zkKYC
A solution concept for KYC without knowing your customer, leveraging self-sovereign identity and zero-knowledge proofs

Pieter Pauwels

pieterpauwels@protonmail.com

June, 2021
Version 1.0

Abstract

Businesses that are subject to AML/CTF regulation must meet their KYC obligations. In this context, to establish and verify a customer's identity, the customer is required to share personal information with these businesses. This creates a Pareto dominated situation where a customer's privacy is typically traded off for the mandated transparency requirements. In addition, this privacy erosion also reduces the security and safety of the customer as shared personal information can be passed on or stolen and used against the best interest of the customer (e.g. identity theft). Recent innovations in self-sovereign identity and zero-knowledge cryptography, along with proper ecosystem design, allow for a novel approach to KYC that protects the customer's privacy without reducing transparency. The proposed solution concept, zkKYC, removes the need for the customer to share any personal information with a regulated business for the purpose of KYC, and yet provides the transparency to allow for a customer to be identified if and when that is ruled necessary by a designated governing entity (e.g. regulator, law enforcement). This approach breaks the traditional privacy vs. transparency trade-off and provides structured transparency, resulting in a net positive outcome for all parties involved.

Keywords: privacy; structured transparency; know-your-customer (KYC); zero-knowledge proof (ZKP); self-sovereign identity (SSI)

1 Introduction

Background Privacy is a multifaceted topic and exists on a continuum. Some people are very protective of their privacy, even at a high cost. Others care very much in principle, but easily trade off their privacy for the benefit of convenience. Last, some people argue transparency always trumps privacy for the sake of security (”you would only care for privacy if you have something to hide”). Wherever one sits on this spectrum, it is obvious that privacy is typically a matter of trade-offs. At times transparency is required for regulatory reasons (e.g. to fight money laundering and terrorist financing) or for the greater good (e.g. medical research) and sharing personal information is considered an acceptable and necessary sacrifice. Simultaneously maximising privacy and transparency is the challenge for legislators, regulators, thought leaders and for each of us as individuals in our daily choices.

Challenge The specific challenge addressed in this paper is how to protect the privacy of customers when on-boarding at a business, while simultaneously providing transparency to the business. The transparency enables a business to meet the know-your-customer (KYC) obligations they have under anti-money laundering and counter-terrorism financing regulation (AML/CTF). For a business that is not regulated under AML/CTF, the transparency enables them to legally identify their customers in case an adversarial situation (e.g. breach of contract, fraud) would arise. One can argue that in the current state the price we pay for this transparency is disproportionate, that the cost outweighs the benefits. Each of us is obliged to share our personal identifiable information with each (regulated) business we interact with, eroding our privacy, security and safety for a fighting chance to catch a few bad actors in the future. Is it not possible to only disclose the personal information of identified bad actors or those under strong suspicion? Innovative technologies combined with proper ecosystem design can help us break this privacy vs. transparency trade-off and shift the Pareto frontier.

Objectives The key objective of this paper is to present an alternative solution concept for KYC, one that is more human-centred, does not rely on upfront sharing of personal information with businesses and still enables a customer to be identified if and when that is required. This proposal does not constitute a detailed solution design, nor does it claim to meet all regulatory requirements or cover all possible edge cases. Rather, it results from a thought experiment that aims to challenge the status quo, to trigger new ways of thinking about an old problem and to question how we can do better by leveraging a modern set of technologies. The paper explores the feasibility of addressing the aforementioned challenge. Technical concepts relevant to the proposed solution concept are identified and explained to a level of detail necessary to understand their relevance and contribution. References provide an option to go into more detail.

2 Problem Statement

Know Your Customer Businesses in specific industries are required to comply with AML/CTF regulation. In some jurisdictions this is limited to financial services organisations, in others this is also applicable to lawyers, accountants, real estate agents and more [PwC17]. AML/CTF regulation requires that these regulated entities verify their customers' identity prior to providing them with a designated service and perform ongoing Customer Due Diligence (CDD). Different customer attributes, as well as the nature of the product or service that a customer wants to acquire or consume, will inform the reporting entity of the level of risk for money laundering and terrorism financing they might get exposed to. This identified risk level may result in additional checks and screenings and can trigger the reporting of suspicious activities and transactions to the appropriate regulator(s).

While KYC activities such as basic identity verification mostly take place in entities under AML/CTF regulation, they can also be (partially) implemented by businesses as part of an overall good governance practice and operational risk management policy. Prior to engaging with new customers or entering into a contractual relationship, such businesses will then execute a basic identity verification. This can inform the business of any unanticipated risk exposure, but will also provide the required input into any legal proceedings that could result from future disputes or contractual breach by the customer.

For the purpose of this paper, we will limit the scope to KYC by regulated entities towards low risk, domestic retail individuals that request to consume basic products or services (e.g. opening a standard transaction account with no credit). This focus covers the bulk of KYC instances towards retail customers [GMW19]. Depending on the jurisdiction, this could include the following types of use cases:

- Finance: open a transaction account, purchase, sell or exchange cryptocurrencies
- Tax & Accounting: obtain advice on accounting or tax matters, register as beneficiary
- Legal: obtain legal advice, representation or support
- Telecom: acquire a mobile plan (pre-paid or post-paid)
- Real Estate: enter into rental agreement

Data Sharing In order to meet its KYC obligations, a regulated entity has to collect and verify a set of identity related attributes of the customer sourced from an exhaustive list of authoritative identity documents. Attributes usually include first name, last name and date of birth or residential address. Authoritative identity documents are often a driver license, a domestic identity card, a health insurance card or a passport. Even though only a subset of all identity documents' attributes must be verified, a lot more data is often requested to be shared, or scans of the full identity documents themselves, resulting in unintentional data sharing. This data may subsequently be verified with the Issuers of these identity documents or with identity verification services that have large connected databases of authoritative identity data. This process of sharing personal data for the purpose of identity verification happens for every regulated service that a user chooses to consume.

The identity attributes shared with and retained by the regulated entity might also have to be shared with third parties such as law enforcement or regulators. When a business wants to file legal charges against one of their customers, it must share customer identity attributes with law enforcement. When a regulated entity identifies suspicious activity, it is required to submit a Suspicious Activity Report (SAR) to the regulator for further investigation. Additionally, customer identity attributes may be passed along required regulatory reporting, even if not absolutely necessary, for the regulator to be able to identify suspicious activity across reporting entities.

Structured Transparency While the objective of transparency via data sharing to meet KYC obligations is well intended and has the protection of individuals and society in mind, there are three key problems that constrain us from achieving structured transparency. Structured transparency is the use of information while preventing misuse, such as the violation of someone's privacy [Tra+20]. These three problems are:

- **Copy Problem:** When users have to share personal information with each regulated entity that they engage with, it is impossible for them to control what these businesses subsequently do with that information. It can be copied, sold, misused, or may indeed be part of a hack or data breach anytime in the future. We have seen too many of these events being reported to be acceptable. This can result in negative outcomes for the users, putting their security, safety and privacy unnecessarily at risk.

- **Bundling Problem:** While AML/CTF regulations usually require only specific data attributes (e.g. name, address, date of birth) of a customer to be verified for KYC purposes, often much more personal data is collected and stored by the regulated entity. Sometimes because copies are taken of full identity documents, revealing more attributes (i.e.: no selective disclosure), or because more data points are considered necessary to perform proper identity verification (i.e. to avoid false positives).
- **Recursive Oversight Problem:** When users have to share personal information with a regulated entity, what governance protections do they have for their information? If data protection regulation does exist in a particular jurisdiction (e.g. GDPR), how is this enforced? How is the regulated entity held accountable for violating data governance obligations? If a regulated entity shares customer identity data with a regulator or other government agency, what privacy protection obligations are they subject to and who holds them to account?

These problems pose the following questions: can we develop an alternative approach to KYC that addresses these problems that erode our privacy, while still providing the transparency that regulators mandate to be able to fight money laundering and terrorism financing? A KYC solution concept that avoids having to share personal data at scale in the first place seems the most obvious way of preventing a lot of the identified problems. This paper proposes such a solution concept.

3 Solution Concept

Before presenting the proposed solution concept, it is valuable to provide a summary of how the current KYC process typically works. Subsequently, we will explore how the introduction of a self-sovereign identity model could address some shortcomings of the current system. Finally, the proposed solution concept, zkKYC, is presented.

3.1 Current state

In the current state, the end-to-end information flow supporting KYC is linear and sequential: personal information, often more than necessary, is passed along the chain, from one entity to the next. Figure 1 below provides an overview of how the different ecosystem roles interact, admittedly in a simplified fashion.

