Connecting Blockchain Addresses with Trusted Entities

Attila Perez

attila@vericha.in

December 2020

Abstract

Public accountability and trust in entities are necessary parts of a working decentralized financial (defi) ecosystem. Vericha.in aims to fill a gap in the defi space by providing a living and breathing, publicly verifiable source for blockchain address ownership. This is achieved using a combination of cryptographic digital identity verification techniques as well as an open backend that allows for vetting of the service's practices by third parties. The service never stores persistent addresses, only hashes, creating a one-way lookup system. Scraping of an entity's owned addresses becomes a fruitless endeavor. An end-user-facing API allows for integration into third party wallets and public projects.

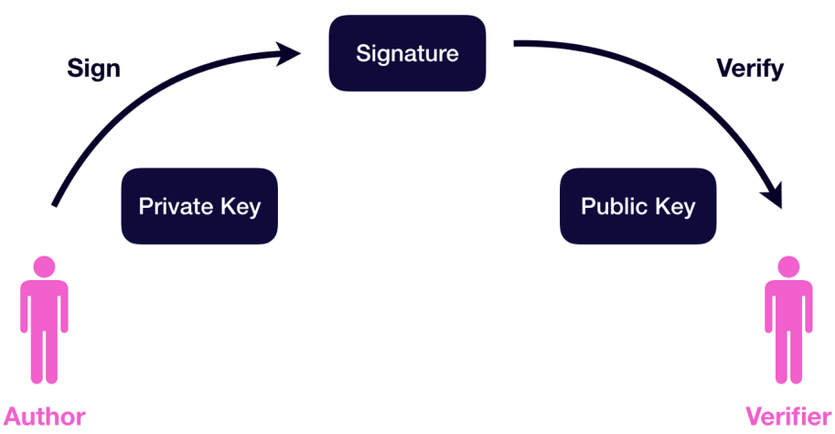
> As society becomes more and more complex, cheating will in many ways become progressively easier and easier to do and harder to police or even understand. There should be whitespace between paragraphs.
>
> -Vitalik Buterin, co-founder, Ethereum

Requirements For Such a Service

1. Completely public codebase, including each vector (frontend, backend, open-source API tools) ☑
2. Capable of defense from scraping of an entity's addresses ☑
3. Third-parties are able to verify the legitimacy of entities ☑

The Solution

Harnessing The Power of Preexisting Cryptographic Frameworks



Cryptographic signatures are built into web3js, allowing ECDSA-method signed messaging completely off-blockchain. This means that there is no gas requirement, decreasing cost per function greatly.

Address Accreditation

- Entity Verification
  - Public cryptographic (RSA) key available on entity's web presence
  - Randomly-seeded encrypted message is requested for verification
    - Entity is provably linked to account creation on service
    - API authentication is generated for the account
- Address Ownership Verification
  - Entities are tasked with creating signature objects from originating private keys
    - Web3js is utilized to recover from the signature object the corresponding public address
      - SHA-512 hash of address is added to indexed database, linked to entity ID
- End-user Lookup
  - API can be integrated into wallets or other applications, where the addresses sent to the lookup API are prehashed on the local machine to decrease computational energy.
    - Entity ID is referenced to entity name and the result is returned to the requesting party.

The Future of vericha.in

As the service does not take commissions from any parties in its current state, publicly offered voting tokens will be used initially for funding. **VRCN** token holders will be able to create proposals and vote on issues concerning the future of the service.