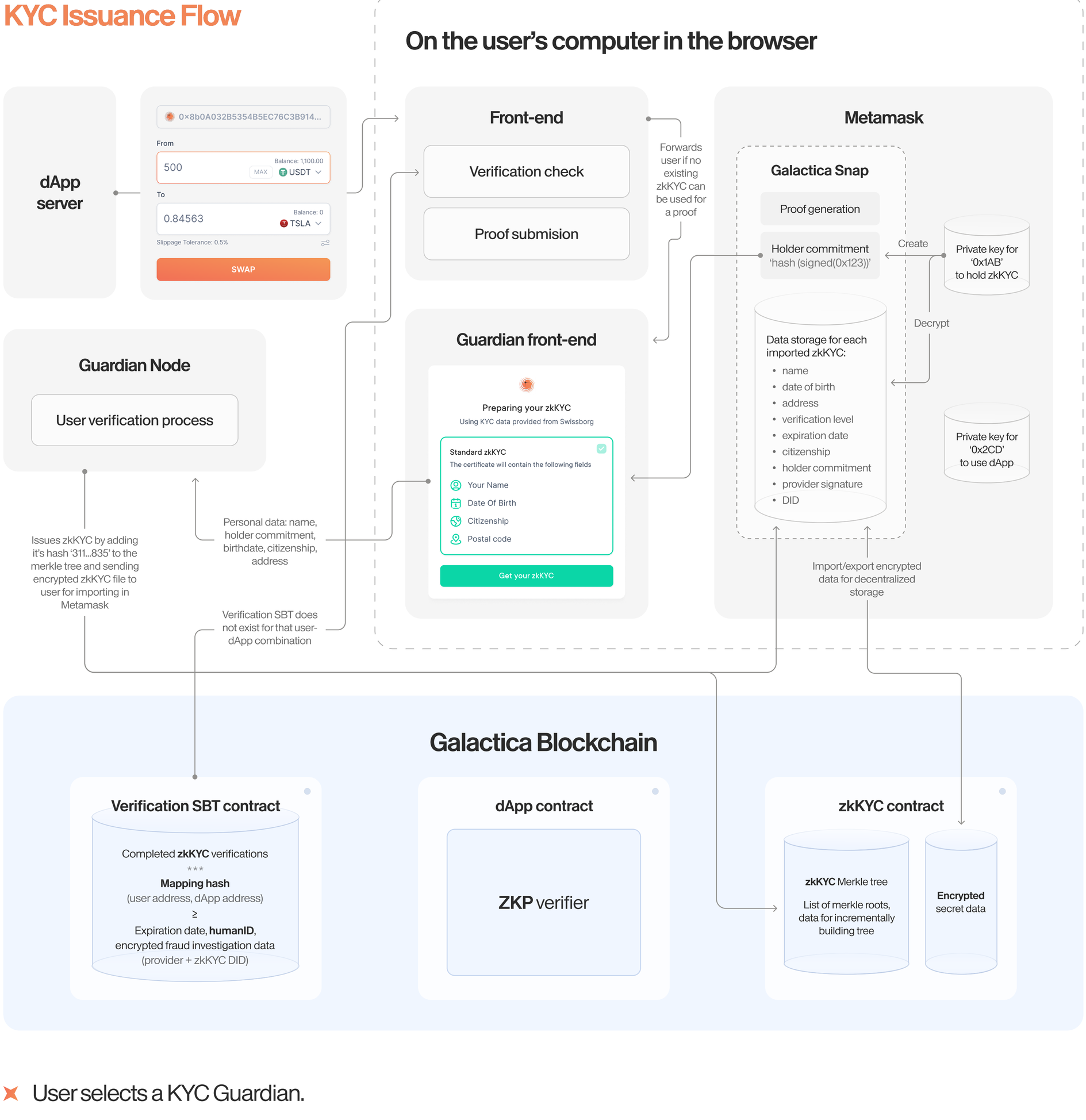
Technical Design

System Overview

1. **Galactica Node** – implements EVM that is crucial for the system operation
2. **Smart contracts**
   - GuardianRegistry – stores the list of approved Guardians
   - ZkCertificateRegistry
     - Stores the merkle root of the ZkCertificate merkle tree.
     - Allows Guardians to add or remove ZkCertificate from the merkle tree.
   - VerificationSBT: A global smart contract that stores verification SBTs, minted by dApp for users submitting zk proofs
   - Verifier wrappers
     - Dapp interacts with these contracts to get access to verifiers.
     - Check conditions on public parameters.
     - Specific examples: ZkKYC, AgeProofZkKYC, TwitterZkCertificate, TwitterVerificationProof, TwitterFollowersCountProof.
   - Verifiers: verify the zk proofs submitted by users through Dapps
3. **Circuit Library**
   - Complete circuits: for specific applications with generated onchain verifiers
     - ZkKYC: checks that ZkKYC exists.
     - AgeProofZkKYC: checks that ZkKYC exists and the age is greater than a certain threshold.
     - TwitterZkCertificate: checks that Twitter ZkCertificate exists.
     - TwitterVerificationProof: checks that Twitter ZkCertificate exists and the Twitter account is verified.
     - TwitterFollowersCountProof: checks that Twitter ZkCertificate exists and the number of followers is greater than certain threshold.
   - Helpers circuits: smaller parts of complete circuits
     - Authorization: verifies that the onchain message sender is authorized to use the zkCertificate record by providing the address signed by the account under the holder commitment of the zkCertificate record.
     - CalculateZkCertHash: Circuit to check that, given zkCertificate info we calculate the corresponding leaf hash.
     - Ecdh: generates an elliptic-curve Diffie-Hellman shared key.
     - EncryptionProof: proves that a message is correctly encrypted so that the receiver can read it.
     - HumanID: calculates the dApp specific human ID.
     - MerkleProof: verifies that merkle proof is correct for given merkle root and a leaf.
     - MimcEncrypt: encrypt/Decrypt extension of MimcSponge taken from https://github.com/iden3/circomlib/pull/16
     - Ownership: circuit verifying the ownership of a zkCertificate with a signature in the holder commitment.
     - Polynomial: computes the polynomial of degree k at n points.
     - PoseidonSponge: poseidon sponge hash of a message split into blocks of 31 bytes.
     - PrivToPubKey: circuit for deriving the public key of a private key.
     - ProviderSignatureCheck: circuit verifying that provider signature is correctly submitted.
     - ShamirsSecretSharing: generates Shamir's Secret Sharing shares, it takes a secret and splits it into in fragments, of which k are needed to reconstruct the secret.

4. **Metamask Snap** – an extension to MetaMask wallet allowing to securely store and use ZK Certificates. Implements methods that dApps can use to generate and publish ZK Proofs.
5. **Merkle Proof Indexers** – backend services intended to speed up proof generation process on end-user devices (Index Merkle Trees and provide Merkle Roots to clients).
6. **Guardian SDK** – library implementing common methods a KYC Guardian should use to issue and/or modify KYC records on-chain (work with our Merkle Trees). Used by KYC providers to integrate with Galactica and become KYC Guardians.