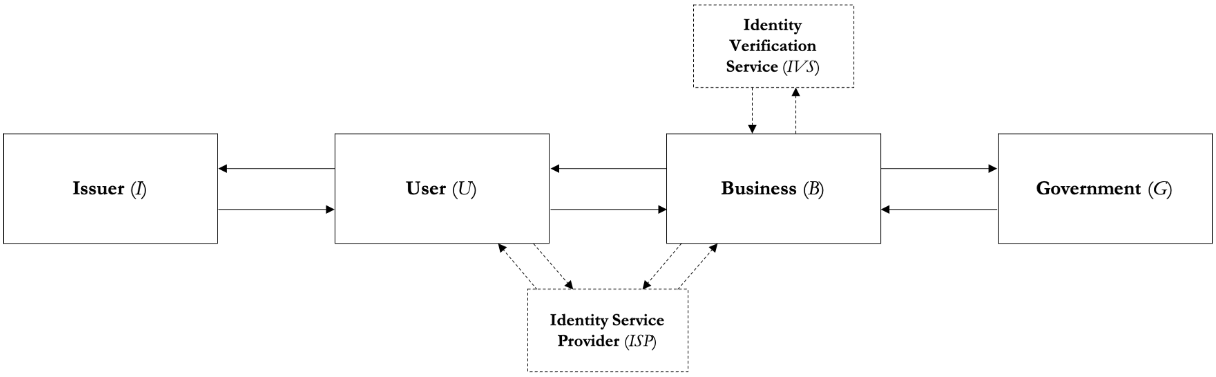
Identity Verification Service (*IVS*)
Issuer (*I*)
User (*U*)
Business (*B*)
Government (*G*)
Identity Service Provider (*ISP*)

Figure 1: Current state.

Concept The Issuer (I) issues identity documents. Usually this is a government agency (e.g. for identity card, driver license, passport) or an institution that is widely recognised and trusted in society. The Issuer issues these documents (typically in physical format) to the User (U). When on-boarding at a Business (B), the User presents the requested identity documents (physically, or a scan thereof). To be able to rely on the presented information, a Business verifies the authenticity and the integrity of the identity documents, or relies on a third-party Identity Verification Service (IVS) to verify it for them. Examples of a Verification Service Provider include Equifax, GreenID etc. If successfully verified, the User is on-boarded as a customer.

Government (G) is a role fulfilled by the regulators and law enforcement. The regulator defines the KYC obligations of a regulated Business and might require the Business to share certain (financial) transactions and associated User information with them, especially for suspicious transactions and Users (i.e. customers) or those under high risk of money laundering and terrorism financing. Law enforcement might also be engaged by a Business in case the User does not meet their contractual obligations and legal charges must be filed. For these reasons, a Business will share personal user information with Government, by choice or by obligation.

Structured Transparency In the current state a User has little control over what information is shared, or what happens to that data once shared. There is a necessary level of transparency for the regulated Business to comply with their KYC obligations, but a low level of privacy and control for the User. This applies to both analogue and digital interactions, where the digital interactions are often simply a digitised version of the analogue (paper based) interactions, for example electronic scans of physical documents. As personal information is copied and distributed across many parties, it becomes more difficult to securely protect that data, resulting in the potential for numerous hacks and data breaches. This risk only increases as large, centralised data repositories of sensitive data are created. Also, as the same personal identifiers (e.g. mobile, email address, driver license number) are shared across many interactions, it becomes easy to correlate an individual's interactions across businesses and track their behaviour. With many of those identifiers being public due to too many data breaches, this problem only grows.

It must be noted that some jurisdictions have taken action to address the privacy implications of (mandated and voluntary) data sharing, such as the General Data Protection Regulation (GDPR) in the European Union. It creates awareness that receiving and holding Personal Identifiable Information (PII) is a responsibility. In addition, some major institutions and regulators around the world have started to experiment with Privacy Enhancing Technologies (PET) to avoid having to "see" or access personal information for the sake of transparency [Max21]. While limited in scale and maturity, it is a positive and encouraging development. An interesting and relevant case study here is a project that the Australian Transaction Reports and Analysis Centre (AUSTRAC) set up to monitor financial transactions to identify crime. In this project AUSTRAC fulfils the role of Government in Figure 1. They have been working on a privacy-preserving encryption system that could enable them to examine patterns in financial transaction data of banks without ever accessing or seeing the underlying information [Lyn20]. This is a great example of attempting to tackle the recursive oversight problem in order to achieve structured transparency.

Digital To address the friction of the current approach in digital interactions, there are many initiatives globally that include the role of an Identity Service Provider (ISP), see Figure 1, often also called an Identity Provider. An ISP aims to improve the convenience for the User by creating a re-usable digital identity for them (based on successful verification of presented identity documents themselves, with help of the Issuer or by relying on an Identity Verification Service). This facilitates the KYC process as part of a digital on-boarding process for a User, as they no longer have to scan and upload their identity documents each time they are asked to prove their identity. Instead, they can now easily share digital identity attributes, verified once by the ISP. Depending on the trust a Business has in an ISP's identity verification processes (i.e. identification assurance level) and the implemented authentication and federation assurance levels [GGF17], they can skip the identity verification step themselves and fully rely on the presented verified information. This reduces cost and time for Businesses. The ability to do this does depend on the particular use case and associated regulatory requirements, which differ across jurisdictions. The issue, however, is that this does not address the aforementioned copy problem, bundling problem or recursive oversight problem. The model underpinning the KYC process stays the same. It mainly improves the convenience and efficiency of proving and verifying one's identity in digital interactions. This benefit for the User might come with the cost of reduced privacy if improper solution design or implementation allow the ISP to track the User's interactions with Businesses.

3.2 Enter Self-Sovereign Identity

Recently the advent of Self-Sovereign Identity (SSI), also called Decentralised Identity, has introduced a new paradigm in how we can manage and prove our identity in a digital context. In parallel it enables many opportunities that extend far beyond the domain of identity [Hel21]. As SSI is a core component of the zkKYC solution concept, it is worth exploring this model at a deeper level. This reveals that SSI, when implemented correctly, provides a range of benefits, some of which contribute to achieving structured transparency in the KYC information flow.

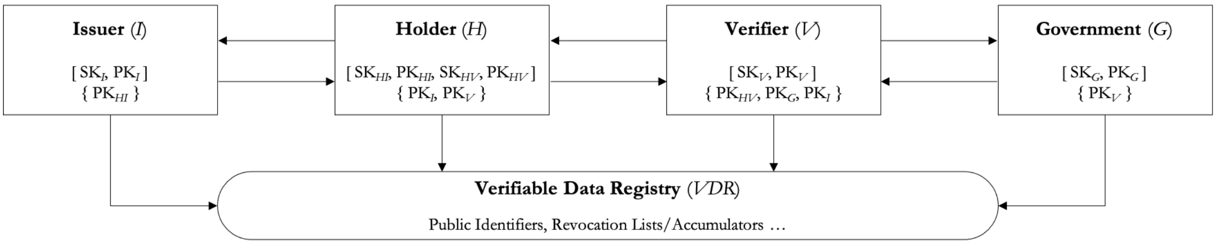

Figure 2: Self-Sovereign Identity and Government reporting.

Concept SSI is designed with digital interactions front of mind and puts the user fully at the centre and in control of their identity information. No longer must individuals rely on third parties to provide, control and manage their digital identity. Privacy is built in by design. Christopher Allen has famously captured the underlying principles in his 'Ten Principles for Self-Sovereign Identity' [All16]. The approach is very similar to how we have been handling physical identity documents, but it leverages modern technologies to improve it and generates new benefits to all participants whilst also allowing both physical and digital processes to align. Figure 2 provides an overview of a KYC solution concept when based on an SSI model. It is an extension of the traditional SSI model with the addition of the Government role. This role has been added to accommodate the KYC context of this paper. Issuers (I) can issue digital identity documents (in the format of verifiable credentials, see below) to an individual, who can securely hold them in a personal digital wallet they fully control. That is why they are called a Holder (H). Upon request, the Holder can generate a presentation of their digital identity documents to Verifiers (V), entirely or partially, who in turn can instantly verify them and rely on them with higher levels of assurance than was possible with physical documents. With regard to the KYC obligations of a regulated Business (a Verifier in this model) and how they are being met, this remains the same as in the current state. A Business still verifies specific identity attributes per regulatory requirements and stores that information for reporting purposes. However, SSI does provide intrinsic benefits that can contribute to achieving structured transparency in context of KYC.

