ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/346485525

Robust KYC via Distributed Ledger Technology

Conference Paper · November 2020

DOI: 10.6084/m9.figshare.13301363

CITATIONS
0

READS
429

3 authors:

Matus Drgon
The University of Edinburgh
1 PUBLICATION **0** CITATIONS
SEE PROFILE

Lamprini Georgiou
The University of Edinburgh
2 PUBLICATIONS **0** CITATIONS
SEE PROFILE

Aggelos Kiayias
University of Connecticut
160 PUBLICATIONS **4,049** CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Blockchain View project

RegTech in the fourth industrial revolution View project

All content following this page was uploaded by Matus Drgon on 30 November 2020.

The user has requested enhancement of the downloaded file.

Robust KYC via Distributed Ledger Technology

Matúš Drgoň[1], Lamprini Georgiou[2], and Aggelos Kiayias[3]

[1] *University of Edinburgh, UK, matus.drgon@gmail.com*

[2] *Blockchain Technology Laboratory, University of Edinburgh, UK, lamprini.georgiou@ed.ac.uk*

[3] *University of Edinburgh and IOHK, UK, aggelos.kiayias@ed.ac.uk*

June 15, 2020

Abstract

Know-Your-Customer (KYC) and Customer due diligence (CDD) are both costly processes that financial institutions are legally required to undertake to conduct business with their customers. Distributed Ledger Technology (DLT) has been recently proposed as a potential coordination mechanism for financial institutions to share KYC costs in a common jurisdiction. While the potential benefits from a cost perspective are enticing, there still exist significant downsides from a security and deployment point of view. In this work we tackle these challenges by providing a novel risk limiting smart contract mechanism that facilitates a trade-off between the security of the KYC process and its cost-efficiency. Using our mechanism, the KYC process becomes more robust as it is possible to fine tune its risk limiting ability and thus repair the "brittleness" of previous DLT-based solutions, a shortcoming we identify. In addition to a theoretical security analysis establishing the robustness of our proposal we also provide an implementation as a smart contract in Solidity and its analysis.

1 Introduction

Know Your Customer (KYC) is a process that financial services firms need to execute to verify the identity of a new customer before they start conducting business with them. The process is obligatorily executed, subject to local laws and is quite costly due to its complexity. Any institution which executes this process needs to authenticate the identity of the customer, verify that the source of the customer's funds is legitimate and assess potential money laundering risks associated with the customer [11]. The legal framework may also evolve over time, requiring updates in the KYC process while institutions must maintain their records up to date and current.

Executing KYC adequately and compliant to a specific legislative framework can be of paramount importance as a failure to execute KYC properly may lead to on-boarding a customer who uses financial services for illicit activities. These include money laundering and terrorism financing among others, whereas the financial institution may face grave legal consequences if it facilitates them. These consequences also are translated to significant fines that the institution has to pay. Such incidents not only affect the annual budget of the financial institution but also their brand and reputation in the market.

The KYC process is supposed to mitigate the risk for the aforementioned possibility of money laundering but in the case of terrorism financing. This implies the process has to adhere to strict regulations. The traditional approach to deal with stricter regulations and increased complexity of the process was the increase in staffing. However, "significant KYC costs have made this operating model no longer sustainable" [22].

According to the Thompson Reuters 2017 Global KYC survey, large financial institutions annually spend $150M and employ several hundreds of employees to conduct KYC. Despite this, the average wait time takes 26 days on expectation to complete. In general, customer dissatisfaction with the process is very high and engaging with more institutions (ranging from 5 to 11 according to various different sources on average per customer) means that customers have to repeat the process losing time, while institutions typically repeat a costly

validation procedure that has already been performed for the same customer multiple times.

The proliferation of financial technology services (Fintechs) during the last decade exacerbates the above issues further. The range of service providers that a customer may choose for managing and obtaining financial services extends well beyond traditional banks and investment firms to technology companies, who, taking advantage of peer-to-peer technology, social media, smartphones and other recent advancements in information technology, deliver products with significantly extended functionality and capabilities.

Motivated by the above, a handful of recent works and projects (see below Section 2, Related Work) considered the use of distributed ledger technology (DLT), [15, 27, 2], as a means to help financial services providers to coordinate and provide KYC in a cost-effective manner.

DLT enables the deployment of a shared database that can only be updated in specific ways that are pre-agreed while preventing the manipulation of the data it contains by any of the participating entities. In this manner, a group of parties may create a shared resource that has high integrity and keep track of relevant information about a certain collective task, as well as metadata associated with the actions of participants in relation to that task.