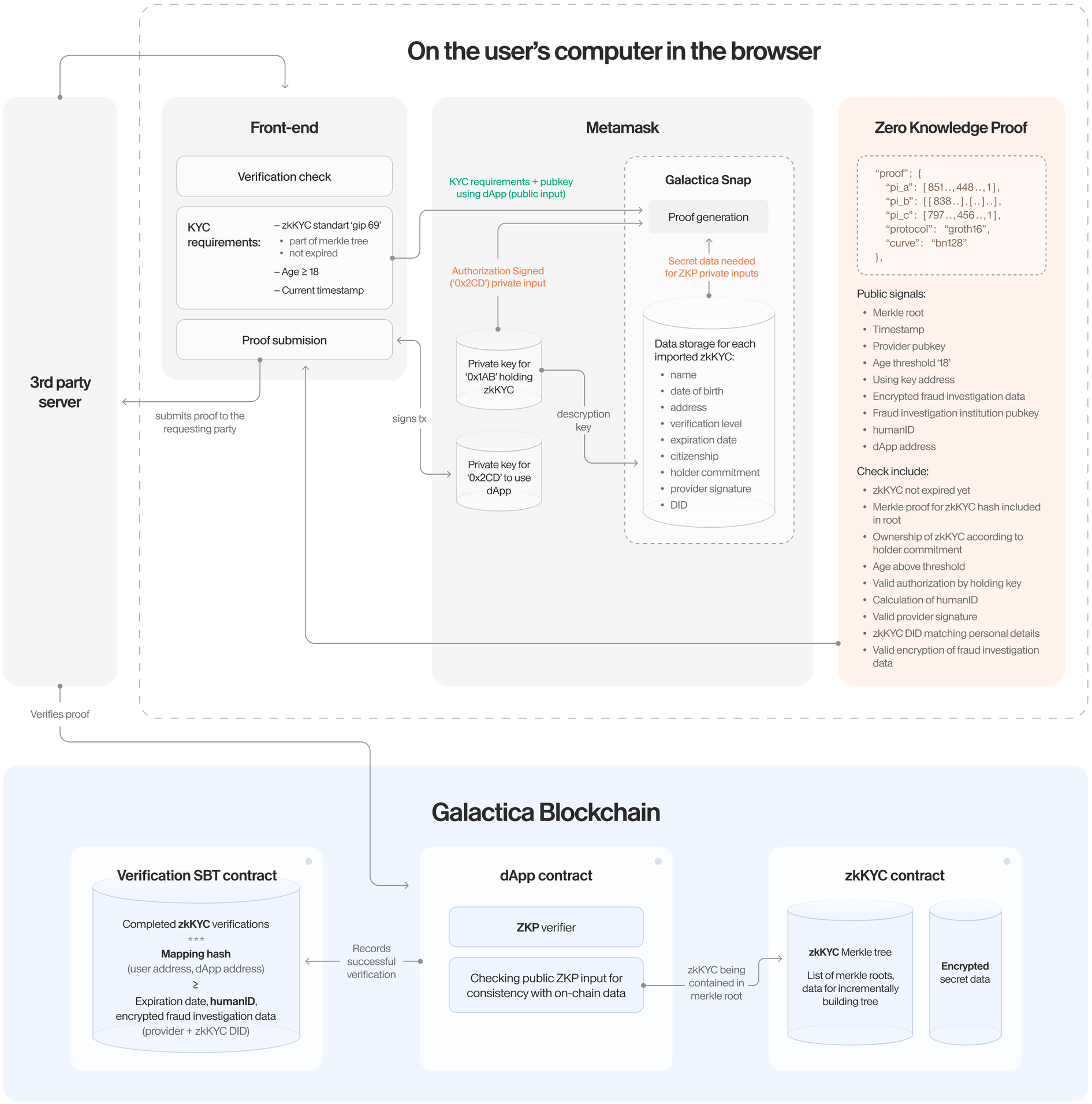
KYC Issuance Flow



- User selects a KYC Guardian.
- User generates a commitment hash, derived from his mnemonic phrase.
- User opens KYC Guardian Front End with a commitment hash, which is used by Guardian later to issue the KYC record.
- User provides identification documents to the KYC Guardian, or the Guardian uses those documents that were provided before.
- Guardian verifies the documents if needed.