Two new W3C standards are fundamental to enable this new approach: Decentralised Identifiers (DID) and Verifiable Credentials (VC). These standards are described in more detail below, as they are critical in generating the benefits of SSI.

Decentralised Identifiers (DID) [Ree+21] A DID can identify any subject (e.g. a person, organisation, thing, data model, abstract entity, etc.) that the controller of that DID decides that it identifies. DIDs have been designed so that they may be decoupled from centralized registries, identity providers and certificate authorities. Specifically, the design enables the controller of a DID to prove control over it without requiring permission from any other party. This contrasts with the vast majority of identifiers we know today (e.g. mobile number, email address, driver license number, passport number, etc.). Those identifiers are not under our control. They are issued by external authorities that decide who or what they identify and when they can be revoked. They impose control and dependency based on the reliance on them in multiple contexts and restrict commercial choice due to the complexity in changing service providers. They might disappear or cease to be valid with the failure of an organization. They can be fraudulently replicated and asserted by a malicious third party, which is more commonly known as "identity theft".

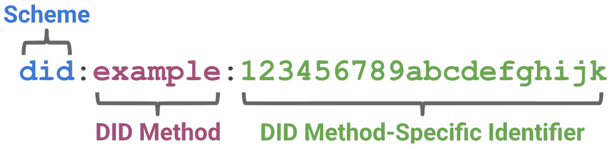
Scheme
did:example:123456789abcdefghijk
DID Method
DID Method-Specific Identifier

Figure 3: A simple example of a decentralized identifier (DID).

As presented in Figure 3, a DID is a simple text string consisting of three parts: 1) the DID URI scheme identifier, 2) the identifier for the DID method, and 3) the DID method-specific identifier. Using a DID method specific resolver, a DID can be resolved and its associated DID document retrieved from a Verifiable Data Registry (VDR), also depicted in Figure 2. A DID document can express cryptographic material, verification methods or services, which provide a set of mechanisms enabling a DID controller to prove control of the DID. Services enable trusted interactions associated with the DID subject.

Each entity can have as many DIDs as necessary to maintain their desired separation of identities, personas, and interactions. The use of these identifiers can be scoped appropriately to different contexts, allowing improved privacy as you no longer require one identifier that can correlate all your interactions. One can have one or multiple public DIDs (stored on a public VDR) and/or private DIDs (for which the DID method resolves to a DID document stored on a private registry). In practice, public Issuers and Verifiers have a public DID, so anyone can verify them and resolve their DID document, and individuals ideally have a unique (private) DID for each relationship they interact with, to avoid correlation. To illustrate, DID_I is the public DID of an Issuer, known to everyone. In contrast, DID_{HI} is the DID generated by the Holder specifically for their interactions with that Issuer. For every Issuer, the Holder generates a different DID_{HI}. For every Verifier, the Holder generates a DID_{HV}. This way, the same Holder is known under a different identifier (i.e. pseudonym) by each Issuer and Verifier.

The DID standard specification does not presuppose any particular technology or cryptography. Each DID method can specify and implement their own choice of technology. The DID standard gives you this much control. In the context of SSI, public DID methods are usually stored on (permissioned or permissionless) public distributed ledgers, reducing the need for dependency on third parties. Private DIDs are stored on private infrastructure (e.g.

digital wallet on a mobile device). A DID has a verification method(s) specified in its DID document, containing public key(s), for which the Holder has the associated private/secret key which they use to prove control of the DID. The public (PK) and secret keys (SK) that each ecosystem role maintains has been reflected in Figure 2. In square brackets, [ ], are each role's own set of keys (associated to their DID(s)). In curly brackets, { }, are the public keys that each role caches of the roles it interacts with, to simplify verification. In this illustration the individual (Holder) has a private, unique DID for each relationship in order to avoid correlation of their interactions (DID_{HI} and DID_{HV}). For clarity, SK_{HI} reads as the Secret Key of Holder (H) specifically towards Issuer (I). PK_{G} reads as the Public Key of Government (G) to anyone.

Verifiable Credentials (VC) [SLC19] We all have a range of credentials in our possession today, many of them stored in our wallet. A driver license, a national identity card, a passport, a vaccination certificate, a health insurance card, a birth certificate, a university degree, etc. Each of these credentials was issued to us by an institution with authority on the topic, after they had successfully verified our identity and that we were an authorised recipient of the credential. Verifiable Credentials provide a mechanism to express these types of credentials digitally, cryptographically secure, privacy respecting and machine-verifiable. The Issuer issues a VC to a Holder and digitally signs this VC with the secret key associated to their DID. A Verifier verifies the presentation of a VC by verifying that the credential's digital signature originates from the DID of the Issuer.

There exist different "flavours" of verifiable credentials [You21; KT20a; KT21; KT20b]. They are all expressed in JSON or JSON-LD format and cryptographically signed by the Issuer (using one of their public DID verification methods). To facilitate interoperability there is a community push to agree on one "flavour" going forward. This looks to be JSON-LD with BBS+ signatures in favour of JSON-ZKP-CL [KT20b; You21; Zun21; Hel20a].

Privacy Decentralised identifiers and verifiable credentials with BBS+ or CL signatures enable privacy enhancing capabilities which physical, paper-based credentials cannot provide. The cryptographic signatures also enable the Holder to generate zero-knowledge proofs (ZKP) towards a Verifier. A ZKP is a cryptographic method to prove to a party that you possess some knowledge without actually revealing the underlying information. Combined, they are able to provide:

- **Selective disclosure:** The ability to reveal only a selected set of attributes of a credential for presentation to a Verifier while hiding the rest. E.g.: share your first and last name of your driver license, but do not disclose other attributes. This protection can substantially address the bundling problem as only those attributes for which transparency is requested are shared, nothing more.
- **Predicate proofs:** A specific type of selective disclosure that involves calculating the presented value based on a predicate (such as "greater than", "less than", "equal to"). E.g.: prove that your age is over a particular number without sharing your date of birth. This is an example of a zero-knowledge proof. This capability is supported for JSON credentials with CL signatures. Current implementations of JSON-LD credentials with BBS+ signatures do not support predicate proofs, but this is actively being worked on. Instead, the Issuer can enclose values in the credential (e.g. "over18") that the Holder can disclose without sharing the attribute containing their date of birth. [You21]
- **Compound proofs:** The ability to combine attributes from multiple credentials into one single proof that is presented to the Verifier. It is also possible to prove that a set of attributes (e.g. name, date of birth) are exactly the same across different credentials (from different Issuers).
- **Non-correlating signatures:** The ability to share a zero-knowledge proof of the Issuer's credential signature, rather than the actual signature. This prevents correlation across multiple Verifiers based on an Issuer's credential signature which could track the Holder's interactions and behaviour.
- **Revocation:** The ability for Issuers to revoke verifiable credentials is indispensable as dispute, compromise, mistake, identity change, hacking and insecurity could make any credential become invalid before its expiration, especially when they are carried around by mobile devices. Different privacy protecting revocation solutions exist, using revocation lists [Hel20b; LS20] or cryptographic accumulators [Har18; CL02; ACN13]. Both are typically stored in the secure, publicly accessible Verifiable Data Registries (see Figure 2) that can be accessed by the Verifier without knowledge of the Issuer for privacy reasons.

Additional Benefits In addition to the privacy enhancing capabilities, there are other noteworthy benefits to self-sovereign identity:

- **Flexibility:** Verifiable credentials can be issued about individuals, but equally about organisations, relationships and things (physical or digital). The same, simple model can be applied to a myriad of use cases and subjects. While verifiable credentials are digital in nature, the format is flexible and it is easy to generate physical, paper-based versions of them (e.g. print of QR code that includes the credential or provides a link to it).
- **Security:** Verifiable credentials provide a very high level of cryptographic security, making them nearly impossible to forge. Cryptographic libraries are open-source and publicly reviewed by industry experts. In

combination with their simplicity and flexibility, verifiable credentials increase overall security as they make it easier for businesses (i.e. Verifiers) to verify credentials and grant authorisations (e.g. access) accordingly.

- **Agency:** Self-sovereign identity provides Holders much more control and agency about their credentials and personal information than previously possible. Holders can also choose to rely on a trusted party to host, secure and backup their personal digital wallet. The key of agency is having the choice.
- **Liability:** Via the selective disclosure capability it is possible to reduce data capture by third parties, which can reduce the risk of data breaches and associated liabilities.
- **Cost:** It is possible for businesses to leverage investments across many business use cases in a cost efficient manner due to the open-source approach of SSI and the flexible nature of the model.

