Disclose: An open protocol for a Decentralized KYC-certified Authentication System on the Ethereum Blockchain

Louis Guthmann – Stanislas de Roquemaurel-Galitzin – DO NOT SHARE

February 8, 2019

Abstract

We implements a number of 0 knowledge protocols that offers a simplified permission layer on top of Ethereum, replacing effectively existing Identity ERC proposals. The protocol seeks to serve as an open standard to lower friction in permissioned environment, especially for KYC certification. It allows Users to prove themselves to Companies, leveraging trusted third parties known as Authorities.

1 Context

Public blockchains have proved their revolutionary functionalities for the past nine years. Those functionalities include, but are not limited to, transparency and proven authenticity over a shared and distributed database. Both features have been central to the development of concepts such as decentralized trust, decentralized transactions and decentralized organizations.

Nevertheless, organizations and people live within a system of different laws and regulations, which they must comply with. As part of the evolution of crime during the 20th and 21st century, anti-money laundering policies and counter terrorism measures have increasingly become more complex and demanding. Since the Bank Secrecy Act of 1970 to the Intelligence Reform and Terrorism Prevention Act of 2004, we are expected to prove who we are and the origins of our money, a process known as KYC (Know-Your-Customer). The KYC issue is acute for online financial services and at the cost of companies. Depending on the level of risks one represents and the amount of money one manages, the requirements will differ between companies and often within a company, between products.

This process has proven to be frustrating, costly and insecure for both Users and Companies.

This statement stands both in an Off-Chain and On-Chain setting. We define Off-Chain as any interaction that happens without involving a Blockchain (aka physical or classical Internet interaction). On-Chain is any transaction which occurs on a Blockchain. This dichotomy is important because a public Blockchain is an enclosed environment, which cannot trust anything coming from outside of it, such as explained by the Oracle Problem.

Let's start with the Off-Chain KYC analysis:

- Frustrating - Users have to provide information, including their Passport, proof of residency and photos to multiple Companies, making the process redundant and frustrating in terms of User Experience.
- Costly - Companies (both big and small) have to inject a great amount of work, which they cannot fully automate. Some decide to internalize the competency, whilst others outsource the operation management. Even when a User wants to open an account in Bank A and fills it with their own money from Bank B, Bank A cannot re-use the KYC of Bank B, even though they trust one other. The same data is processed multiple times and the global cost of handling this collectively adds up.
- Highly insecure - Some users get their identity stolen, often by having their Personal Identifier (PID: Passport, ID, Social Security Number, etc.) stolen through a Company hack and directly from them. In both cases, an individual is often powerless and at risk of an Identity Theft.

In an On-Chain setting, KYC only exists in a very restricted form, as a Whitelisting mechanism at ICO time. Users commit a KYC to the company and provide a BTC/ETH address associated with it. The addresses are uploaded to the Smart Contract, and modification of the Whitelist requires the intervention of the Smart Contract owner.

Let's dig further into Whitelisting for On-Chain KYC.

- Frustrating - Users would have to KYC each of their wallets and ask their Smart Contract owner to update the whitelist. Additionally, adding an address later on would require the user to repeat the same mundane process.
- Costly - In conjunction with the Off-Chain scenario, the current On-Chain KYC cannot handle the form of permission transactions that legislation requires, as it is required for the Smart Contract owner to update the whitelist on a continuous basis.
- Insecure - In addition to the Off-Chain scenario, mixing Wallets and Identity into one concept would create a bigger point of failure.

To this end, On-Chain KYC is just as frustrating, costly and insecure as Off-Chain KYC.

Decentralized KYC-certified authentication systems are an important improvement over this current state.

- Allowing users to reuse a certified KYC by using a single Authentication

- Lowering the management cost of KYC for companies
- Enforcing better security by reducing the number of entities, accessing PID, and allowing easy revocation via the update of a single Smart Contract.

Civic, SelfKey and Traceto, to name a few have made great progress to simplify KYC over ICOs and exchanges. While honorable, they solely focus on Off-Chain KYC, a much simpler case and easier to solve. On the other hand, On-Chain KYC has not been extensively studied. On a first attempt, Zeppelin recently presented their TPL paper, where Users' KYC can be enforced by the Smart Contract itself. While a great protocol, leveraging years of experience of Internet certificates, TPL could turn out to be very difficult to manage for exchanges (Centralized and DEX), does not integrate the Off-Chain use case and could face serious issues under European GDPR as the status of metadata is still unclear (as illustrated by PICOPS shutdown).

On-Chain KYC can be seen as a form of permissions management over the network.

By merging Off-Chain and On-Chain cases, while preserving privacy, Disclose solves KYC once and for all. In the rest of this paper, we will focus on On-Chain KYC, as this is the hardest case and a founding block of security tokenization.