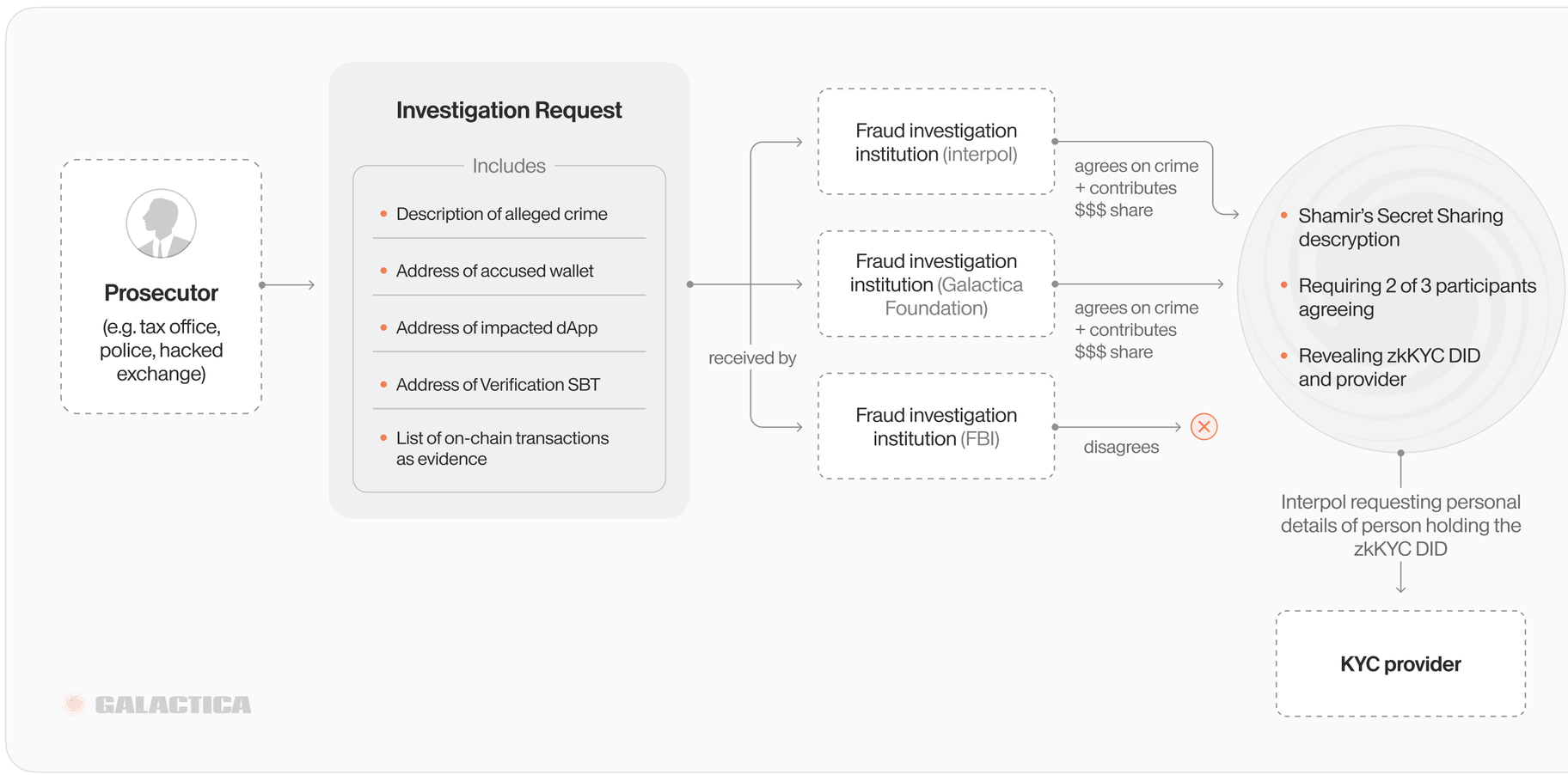
- Guardian calculates a leaf hash of a user's KYC data and posts it with TX to the Merkle Tree on-chain.
- Guardians issues a Secret file with the user's KYC data and sends it directly to the user (allows the user to download the secret file on the Front End).
- User uploads the secret file to the Wallet (non-custodial).
- From this point the KYC may be used. DApps can utilize the wallet API to request generation of required zkProofs.