KYC The use of an SSI model does not fundamentally change the end-to-end information flow in context of KYC and AML/CTF compliance. It is still a linear model, passing personal information from Issuer, via Holder (User) and Verifier (Business), to Government (regulator/law enforcement). Compared to the current state the main changes can be found in the interactions and the agency of the Holder. Privacy improvements can be achieved between a Holder and a Verifier, addressing the bundling problem. From a privacy or transparency perspective SSI does not change the interaction between a business (i.e. Verifier) and the Government role. Businesses are still required to capture, store and report certain personal information about their customers.

Frictionless The focus of many products and solutions in the digital world is on delivering an optimal user experience. For many this has translated into an obsession with providing a ”frictionless” experience. It is worth calling out though that some amount of friction can be valuable. While the necessary and acceptable level of friction in a digital context might be lower than in the physical world, it should not be dismissed entirely [Her21]. We must be mindful of the second and third order consequences of pursuing a ”frictionless” approach. Self-sovereign identity greatly improves the user experience for proving claims in a trustworthy manner. When it becomes too easy to share trusted credentials and verify them instantly, we risk the proliferation and normalisation of identification requests and the resulting capture of personal information [Ren21]. While the standards underpinning SSI provide certain privacy protecting capabilities from a technological perspective, we must choose to apply them consciously and implement them appropriately to realise their potential. Ecosystem governance has a major role to play here, defining best practices and providing transparency in their application (or lack thereof). Alternatively, we risk increasing the magnitude of the copy problem and the bundling problem, along with profound societal implications.

Conclusion Self-sovereign identity improves the agency and control an individual has over their personal information via the implementation of the W3C standards of verifiable credentials and decentralised identifiers. Cryptographic primitives and protocols allow for data sharing to be minimised, but to comply with KYC obligations Holders must still share some personal information with regulated businesses, who verify it and (upon request) make it available to regulators. Table 1 provides an overview of the impact of self-sovereign identity on the three problems preventing structured transparency in the current state for KYC.

Problem	Impact of Self-Sovereign Identity
Copy Problem	No material impact in context of KYC as Holders must still share specific identity attributes with regulated businesses to comply with KYC obligations. The personal customer information held by the businesses is fully in their control, sustaining the copy problem. Some positive impact may perhaps be found in the fact that businesses now receive sensitive data in a structured, digital format, making it easier to manage and secure than unstructured data such as physical copies of documents or electronic scans.
Bundling Problem	Positive impact as selective disclosure and non-correlating signatures allow for the Holder only to share those identity attributes required for KYC purposes. If additional personal information must be shared for the business to be able to provide the requested service, then also predicate proofs and compound proofs can help to improve privacy and minimise personal information sharing. This effectively solves the bundling problem.
Recursive Oversight Problem	No impact to the current state.

Table 1: Impact of Self-Sovereign Identity

3.3 zkKYC

Self-sovereign identity provides many benefits and contributes to achieving structured transparency by addressing the bundling problem. To also tackle the copy problem and the recursive oversight problem, we have to go one step further. zkKYC, or zero-knowledge KYC, is a solution concept that aims to do so.

Business Requirements Table 2 provides the high-level business requirements for the zkKYC solution concept to achieve maximum structured transparency in context of KYC. These requirements are not exhaustive or definitive, but provide a good direction for what is expected from the solution concept. The terminology used in the requirements is based on the role names in the current state (see Figure 1) but can be easily exchanged for the role names the self-sovereign identity model (see Figure 2) due to the first business requirement.

ID	Business Requirement
BR01	The level of user control, agency and privacy provided and enabled by the self-sovereign identity model MUST be preserved or enhanced. See section 3.2 for details.
BR02	A User SHOULD NOT share personal identifiable information (e.g. name, address, date of birth) when on-boarding at a Business.
BR03	A User MUST prove they meet the criteria defined by the Business or relevant regulator(s) to consume the provided service (e.g. adult, domestic resident, valid driver license for specific vehicle category, verified email address).
BR04	A Business that suspects a specific User of fraud, money laundering or terrorism financing MUST be able to report that User to Government (e.g. regulator).
BR05	A Business that wants to file charges against a specific User due to breach of contract or other dispute MUST be able to report that User to Government (e.g. law enforcement).
BR06	Government (e.g. regulator, law enforcement) MUST be able to identify a reported User based on the information provided and on the ground of reasonable suspicion.
BR07	When a Business reports a User to Government, this MUST NOT be disclosed to the User (i.e. tipping-off).
BR08	A Business SHOULD NOT hold personal identifiable information on a User, unless it is provided to them by Government in context of a reported issue.

Table 2: Business Requirements

Concept zkKYC extends the self-sovereign identity model (BR01, BR03). The key difference is that Users (i.e. Holders) do no longer have to provide personal identifiable information to each Business (i.e. Verifier) they join (BR02). Rather, based on the verifiable credentials they hold, Holders can provide proof they are eligible and use their digital identity wallet to generate a unique token that represents their identity for that Verifier. Let's call it a zkKYC token. If and when the need arises (e.g. legal charges, regulatory reporting), a Verifier can then present this token to Government (BR04, BR05). Only Government will be able to read the token to identify the originating Issuer(s) of the credential(s) used to generate the token from (BR08). Read access to the token will also reveal a unique Holder identifier that enables the originating Issuer to provide Government with the personal information about the Holder associated to that Holder identifier to pursue their investigation or legal action (BR06). As presented in Figure 4, this setup creates a circular model instead of linear model. The reason why resolving the Holder's identity is assigned to the Government role and not a Verifier (BR08) is to make sure this is a trusted entity with an assigned governance and oversight responsibility (in pursuit of structured transparency) and to preserve the privacy characteristic that Issuers do not learn at which Verifiers Holders present their issued credentials (BR01).

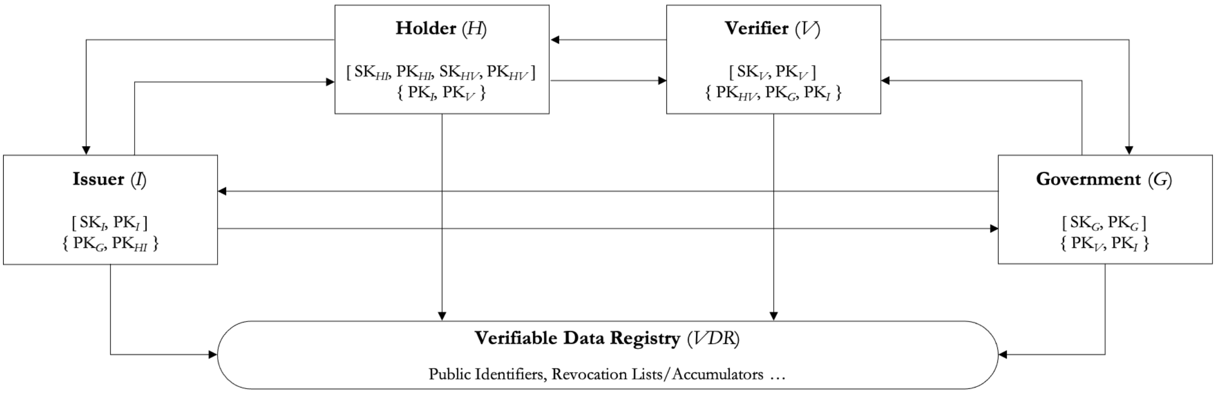
Holder (H)
$[\, SK_{HI}, PK_{HI}, SK_{HV}, PK_{HV} \,]$
$\{\, PK_I, PK_V \,\}$
Verifier (V)
$[\, SK_V, PK_V \,]$
$\{\, PK_{HV}, PK_G, PK_I \,\}$
Issuer (I)
$[\, SK_I, PK_I \,]$
$\{\, PK_G, PK_{HI} \,\}$
Government (G)
$[\, SK_G, PK_G \,]$
$\{\, PK_V, PK_I \,\}$
Verifiable Data Registry (VDR)
Public Identifiers, Revocation Lists/Accumulators …

Figure 4: zkKYC solution concept overview

This approach to KYC achieves two major privacy benefits:

- No Verifier needs to receive, store and manage personal information of its Users anymore in order to comply with KYC obligations.
- A User's personal information is only shared with a third party fulfilling the Government role, if and when there is a legal or regulatory reason to do so, not sooner than that.