2 Existing Work

This section is largely inspired by 0x compliant Peer to Peer trading post and contains assumptions based on 0x claims.

On-Chain KYC implemented on top of Ethereum has been limited so far to ICO. As explained previously, the state-of-art method is whitelisting of wallets. This methodology has failed to generalize due to inefficiencies and frictions, making it difficult for Users to apply and interact with those companies (as the example of 0x ICO and their usage of Civic shows). As Harbor, PolyMath and others are showing, interests in securities tokenization are high but KYC frictions seems to be under-addressed as they seem to believe whistelisting will be scalable.

To make the whitelisting mechanism more scalable, Zeppelin presented TPL, discussed above. TPL can be interpreted as a combination of the TLS/SSL protocol heavily used by HTTPS and a DNS server. It enforces compliance through a set of trusted third-party certificate authorities to establish the authenticity of KYC. This allows a fast pruning of bad authorities, the establishment of decentralized jurisdictions, setting a framework of analysis for permissions management and fast permission verification through a smart contract callable wallet registry.

While a clear improvement over the current state, we, at Disclose, would like to argue the whole whitelisting approach is wrong and can be improved.

As a start, we discard solutions envisioned by current security token companies, as they expect whitelists for each smart contract, making it very difficult for a user to interact with a great number of smart contracts. Now, let's discuss the solution proposed by TPL. As stated above, TPL directly inherits its architecture from TLS/SSL. TLS/SSL and X.509 certificates are solving an issue called the Public Key Infrastructure (PKI). The question PKI is solving is the following: "How do I know that the public key I'm talking to is the one of the entity I want to interact with?". We require a PKI to link names to public keys because we would otherwise risk undetected Man-In-The-Middle attacks (MITM). The way TLS/SSL solves this issue consists in having an extremely trusted certificate authority known as Root Certificate such as Microsoft installed in browsers and delegating their power to other smaller Certificate Authorities for scalability, which can also delegate their authority. Each certificate contains the certificate Authority public key. As such, we have chains of certificates signed from the smallest up to the Root Certificate. A User can therefore verify the chain of signatures and trust the public key to be legit.

In the case of TPL, the Public Key becomes the Wallet address and certificates are divided between the registry and the rules of the jurisdiction. While an interesting take on this subject coming from the Blockchain world, we already discussed that PKI seeks to protect against MITM attacks. Those attacks cannot happen on a Blockchain as every information miners require to do their work exist inside the Blockchain. As such, the system is wrongly suited for the task at hand.

In addition, TPL and more generally, Whitelisting mechanisms are about to encounter serious implementations issues:

- Centralized exchanges usually perform transactions by generating new wallets on the fly. This works perfectly fine under the utility ERC-20 protocol. Tokens are permissionless in a similar fashion to Ethereum. As such, any newly created wallet can receive and send tokens. In the case of permissioned tokens, this could prove to be hard. Exchanges would have to whitelist a great number of wallets in advance. This whiteslisting must append off-chain and therefore, could slowdown or even render the whole system impossible.
- Among the three major types of DEX (embodied by 0x, Kyber Network and EtherDelta/ForkDelta), two (Kyber Network and EtherDelta/ForkDelta) are in jeopardy under the current model. According to the 0x compliance blogpost discussed above, those exchanges are required to deposit tokens into their Smart Contract. As such, they will not be able to whitelist users, as everyone interacting with them would be able to trade tokens with the necessity to be permissioned.
- Any new wallet would need to go through an Off-Chain KYC, creating friction and slowness for the system.

To wrap things up, the underlying issue lies within the fact that the industry has been framing the KYC issue in a questionable fashion and notably, decided to combine the notion of Identity and Address together in order to mimic the successful TLS/SSL protocol.

This can be greatly improved by changing the perspective on such matters. TPL seems to take the problem as a framework and organization issue. As such, TPL perceives Wallets as members of a rules-constrained organization. We think of KYC, and more generally of compliance, as a permissions issue.

This approach is also taken by recent ERC 725, 731 and affiliates. They do pose a real implementation complexity and a strong privacy breach by revealing information in clear, making it very easy to do forensics on them. Disclose comes up with a newer approach, offloading the management to existing cryptographic primitives, bringing both privacy and full control over submission to users making GDPR compliant.

3 Our Approach

3.1 The Game

Regulatory compliance describes the goal that organizations aspire to achieve in their efforts to ensure that they are aware of and take steps to comply with relevant laws, policies, and regulations. Based on this definition, compliance is an organizational goal. Compliance regulations are externalities to businesses and complying with them does not involve being useful for the business per se.

As such, companies can decide to internalize or externalize their compliance verification. Disclose offers a protocol and a set of tools to differentiate each part of the compliance game and provide proof of compliance, whilst minimizing the need for trust between each player of the game we describe below. We describe here one of the protocol Disclose plans to implement. Others are either reduction or generalization of this said protocol.