zkProof Issuance Flow



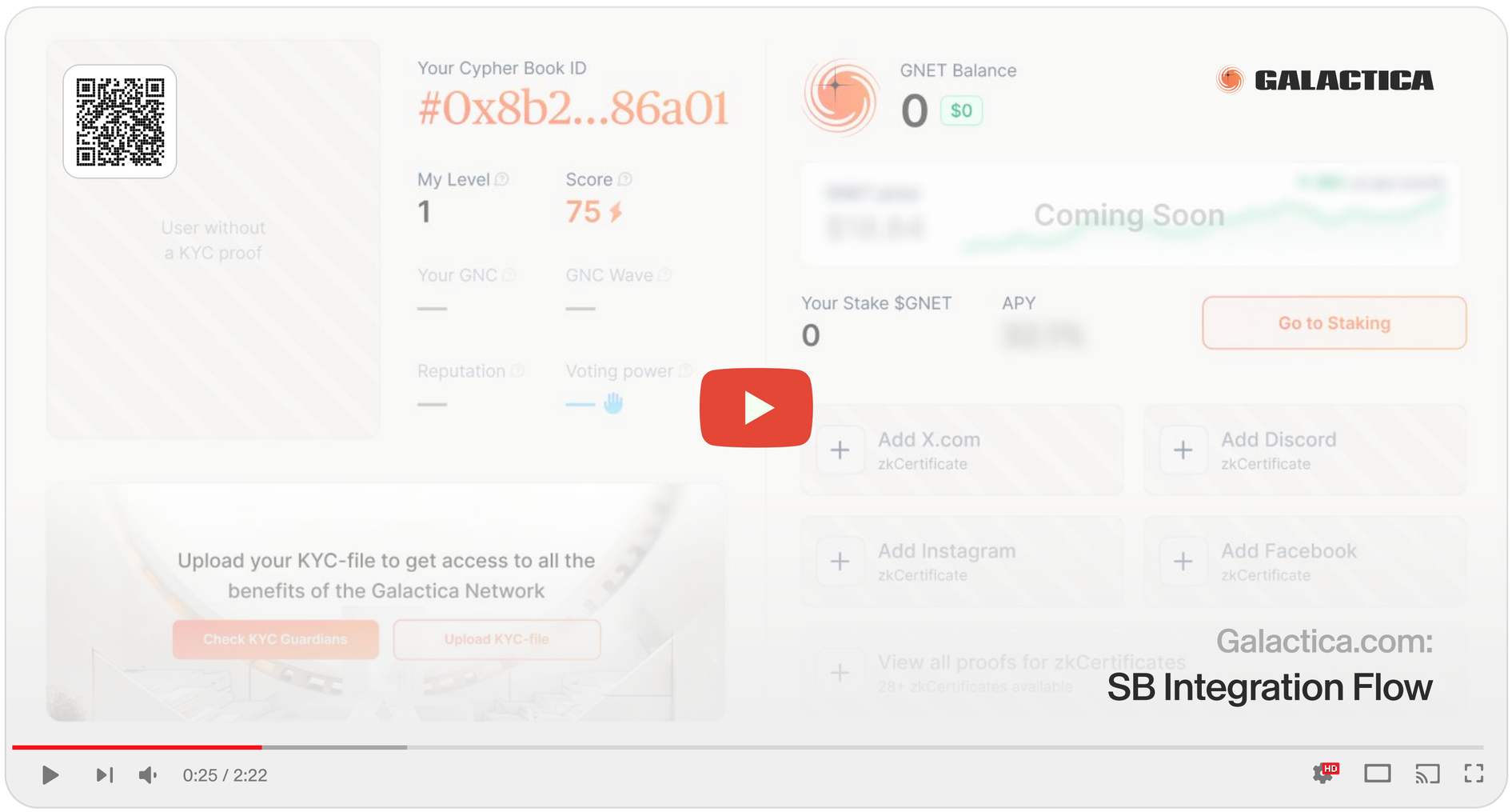
- dApp FE asks for a certain statement to be proven. The requirements are sent to the wallet API.
- Wallet generates zkProof using Holding Key (derived from mnemonic) and KYC data (uploaded with a secret file).
- Wallet sends the zkProof back to the dApp.
- dApp verifies the zkProod using Galactica's ZKP Verifier (A special SDK in some time).
- dApp FE forms the TX to publish the zkProof to the network (if needed).