Solution Requirements The key element that provides additional privacy to the SSI model is the zkKYC token. This token avoids having to share personal information for KYC purposes with each and every business. The following solution requirements drive the token design and the more detailed information flow between ecosystem roles:

ID	Solution Requirement
SR01	The token MUST be generated by the Holder, using their SSI agent/wallet.
SR02	The token MUST contain an identifier about the Holder that is present in a verifiable credential issued to them by a white-listed, trusted Issuer. Only the credential Issuer and the Holder should know the association of the Holder identifier to the Holder's identity.
SR03	The token MUST be unique and specific to each Verifier so that if stolen (e.g. hack, data breach) it cannot be of value to anyone else.
SR04	The token MUST be readable by Government only (i.e. NOT by a Verifier or Issuer).
SR05	Although the Verifier cannot read the information inside the token, they MUST be able to verify that the token contains the correct information. If Verifiers would not be able to verify this, then disingenuous Holders could include false information in the token, which is exactly what we want to prevent.
SR06	Government MUST be able to verify that the token was generated by the claimed Holder and specifically for the Verifier that provided the token to Government.
SR07	Government MUST be able to identify and contact the Issuer based on information in the token. The Issuer can then reveal the Holder's identity to Government based on information in the token as only they know the identity of the party associated to the Holder identifier.

Table 3: Solution Requirements

The zkKYC solution concept based on these business and solution requirements is detailed below for each of the information flows between the different ecosystem roles.

Issuer ↔ Holder The information flow between Issuer and Holder remains the same as in the self-sovereign identity model. zkKYC does not change this information flow as such.

Holder ↔ Verifier Although the information flow interaction pattern between Holder and Verifier does not fundamentally change with zkKYC, the information exchanged between them is very different. A Verifier under zkKYC will request the Holder to present three types of information:

1. Eligibility proof: proof(s) that the Holder meets the (business) criteria set out by the Verifier to be able to provide access to their service.
2. zkKYC token: encrypted data object that contains information to enable the Holder's identity to be revealed to specific parties only.
3. Validity proof: proofs that the presented zkKYC token contains the correct information, without disclosing what that information is.

Eligibility proof is a subset of what otherwise would be typically requested by the Verifier in a self-sovereign identity model to meet their KYC obligations. With zkKYC this set of data elements does not contain any personal information that is able to (uniquely) identify the Holder. Rather, this set contains proof that the Holder holds specific valid credentials issued by white-listed, trusted authorities that otherwise would be used for KYC purposes and that they meet the criteria required to authorise access to the provided service(s). For example, the eligibility proof can include that the Holder holds a non-revoked driver license credential issued by a local government agency and that it is not expired. This can also prove that the Holder is over 18 years of age and is a local resident. This might be enough information for the Verifier to know in order to provide access to their service, without knowing personal information such as name, address, date of birth. In the Verifier's presentation request to the Holder, they will have to specify what verifiable credential(s) and attributes they accept as input for generating the eligibility proof.

The **zkKYC token** is an encrypted data object (e.g. following the JSON Web Encryption (JWE) standard for JSON Web Tokens (JWT) [Sir16]) generated by the Holder using their SSI agent/wallet based on information elements sourced from verifiable credentials they hold. Taking abstraction of technical implementation details, this token contains information elements encrypted with Government's public key (PK_G) so that only Government can decrypt the token and read its content (SR04). It is the Verifier that must pass relevant Government(s) identifier(s) to the Holder in the presentation request so the Holder knows which public key(s) to use. Multiple Government parties might be relevant to the Verifier and therefore be included in the request: regulator(s), law enforcement etc. In a self-sovereign identity context these identifiers would be in the form of public DID(s) (DID_G) that are published on a verifiable data registry so that the Holder can verify they are associated to a known and valid Government party. The (non-technical) information elements embedded in the token are:

- A list of one or more two-tuples that each include an Issuer identifier (DID_I) and the Holder identifier (DID_{HI}) that this Issuer used in the verifiable credential issued to that Holder (SR02). These verifiable credentials must be the same as those requested for by the Verifier in the eligibility proof.
- One two-tuple that includes the Holder identifier towards the Verifier (DID_{HV}) and the Verifier identifier (DID_V). This information makes the encrypted zkKYC token unique and specific to each Verifier so that it is of no value to others (SR03).

Validity proof contains two zero-knowledge proofs generated by the Holder (SR05). The first zero-knowledge proof proves that the zkKYC token contains the Holder identifiers and Issuer identifiers that are present in the requested verifiable credentials. The power of zero-knowledge proofs is that this can be proven without disclosing to the Verifier what these exact Holder identifiers are. This provides cryptographic certainty to the Verifier that the Holder did not put Holder identifiers in the zkKYC token that does not relate to themselves or the requested verifiable credentials. It ensures the Verifier that they can let Government identify the Holder when the need arises without themselves ever having to know the Holder or having to store personal information about them.

The second zero-knowledge proof in the validity proof proves that the zkKYC token contains the correct Verifier identifier. This is important as we want the zkKYC token to be unique and specific for this Verifier. This removes any incentive for other parties to steal this token from the Verifier and it provides certainty for the Verifier that their identifier is in the zkKYC token.

The combination of eligibility proof, zkKYC token and validity proof enables the Verifier to meet KYC obligations without knowing a Holder's identity.

Verifier ↔ Government zkKYC changes the information flow between Verifier and Government in the sense that no personal information about a Holder is shared anymore between these two roles. The reason for this change is, of course, that the Verifier no longer has this information. Information exchanged between Verifier and Government (can) contain Holder pseudonyms, more specifically Holder identifiers specific for that particular Verifier (i.e. DID_{HV}). Regulators (as a party fulfilling the Government role in this ecosystem) can analyse the reported transactions (incl. identifiers) and via risk pattern matching identify suspicious transactions and associated Holder identifiers [Bir21]. In case Government needs to reveal a Holder's identity as part of an investigation (e.g. suspicious transactions, legal charges ...) then they ask the Verifier for the zkKYC token issued to them by the relevant Holder. Using their private key, Government can decrypt the token and verify that it contains the correct Holder identifier (DID_{HV}) and the correct Verifier identifier (DID_V) (SR06). The token will also reveal one or more Issuer identifiers (DID_I) and associated Holder identifiers (DID_{HI}) related to the Holder they want to know the true identify of (SR07).

To avoid abuse of the decryption power by Government it is possible to implement a threshold cryptosystem on their side for the usage of their private key. This builds in additional security as an authorised set of m out of n persons within Government must then agree to decrypt a zkKYC token.

Government ↔ Issuer The information flow between Government and Issuer is new and is focused on revealing the Holder's identity. Based on the information extracted from the token (i.e. one or more DID_I), Government can contact the identified Issuers using service endpoints specified in the DID Documents of their resolved DIDs. The Issuers are asked to provide personal information about the associated Holder identifiers in the token (i.e. one or more DID_{HI}). Considering the verifiable credentials were issued by these Issuers, they must have successfully verified the identity of these Holders. This verified identity information can now be used by Government to identity the Holder (SR07) and pursue their investigation.

It is important pointing out that the exact circumstances under which an Issuer must share this information to Government should be clearly analysed, specified and communicated. Which parties taking up the Government role can request this information? Under what conditions? For what purpose? What must be logged and published about these information requests? This is typically addressed in the governance framework of the broader identity scheme (see Trust over IP Governance Stack), data sharing regime or even legislation. If we agree in our society that illegal activity should be identified and prosecuted, then this last information flow is critical. It is equally critical we minimize its usage to what is absolutely necessary though. The zkKYC concept is designed to minimise the disclosure, spread and storage of personal information in order to maximise privacy while providing the transparency required to identify those that are under strong suspicion of illegal activity or face legal prosecution. Pushing this zkKYC objective even further, an Issuer could request from Government a cryptographic proof that (AML/CTF) analytical models were run in a trusted executed environment that confirmed certain transactions or the Holder himself to be suspicious. Alternatively, a cryptographic proof could be generated to prove that a Verifier has formally submitted legal charges against the Holder (e.g. in the format of a verifiable credential), justifying their identity being revealed. While this might seem to be stretching the boundaries of what is realistic, it surely is possible with existing technologies.

Example To make the zkKYC concept come to life, this section illustrates it with a simplified example. Two government agencies issue a Holder with a verifiable credential. One issues a driver license credential, the other issues a digital health insurance credential. The driver license credential is issued by `did:example:id123` (DID_I) to `did:example:id234` (DID_{HI}), expires "2025-12-01" and applies to vehicle category "car". The credential also includes the first name, last name, date of birth and address of the Holder. The digital health insurance credential is issued by `did:example:id345` (DID_I) to `did:example:id456` (DID_{HI}) and expires "2023-12-01". It also includes the Holder's first and last name. The Holder controls the private keys of both `did:example:id234` and `did:example:id456`.