In the context of KYC the use of DLT can potentially offer a number of benefits:

1. Reduced costs for customer compliance, as institutions can share KYC documents and distribute the cost of the KYC process among them.

2. Increased effectiveness in fight against money laundering and terrorism financing, as sharing information between institutions can promote the flagging of suspicious activity.

3. Enhanced customer satisfaction, as by sharing KYC documents as well as the results of the process between the institutions, the average waiting time for a customer would decrease.

Parra Moyano and Ross [19] put forth the concept of DLT-based KYC systems (in short

DLT-KYC) and identified the following desiderata.

- Proportionality - the ability of the system to distribute the costs between all financial institutions equally. The cost of executing the KYC process should not bring any financial institution an advantage or a disadvantage.
- Irrelevance - it should not be relevant which institution executes the core KYC process. A financial institution should not have an incentive to either execute the core KYC process itself, or not execute KYC and let another institution do so.
- Privacy - a financial institution operating with a customer should not know other institutions operating with the customer.
- No-minting - ensures that a financial institution cannot simulate having executed the core KYC verification without actually doing so.

Our Results. While the above provide a sensible set of requirements for DLT-KYC, there is an important aspect that has remained unaddressed. Distributing the workload of the KYC process across a wide range of DLT participants creates the potential of subverting the system by focusing the resources in the weakest participant. In particular, if someone wants to evade the system they need to identify the weakest participant and subvert the KYC process once; subsequently, they may leverage this level of access to create multiple accounts. We refer to this issue as the potential "brittleness" of DLT-KYC solutions.

We put forth a design and an implementation for DLT-KYC that addresses the problem of brittleness identified above. The approach we follow leverages the ability of DLT systems to generate unpredictable randomness that can be used to perform a risk-limiting validation of the KYC process by repeating it. The resulting "robust" DLT-KYC offers a tunable parameter that enables a natural security/cost-efficiency tradeoff. The higher the repeating probability the higher assurance the system offers. We also present a smart-contract implementation of our robust DLT-KYC system in the Solidity programming languages that was

originally developed in the context of the Ethereum blockchain but now can also be used in the setting of a permissioned ledger. We give a detailed analysis of its properties as well as its security/performance tradeoff. Specifically, we argue how our solution satisfies the properties of proportionality, privacy, irrelevance and no-minting, while at the same time it is robust as it incorporates a risk-limiting KYC validation process tunable by a smart contract parameter.

2 Related work

Britton [3] outlines that an efficient solution for KYC would allow each financial institution to access customer data that would be stored in a distributed database - a solution based on the distributed ledger technology. It is identified that one advantage of using the blockchain, is the creation and management of a digital identity for each customer. The digital identity would be associated to each transaction connected to this customer and would store all relevant information about the customer. These information could be used during AML/transaction monitoring and in this way increase the efficiency of AML checks. It is further outlined that a DLT-based solution would enhance customer experience, reduce operational costs for the financial institutions, increase security and transparency for regulators.

The Hong Kong Monetary Authority[1] researched the use case of KYC for DLT and identified that it could improve customer experience and cut down operational costs incurred by the process for financial institutions due to avoiding repetitive tasks. This refers to the KYC checks that currently have to be executed by each financial institution separately. Their whitepaper [1] further outlines the following challenges of such an approach: cyber security, legal and regulatory compliance. The DLT-based solution would have to be executed in cooperation with regulators to comply with the appropriate legal framework and furthermore ensure customer privacy.

[1] https://www.hkma.gov.hk/eng/

Pachaiyappan et al., [17], put forth a smart-contract implementation of the KYC process on Ethereum blockchain. It very well describe what the KYC process consists of and outline the possible use of Corda blockchain by R3[2] instead. However, the contract length is rather short, it lacks sufficient analysis of how the contract ensures customer privacy, distribution of the cost, how it prevents possible contract manipulation by financial institutions or third parties and how the financial institutions should interact with it.

Sinha et al. [25] propose a system architecture with a sample smart contract utilizing IPFS[3] on Ethereum blockchain. A fully decentralized KYC system on the public Ethereum network is suggested and a gas cost analysis of executing and interacting with the smart contract is also provided. While this implementation can be used as a base for further work, it lacks as an analysis of its basic properties and suitability for this process.