Let's introduce the compliance game we discussed earlier:

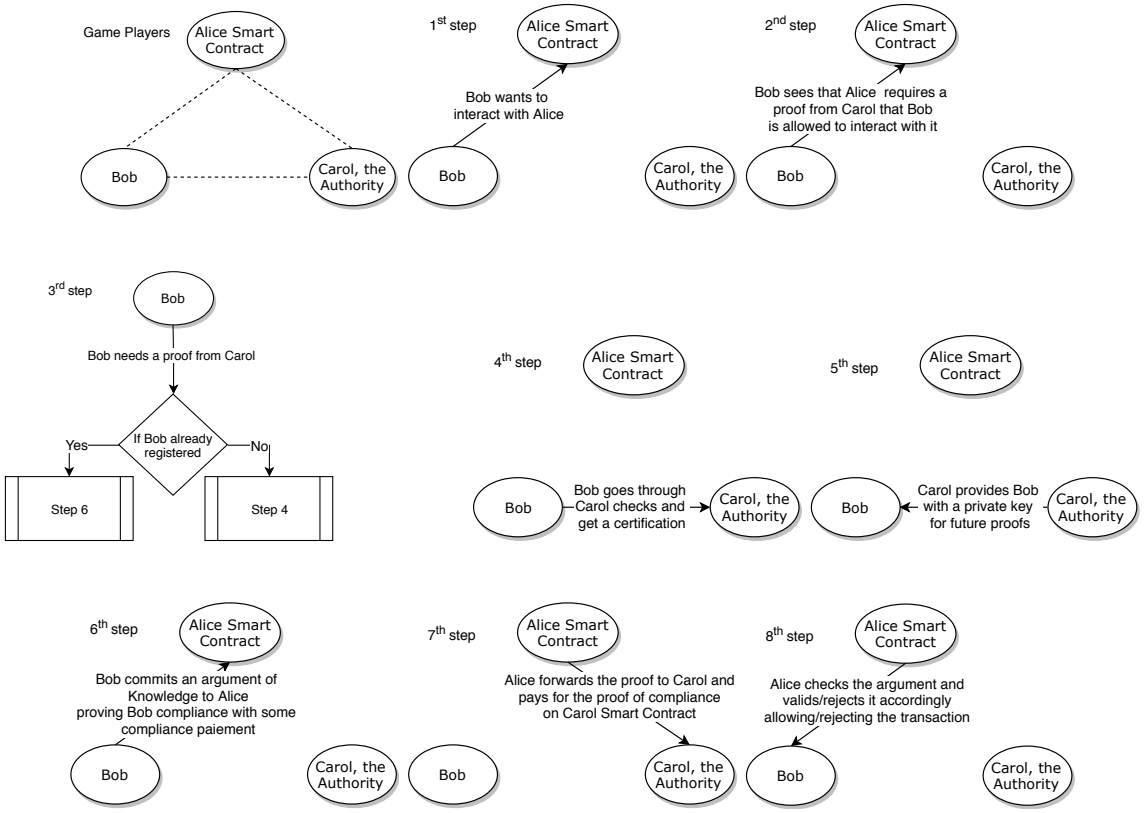
- There are three parties: Bob, the user; Alice, the company which operates a Smart Contract; Carol, the trusted authority. For the sake of the game to function under the hardest rules, Alice and Carol cannot be the same entity. Carol's role can be broken down into three parts: key issuer, key revocator and proof opener. For simplicity, we won't distinguish between each role.
- Bob needs to prove to Alice that he complies with Alice's requirements but does not trust Alice to handle his information. Without this proof, Alice won't let Bob issue any transaction.

- Bob and Alice trust Carol as a third party certification authority.
- Bob and/or Carol must be able to revoke Bob's identity.
- Carol must not be able to prove herself as Bob.
- Every communication is known to the public but no one except Carol should learn anything about Bob from the game's transactions.

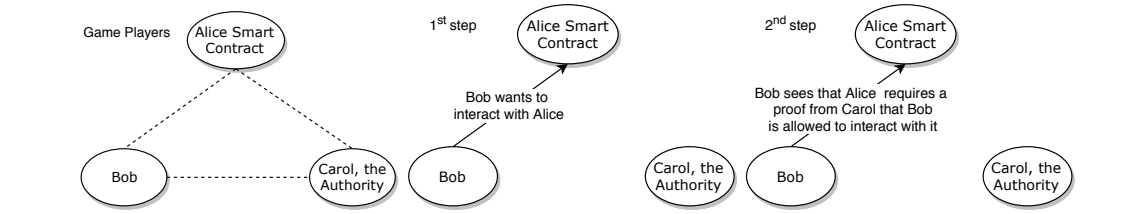
Under cryptographic terms, Disclose resulting proof of knowledge can be translated as a set of properties: Sound, Complete, Anonymous, Revocable and Framing-proof from any party.