For on-boarding at a low risk regulated Verifier `did:example:id567` (DID_V), the Holder generates a new identifier they control, `did:example:id678` (DID_{HV}). The Verifier requests the Holder to generate eligibility proof, a zkKYC token and matching validity proof. They request the eligibility proof to include the Issuer of a driver license, the vehicle category, country of residence, a zero-knowledge proof that the Holder is over the age of 18, a zero-knowledge proof that the credential is not expired and a zero-knowledge proof that it was issued to the Holder and signed by the Issuer stated at the start of this eligibility proof. The eligibility proof should also include the Issuer of the public health insurance credential, a zero-knowledge proof that it is not expired and a zero-knowledge proof that it was issued to the Holder and signed by the Issuer mentioned before. Note that the requested eligibility proof does not include any personal information about the Holder, only proofs that the Holder is eligible. The Verifier is able to verify these proofs as the Issuer DIDs that signed the issued credentials are publicly resolvable. The Verifier can also validate that these credentials have not been revoked by the Issuer without revealing their identity to the Issuer. This assures the Verifier that the Holder has valid identity credentials issued to them by trusted Issuers. The user also generates a zkKYC token that includes the following two-tuples: [`did:example:id123`, `did:example:id234`], [`did:example:id345`, `did:example:id456`] and [`did:example:id678`, `did:example:id567`]. A matching zero-knowledge validity proof is generated to prove to the Verifier that the encrypted zkKYC token includes the correct information.

If the Holder is ever found suspected of money laundering or terrorism financing, the regulator can request their zkKYC token from the Verifier, decrypt it, resolve the two Issuer DIDs and request them to provide the identity information of `did:example:id234` and `did:example:id456` respectively. Each Issuer looks up the received Holder DID, matches it to one of their users and provides the regulator with the requested identity information. The Issuers will not know which Verifier their users interacted with and the Verifier will not know any personal information about their customer. Yet, Government (regulator) is able to fulfill their role and investigate the suspected customer.

Tokenisation For people with knowledge of tokenisation in the credit card payments world [Ram19], the zkKYC concept as explained above might sound familiar. The objective definitely is: improving security and privacy by taking away the need for businesses/merchants to have to process and store sensitive customer information (i.e. credit card details). Instead of sensitive information, a seemingly random string of characters is passed, unique and specific for that transaction and that business. There are however also some differences.

A first major difference is that the credit card model is based on trust in the Payment Network (e.g. VISA, Mastercard) and between the different participants. This translates into trusted, centralised intermediaries (e.g. Token Service Providers, Apple Pay ...) who enable a smooth and secure experience, but also have a high level of control (often translated in high fees and profitability). In a self-sovereign or decentralised identity model, the objective is to minimize the reliance on trusted intermediaries and put the user in more control and strengthen their agency. Considering zkKYC lacks the same trust assumptions as in a credit card scheme, it requires the generation and provision of (zero-knowledge) validity proof along with the data that is being passed. Cryptography replaces some trust assumptions and enables stronger sovereignty of the Holder. A strong trust axis between Verifier and Issuer remains. That is actually the foundation of self-sovereign identity based schemes.

A second major difference between zkKYC and the credit card tokenisation is that zkKYC does not have real-time redemption of the token into the underlying sensitive information. When executing a credit card payment with tokenisation, the Payment Token is redeemed for its underlying information and validated in order to authorise the payment. With zkKYC, the purpose is not to reveal personal information at all, unless at some point in the future that is required for legal or regulatory reasons. Therefore, it is critical that validity proof can demonstrate that the correct information is embedded within the zkKYC token, as no real-time redemption and validation of that information by a trusted party takes place. This delayed redemption of the underlying sensitive information is a second reason why we require cryptographic proof in the present to confirm validity.

Table 4 provides an overview of how some of the concepts in both worlds compare. They are not identical, but there are conceptual similarities.

Credit Card Tokenisation	zkKYC
Issuer Bank	Verifiable Credential Issuer
Token Service Provider	SSI Agent/Wallet
Payment Token	Holder Identifier to the Issuer (DID$_{HI}$)
Apple Pay	SSI Agent/Wallet
Dynamic Cryptogram	zkKYC token
Merchant	Verifier
Payment Network	Government
Trusted intermediaries	Advanced Cryptography

Table 4: Credit Card Tokenisation vs. zkKYC: concepts compared

Conclusion zkKYC builds on top of self-sovereign identity and eliminates the need for personal information to be shared with Verifiers for the purpose of KYC. Criteria that must be met by the Holder can be proven and verified (cryptographically) using verifiable credentials issued by trusted Issuers. A Holder generates a unique zkKYC token specifically for each Verifier, which a Verifier can share with Government if and when required. Government is the only role in this model who can decrypt the token and based on identifiers enclosed in it, can request the relevant Issuer(s) to reveal the Holder's identity. Table 5 provides an overview of how zkKYC addresses the three problems preventing structured transparency in the current state for KYC.

Problem	Impact of zkKYC
Copy Problem	Positive impact to the current state as the need for personal information to be shared with Verifiers for KYC is eliminated. The shared information does not reveal personal identifiable information and the zkKYC token is cryptographically secured to be unique and specific to the Verifier so it is worthless for other parties, discouraging a hack.
Bundling Problem	Positive impact to the current state as zkKYC builds on top of SSI and SSI addresses the bundling problem (see Table 1).
Recursive Oversight Problem	Positive impact to the current state as zkKYC reduces the number of parties that can access and hold personal information to a minimum. While it is impossible to fully remove the recursive oversight problem in this context, it has been addressed substantially and sufficiently.

Table 5: Impact of zkKYC

4 Alternative Applications

The zkKYC concept as presented in this paper is focused on businesses meeting their KYC obligations under AML/CTF regulation. In addition, it could be implemented by non-regulated businesses to perform KYC without capturing personal information, for example to be prepared in case an adversarial (legal) situation arises with their customer. As outlined below, there are other applications of the zkKYC solution concept possible.

Decentralised KYC As zkKYC is built on top of a decentralised (self-sovereign) identity model, it is well suited to be applied for decentralised KYC. The obvious candidate use case for this is Decentralised Finance (DeFi). With no central party in full control of a DeFi service, the set of smart contracts of a Decentralised Autonomous Organisation (DAO) could perform the role of Verifier. An on-chain Verifier smart contract can implement the required SSI verification capabilities considering it has its own DID and can rely on an oracle to source trusted Issuer identifiers. Also the verification of the validity proof of the zkKYC token can be executed by a smart contract. This is already happening today on public blockchains for privacy focused cryptocurrencies and layer 2 scaling technologies (i.e. zkRollup). This enables DeFi DAOs to capture and verify zkKYC tokens when onboarding users and release them to a governance body (i.e. Government) when required. The zkKYC concept could also facilitate undercollateralised or even uncollateralised lending, given the DAO could verify a zero-knowledge proof of a credit score from a trusted Issuer.

Account-to-Account Payments If a bank were to issue its customers a verifiable credential for owning a bank account, then this credential and its associated Holder identifier (DID$_{HI}$) could be used as input to a zkKYC like approach to authenticate themselves (even aligned with Strong Customer Authentication (SCA) per PSD2) and authorise account-to-account based payments at merchants, online or in person. It is similar to the KYC use case, but the main difference is that there is real-time redemption instead of delayed and the "zkKYC token" must also be associated with the transaction at hand. Additionally, the bank plays the role of both Issuer and Government. To avoid confusion with the KYC use case and align terminology with credit card tokenisation, let us rename 'zkKYC token' to 'payment cryptogram'. As a result, the following (non-technical) data is exchanged between Holder (payer) and Verifier (merchant/payee):

- Eligibility proof includes the Issuer DID of the bank that has issued the Holder with a verifiable credential about a bank account they have with the bank. It also includes a zero-knowledge proof that the credential is not expired and a zero-knowledge proof that it was issued to the Holder and signed by the Issuer.
- Payment cryptogram is generated by the Holder using their SSI agent/wallet, encrypted with the credential Issuer's public key and contains:
  - One two-tuple that includes an Issuer identifier (DID$_I$) and the Holder identifier (DID$_{HI}$) that this Issuer used in the bank account verifiable credential issued to that Holder.
  - The Verifier identifier (DID$_V$). This information makes the encrypted cryptogram unique and specific to the Verifier so that it is of no value to others.
  - Transaction number.
  - Transaction amount.
- Validity proof can contain three zero-knowledge proofs generated by the Holder.
  1. Proof that the zkKYC token contains the Holder identifier and Issuer identifier that are present in the bank account verifiable credential referenced in the eligibility proof. This provides certainty to the Verifier that the Holder did not put a Holder identifier in the cryptogram that does not relate to their own bank account credential.
  2. Proof that the cryptogram contains the correct Verifier identifier. This is important as we want the cryptogram to be unique and specific for this Verifier. This discourages other parties to steal this one-time cryptogram from the Verifier.
  3. Proof that the correct transaction amount is included in the cryptogram.