Fraud Investigation



- A prosecutor requests the involved institutions.
- If they reach a consensus, they can recover the user's ZK certificate DID and the provider that issued it.
- With this DID, prosecutor can query the personal data of the user from the provider that issued the zkCert.

Integrations (SwissBorg)



Demo

zkKYC Documentation

Zero Knowledge KYC | DApp-Specific HumanID | Holder Commitment | KYC Guardian

Verification SBT | Fraud Investigation Process | Privacy Precautions | Miro Scheme

MME Case Study

Traditional KYC methods pose challenges to DeFi's privacy and efficiency. Galactica's zkKYC aims to address these by enabling privacy-preserving identity verification on-chain. Recent article highlights the regulatory challenges of implementing zkKYC in Switzerland, particularly its compatibility with Swiss Anti-Money Laundering (AML) laws.

The core issue is the lack of standards and jurisdictional applicability for zkKYC. While Swiss regulations extend to some DeFi platform providers, fully decentralized systems are exempt. zkKYC alone doesn't fulfill KYC obligations as it doesn't share verifiable identity data with financial intermediaries.

Galactica's solution involves the Guardianship model, where trusted entities verify compliance information on-chain while keeping user data off-chain. Users maintain custody of their data and create private proofs for transactions, enhancing privacy and compliance. This approach balances the regulatory needs with DeFi's ethos of decentralization and privacy.

	Compliance Oracles	Galactica KYC Guardian
Documentation	Article [1]	Documentation [2]
Role in zkKYC	Trusted company verifying identity documents of customers off-chain	Trusted company verifying identity documents of customers off-chain
Role in AML and CTF regulation	Delegated provider for implementing the KYC identification process and keeping required documentation for fraud investigation	Delegated provider for implementing the KYC identification process and keeping required documentation for fraud investigation
Communication process to confirm statement to financial intermediary/DeFi	Direct Compliance oracle sends signed ZK proof of the statement to the financial intermediary/DeFi	Indirect Guardian issues verification hash of KYC data on-chain (zkCertificate). Customer generates and sends ZK proof of his data satisfying the statement to the financial intermediary/DeFi
Privacy benefits	Financial intermediary does not know any more personal details than required	Financial intermediary does not know any more personal details than required
Privacy benefits	zkKYC proof hiding personal details on public blockchain	zkKYC proof hiding personal details on public blockchain
Privacy benefits	–	Guardian does not know about customers' on-chain activity (unless fraud investigation [4] takes place) because of holder commitments [5] and because users can create ZK proofs that are verifiable against the Merkle tree of zkKYC hashes
User experience benefits	Convenient reuse of a completed KYC for multiple services	Convenient reuse of a completed KYC for multiple services
Business operation benefits	Financial intermediary does not need to verify ID documents, store sensible ID document data, or handle it securely	Financial intermediary does not need to verify ID documents, store sensible ID document data, or handle it securely

Full review can be found in the following article: https://galactica.com/news/zkKYC-in-Decentralize-Finance