3.2 The Protocol

To solve this game, we imagine a set of interactions both off-chain and on-chain. We focus on presenting the scenario under one primitive we planned to implement: Group signatures.
Here is the whole process:

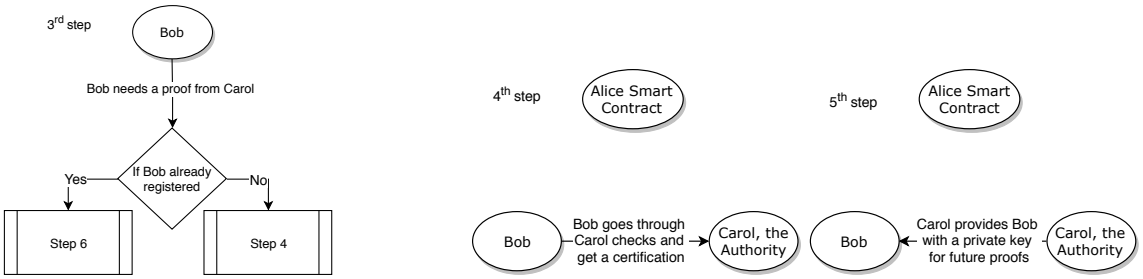


Let's dive into the inner working. This first set of figures presents the starting point of the game: three parties with different objectives. Bob wants to interact with Alice's Smart Contract. Alice wants to maintain a compliant ledger of all tokens' ownership and Carol is willing to take the liability for Alice's account.

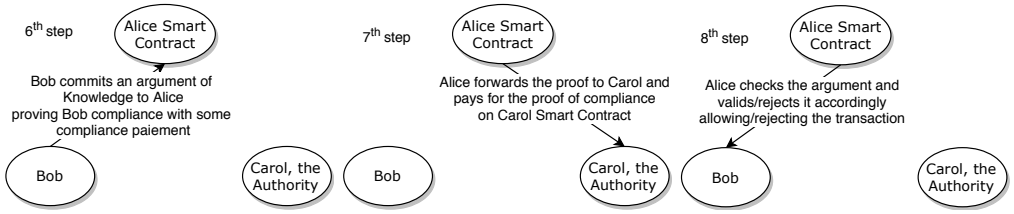


This second set of figures presents the on-boarding scenario. If Bob is not registered with Carol for the level of compliance required by Alice, Bob goes to Carol to get KYC'd.
Compliance levels correspond with a set of rules defined by Carol and Alice and should be standardized according to current laws. Disclose protocol does not pretend to preset law requirements

and expects authorities to set and modify them according to their expert knowledge. A compliance level can be assimilated to a product sold by the authority that users and companies would use/pay for.
As a result, an interactive process is set to provide Bob's private key while maintaining Exculpability. This happens off-chain and works according to Carol's internal processes.



This third set of figures presents the On-Chain identification mechanism. For simplicity, we assume that everything happens On-Chain. This could however be optimized for gas efficiency.
The compliance game continues as Bob provides a proof of his compliance to Alice alongside the transaction itself, including Carol Smart Contract fees. Alice's Smart Contract forwards the proof to Carol's Smart Contract and commits the proof and the payment provided by Bob. The fee settles the commercial relationship between Alice and Carol.
Carol checks that the message is correctly formed (various different requirements are needed to deter cheating from malicious companies and other technicalities). At the same time, analysis for fraud detection and other technicalities can be performed Off-Chain.
Once the message and the proof are valid/rejected, Carol informs Alice to allow/prevent the transactions. In case of dispute by any form of regulatory entities, Alice can go to Carol to obtain



the required information. The mechanism does not reveal anything more than the identity of the inspected user (mentioned above), preventing leakage and full access to all of Alice's users. If required, Bob can revoke his identity or in the case of lost/theft tell Carol to do it on his behalf.

Disclose leverages various zero knowledge cryptographic primitives such as Anonymous Credentials, Group signatures, Attribute-based signatures and extensions and uses pairing-based cryptography as available on Ethereum since Byzantine. Its main advantages over Identity ERC are

- Simplified management
- Fully managed OffChain
- Optimal privacy
- Embeds proof into the transaction data (simplifying onboarding and allowing GDPR compliance)
- Reusability across addresses, blockchains and OffChain scenario
- Smaller memory footprint
- Optional authority decentralization
- Delegation of permission to other parties
- Potential cheaper submission via relayer network

Disclose takes a direct inspiration from the OpenID protocol. As such, it delegates the whole compliance issue to Authorities such as Carol. This allows a much greater deal of flexibility to the compliance mechanism while providing the liability holder with the required regulatory precautions.