  It must be noted that these (validity) proofs are not strictly necessary considering the real-time redemption context of this use case. The Verifier (merchant) will want real-time authorisation from the Issuer that there is enough funds available for the payment and that the cryptogram is valid. Verifying these proofs can however avoid the merchant or their payment service provider from making an unsuccessful round trip to the Issuer. The alternative is for the validity proof to contain the Holder signature of the payment cryptogram using the private key (SK$_{HI}$) associated to the Holder identifier towards the bank account credential Issuer (DID$_{HI}$). This signature then proves to the Issuer (bank) that the bank account credential Holder (i.e. rightful account owner) generated the cryptogram and wants to authorise the account payment.

Based on the Issuer DID in the eligibility proof, the Verifier's payment service provider can resolve the associated DID Document and retrieve the issuing bank's endpoint for their payment initiation service. They send the Issuer the payment cryptogram along with the validity proof and the Verifier's details (DID$_V$ and beneficiary bank account details). The issuing bank decrypts the cryptogram and validates its content. Is their own Issuer identifier included? Is the included Holder identifier associated to one of their accounts for which they have issued a valid verifiable credential (not expired, not revoked)? Does the identified account have sufficient funds to clear the transaction amount stated in the cryptogram? Is the embedded Verifier identifier (i.e. merchant/payee) the same as the one referenced in the payment request? The Issuer also validates the validity proof to make sure that it was the rightful account owner who initiated this transaction and generated the payment cryptogram. If all these validations are successful, a payment authorisation can be provided to the Verifier (merchant) and the payment process can be initiated. In case of fast payment rails (e.g. NPP in Australia) the payment can be cleared and settled in seconds. The merchant receives the payment in their account near real-time and possibly for lower fees than with card based payment rails. An additional benefit of this approach is that the authentication process of the account owner is very simple, even multi-factor (SCA), and at the time of initiating the transaction. The method of payment selection and user authentication are combined in one step, similar to Apple Pay's approach, but then for account based payments and without a trusted intermediary orchestrating the process. Last, it avoids the Verifier/merchant having to receive, manage and store payment details of their customers.

5 Conclusion

The zkKYC solution concept as presented in this paper provides an approach for KYC that avoids having to share personal information with every (regulated) service provider a user enters into a relationship with. It also ensures that the user can be identified if and when it is legally or regulatory required and possibly even required to be successfully proven. This demonstrates how zkKYC can shift the Pareto frontier and achieve structured transparency for KYC using modern technologies such as advanced cryptography. The privacy of users can be strengthened without compromising the required transparency. As this concept builds on top of a decentralised identity foundation, a network of parties across the different roles of the ecosystem must participate to realise its potential. Further considerations and possible next steps are shared below.

5.1 Further Considerations

Retail The zkKYC concept as presented in this paper focuses on KYC of individuals in a retail context, rather than individuals that represent organisations or organisations themselves. The two main motivations for that are that this paper focuses on protecting the privacy of the individual in a self representing context and that the identity of organisations is a much more complex topic. However, the zkKYC concept could also be applied to organisations or delegates of organisations.

Regulation There are regulatory changes required to enable a wide scale implementation of zkKYC. AML/CTF regulated entities must be able to rely on cryptographic proof of identity verification performed by others, even without access to the personal information, rather than doing their own Customer Due Diligence. Although in many jurisdictions AML/CTF regulation suggests a risk-based interpretation and implementation approach, clarity and explicit approval of this practice would likely facilitate adoption. A recent publication from FATF [FAT20] provides guidance for the use of Digital ID for the purpose of Customer Due Diligence. This provides a helpful direction for regulators to define and implement their own policies. It also serves as a clear reminder that technology alone will not provide the answer. Equally, if not more, important are the assurance frameworks (e.g. identification, authentication and federation assurance levels) and standards (technical, legal, business) that are required for effective governance of any digital identity scheme and service.

Government In the context of AML/CTF Customer Due Diligence, zkKYC assumes that the regulator itself also participates in the (self-sovereign) identity ecosystem, assuming the role of Government. The regulator would have its own identity (associated to DID$_G$) and associated cryptographic keys (PK$_G$ and SK$_G$). This enables Holders to asymmetrically encrypt the zkKYC token so that solely the regulator can read its content when required. Admittedly, this is a significant implementation challenge for the zkKYC solution concept. Therefore, it might be worthwhile to consider which parties could take up the Government role. A suggestion may be to implement the zkKYC concept in non-regulated use cases first, for example where a business wants to protect themselves from any future dispute with their pseudonymous customers. In this case, an authoritative party that performs a governance function can be involved in the digital identity ecosystem to take up the role of Government. An obvious candidate is a formal law enforcement entity, but the role could also be taken up by a type of governing council, one set up to govern the identity scheme or the wider ecosystem. To minimise the required trust in centralised entities, this council could consist of a representative group of ecosystem participants or even be fully decentralised as a DAO. This makes it clear that the Government role can be implemented by one or many parties, in multiple different ways, per the preferences and objectives of the ecosystem design.

KYC-Issuers The concept as presented in this paper focuses on a basic Customer Due Diligence process, suitable for low risk use cases, but regardless addresses the largest part of the KYC volume (and related sharing of personal information) [GMW19]. zkKYC as presented does not cater for Enhanced Customer Due Diligence (ECDD) when the money laundering or terrorism financing risk is high. ECDD can include ongoing and transaction specific checks whether a customer is a foreign politically exposed person (PEP), whether a customer is on a sanctions list or who the beneficial owner of the customer is and what risk they pose. Without knowing who your customer actually is, it is impossible for a regulated entity to perform these enhanced checks. AML/CTF regulated entities today already outsource a large part of the operational and technology stack for KYC activities to specialised providers (e.g. Equifax[1], TrustID[2] and FrankieOne[3]). Taking this evolution one step further into the direction of decentralised identity, one can imagine the role of KYC-Issuers as a special type of Issuer in the zkKYC concept. Based on successful identity verification (using credentials issued by regular Issuers), a KYC-Issuer could perform Enhanced Customer Due Diligence and issue a KYC verifiable credential directly to the customer (Holder), reflecting the level of Customer Due Diligence performed. This KYC credential can then be used in the generic zkKYC concept as presented in this paper. Specific credential attributes can be required to be included in the eligibility proof (e.g. level of Customer Due Diligence) and also the zkKYC token can be generated using attributes of this credential, i.e. DID of the KYC-Issuer (DID_K) and DID of the Holder towards that KYC-Issuer (DID_{HK}).

This model would require regulatory guidance on the reliance of KYC and ECDD performed by third parties. A KYC-Issuer will require a Holder's personal information but with the benefit that this is only one party instead of every regulated entity requiring this data. KYC-Issuers could also be designated parties that require regulatory accreditation, meeting the highest levels of security and privacy practices. There are several organisations in society today that could meet this profile, for example financial institutions.

Zero-Knowledge Zero-Knowledge Proof as applied in the zkKYC validity proof is a type of advanced cryptography, alongside secure Multi-Party Computation (MPC) and Fully Homomorphic Encryption (FHE). While this particular concept exists for several decades, it is with the advent of blockchain technology and privacy focused cryptocurrencies that we have seen a Cambrian Explosion of research and implementations of zero-knowledge protocols [Ben20]. The innovation and speed of progress in this space has been impressive. The improvement of privacy and scalability are the main applications of this technology. There are different implementations and approaches, each with different cryptographic assumptions, performance metrics and proof sizes. A key area for zkKYC to further explore is the performance cost of generating the validity proof. A prototype that focuses on this particular aspect would be valuable to measure the performance cost and assess its impact on the user experience.

Standardisation Standardisation of anonymous credentials (e.g. JSON-LD with BBS+ signatures), predicate based proofs, privacy preserving credential revocation and zero-knowledge protocols for the validity proof of the zkKYC token are required. This will support interoperability and portability across different SSI and zkKYC solutions and products (e.g. wallets and verification services).

5.2 Next Steps

Open discussions with and constructive feedback from subject matter experts on topics including AML/CTF regulation, privacy and (zero-knowledge) cryptography are highly welcome to evaluate and validate the elements underpinning the zkKYC solution concept. Based on this feedback the concept could be further refined, a more detailed design developed and possibly a prototype built. Such prototype could validate the optimal zero-knowledge proof technology to cater for a performance cost of generating the validity proof that is acceptable from a user experience perspective.

Last, feedback from regulators and businesses in different jurisdictions would be valuable the assess the desirability of the zkKYC concept and the outcomes it aims to achieve.

[1] https://www.equifax.com.au/business-enterprise/products/idmatrix

[2] https://www.trustid.co.uk/

[3] https://www.frankieone.com/

Acknowledgements

I want to acknowledge the open and inclusive communities of self-sovereign identity and zero-knowledge cryptography. I deeply respect and I am thankful for the individuals that drive and lead these communities, create platforms for respectful but rigorous discussion and interaction, generate valuable podcast material and write insightful articles and papers. They are driven by a greater cause to simply make things better, raise the bar and break seemingly obvious trade-offs.

I also want to express gratitude for the thorough review and generous feedback from Daira Hopwood, Robert Porter, Sagit Shiran and Jo Spencer. Their domain expertise and insightful comments were invaluable. Thank you.

References

[PwC17] PwC. *Don't get blindsided by new regulations. What 'tranche two' means for your business.* June 2017. URL: https://www.pwc.com.au/publications/assets/tranche-two-blindsided-new-regulations-jun2017.pdf.

[GMW19] Mette Gade, Daniel Mikkelsen, and Dan Williams. *Making your KYC remediation efforts risk and value-based.* McKinsey & Company. Aug. 23, 2019. URL: https://www.mckinsey.com/industries/financial-services/our-insights/banking-matters/making-your-kyc-remediation-efforts-risk-and-value-based.

[Tra+20] Andrew Trask et al. *Beyond Privacy Trade-offs with Structured Transparency.* report. University of Oxford, OpenMined, Dec. 2020. arXiv: 2012.08347v1 [cs.DS].

[Max21] Nick Maxwell. *Case studies of the use of privacy preserving analysis to tackle financial crime.* report. Version 1.3. Future of Financial Intelligence Sharing (FFIS), Jan. 8, 2021. URL: https://www.future-fis.com/the-pet-project.html.

[Lyn20] Nathan Lynch. *Australia unveils world-first "privacy preserving" fintel encryption project.* July 23, 2020. URL: https://www.linkedin.com/pulse/australia-unveils-world-first-privacy-preserving-fintel-nathan-lynch/.

[GGF17] Paul A. Grassi, Michael E. Garcia, and James L. Fenton. *Digital Identity Guidelines. NIST Special Publication 800-63.* Revision 3. National Institute of Standards and Technology (NIST). June 2017. DOI: 10.6028/NIST.SP.800-63-3. URL: https://pages.nist.gov/800-63-3/.

[Hel21] Nader Helmy. *The State of Identity on the Web.* MATTR Global. Mar. 15, 2021. URL: https://medium.com/mattr-global/the-state-of-identity-on-the-web-cffc392bc7ea.

[All16] Christopher Allen. *The Path to Self-Sovereign Identity.* Apr. 25, 2016. URL: https://www.lifewithalacrity.com/2016/04/the-path-to-self-soverereign-identity.html.

[Ree+21] Drummond Reed et al. *Decentralized Identifiers (DIDs) v1.0. Core architecture, data model, and representations.* W3C. Mar. 18, 2021. URL: https://www.w3.org/TR/did-core/.

[SLC19] Manu Sporny, Dave Longley, and David Chadwick. *Verifiable Credentials Data Model 1.0. Expressing verifiable information on the Web.* W3C. Nov. 19, 2019. URL: https://www.w3.org/TR/vc-data-model/.

[You21] Kaliya "Identity Woman" Young. *Verifiable Credentials Flavors Explained.* COVID-19 Credentials Initiative (CCI). Feb. 11, 2021. URL: https://www.lfph.io/2021/02/11/cci-verifiable-credentials-flavors-and-interoperability-paper/.

[KT20a] Dr. Nuttawut Kongsuwan and Rachata Tosirisuk. *Anonymous Credential Part 1: Brief Overview and History.* Oct. 1, 2020. URL: https://medium.com/finema/anonymous-credential-part-1-brief-overview-and-history-c6679034c914.

[KT21] Dr. Nuttawut Kongsuwan and Rachata Tosirisuk. *Anonymous Credential Part 2: Selective Disclosure and CL Signature.* Feb. 4, 2021. URL: https://medium.com/finema/anonymous-credential-part-2-selective-disclosure-and-cl-signature-b904a93a1565.

[KT20b] Dr. Nuttawut Kongsuwan and Rachata Tosirisuk. *Anonymous Credential Part 3: BBS+ Signature.* Oct. 28, 2020. URL: https://medium.com/finema/anonymous-credential-part-3-bbs-signature-26797721ca74.

[Zun21] Brent Zundel. *Why the Verifiable Credentials Community Should Converge on BBS+.* Evernym. Mar. 24, 2021. URL: https://www.evernym.com/blog/bbs-verifiable-credentials/.

[Hel20a] Nader Helmy. *A solution for privacy-preserving Verifiable Credentials*. MATTR Global. May 8, 2020. URL: https://medium.com/mattr-global/a-solution-for-privacy-preserving-verifiable-credentials-f1650aa16093.

[Hel20b] Nader Helmy. *Adding support for revocation of Verifiable Credentials*. MATTR Global. Oct. 21, 2020. URL: https://medium.com/mattr-global/adding-support-for-revocation-of-verifiable-credentials-2342b66b0997.

[LS20] Dave Longley and Manu Sporny. *Revocation List 2020. A privacy-preserving mechanism for revoking Verifiable Credentials*. W3C. Dec. 29, 2020. URL: https://w3c-ccg.github.io/vc-status-rl-2020/.

[Har18] Daniel Hardman. *Indy HIPE 0011: Credential Revocation*. Hyperledger Indy. Feb. 1, 2018. URL: https://github.com/hyperledger/indy-hipe/blob/master/text/0011-cred-revocation/README.md.

[CL02] Jan Camenisch and Anna Lysyanskaya. *Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials*. report. IBM Research Zurich Research Laboratory, MIT LCS, Feb. 2002. URL: https://cs.brown.edu/people/alysyans/papers/camlys02.pdf.

[ACN13] Tolga Acar, Sherman S.M. Chow, and Lan Nguyen. *Accumulators and U-Prove Revocation*. report. Intel Corporation, Microsoft Research, Department of Information EngineeringChinese University of Hong Kong, June 5, 2013. URL: https://ifca.ai/pub/fc13/78590185.pdf.

[Her21] Matt Herlihy. *In Praise of Friction*. Feb. 26, 2021. URL: https://medium.com/better-by/in-praise-of-friction-71a6893e0138.

[Ren21] Elizabeth M. Renieris. *What's Really at Stake with Vaccine Passports*. Apr. 5, 2021. URL: https://www.cigionline.org/articles/whats-really-stake-vaccine-passports.

[Sir16] Prabath Siriwardena. *JWT, JWS and JWE for Not So Dummies! (Part I)*. Apr. 27, 2016. URL: https://medium.facilelogin.com/jwt-jws-and-jwe-for-not-so-dummies-b63310d201a3.

[Bir21] David G.W. Birch. *The Case Against The Anti-Money Laundering Rules*. May 3, 2021. URL: https://www.forbes.com/sites/davidbirch/2021/05/03/im-anti-the-anti-money-laundering--rules/.

[Ram19] Prashant Ram. *How Apple Pay works under the hood?* Nov. 6, 2019. URL: https://codeburst.io/how-does-apple-pay-actually-work-f52f7d9348b7.

[FAT20] FATF. *Guidance on Digital Identity*. Paris, Mar. 2020. URL: https://www.fatf-gafi.org/publications/fatfrecommendations/documents/digital-identity-guidance.html.

[Ben20] Eli Ben-Sasson. *A Cambrian Explosion of Crypto Proofs*. StarkWare. Jan. 8, 2020. URL: https://nakamoto.com/cambrian-explosion-of-crypto-proofs/.