Valkanov et al. [26] identified the potential of KYC systems that leverage DLT in terms of reporting and monitoring transactions. Creating a cross institutional profile for each customer on the blockchain can help the regulator and financial institutions to keep better track of customers' behavior and, given the immutability and ordering offered by DLT, may prove to be effective in detecting activities related to money laundering and financing of terrorism. Refinitiv study on a blockchain KYC solution [22] identifies that a unique ID created for a client as part of the identification and verification process, could connect a greater number of sources of information relating to an individual client, to build a richer picture of their behavior and potentially uncover hidden risks. Both studies discuss the potential of the technology on a broader level, but do not dig deeper into specifics of how this could be achieved.

Parra-Moyano and Ross [19] offer the most comprehensive treatment of the subject so far. They proposed a design for how the KYC process can be executed using the DLT and outline a more centralised, as well as a fully decentralized solution. They provide an architecture of the system, define the basic properties of proportionality, irrelevance, privacy

[2] `https://www.r3.com/`
[3] `https://ipfs.io/`

and no-minting we quoted above, but do not provide a specific implementation.

Parra-Moyano et al. [20] propose an improved version of [19] by providing a smart-contract implementation in Solidity for the Ethereum blockchain. They also outline the following aspects that were improved in comparison to [20]. The first aspect outlines the need of a trusted third party (TTP) that has to periodically check that the financial institutions have paid their proportion for executing the core KYC to keep the system fair. The second aspect identifies the possible need to change the status of the customer in a decentralized manner. The third aspect identifies the need to update the KYC process over time, due to possible changes in the legal and regulatory framework.

3 Regulatory and Procedural background

The legal framework for KYC processes is closely tied to regulations regarding Anti-Money Laundering (AML) and Counter Financing of Terrorism (CFT). The origins of KYC in the US date back to the Patriot Act 2001 that was the first to introduce a legal framework for KYC procedures. The Patriot Act 2001, which was passed after the 9/11 attacks, was influenced by previous legal acts, perhaps the most notable ones being Bank Secrecy Act of 1970 and Money Laundering Control Act of 1986. The Bank Secrecy Act of 1970 was designed to help identify the source, volume, and movement of currency and other monetary instruments transported or transmitted into or out of the United States or deposited in financial institutions [8, 7]. Furthermore, it required banks to report cash transactions over $10,000 and properly identify people conducting such transactions. The Money Laundering Control Act of 1986 was the first regulation that introduced money laundering as a federal crime and was the building block for further legislation in this field.

In more detail, Title III of the Patriot Act specifies two KYC requirements: the Customer Identification Program (CIP) and Customer Due Diligence (CDD). The CIP process identifies the customer and verifies whether they are a real person. The customer might be asked

to provide documents such as passport, national identification card or driver's licence. The CDD process is more complicated as its objective is to estimate what type of transactions the customer will be conducting and what level of risk they may pose to the institution. This is important as it renders it easier for the institution in the future to track suspicious transactions as well as dedicate a considerate level of monitoring on the customer depending on the level of risk the customer possesses. A risk customer might be asked for additional information, such as the purpose of opening an account, financial statements, banking references, description of business operations etc. [28]. For example, it is recommended that a financial institution before starts operating with a politically exposed person should get clearance by the senior management of the institution [4].

The aforementioned KYC and subsequent KYC regulations in this field all have in common that they require proper verification of a customer before a financial institution can open an account for this customer and start conducting business with them. By performing thorough verification of a customer's actions, the risk associated with illicit activities, such as money laundering and financing of terrorism, decreases.

While KYC regulations have attracted most attention in the US, there are international regulations related to money laundering and KYC, such as Anti-Money Laundering and Countering Financing of Terrorism Act 2009 [13] and international recommendations for good KYC practices [5].

In KYC every regulator imposes the legal rules of the jurisdiction where the KYC process is performed. Across jurisdictions, there is no uniformity in KYC regulation and the fact that significant numbers of financial transactions are cross-border, simultaneously belonging to multiple jurisdictions, exacerbates the complexity of KYC compliance. We note that central regulatory motivation for KYC is enforcing anti money laundering (AML) and combating the financing of terrorism (CFT).

In the European Union, the fifth (5th) AML Directive - EU 2018/843- came into force on January 2020 and represents the European approach in KYC regulation. The new Directive

enhances the previous fourth (4th) Money Laundering Directive ((EU) 2015/849) focusing on the following points.

1. It enhances the limits of transparency by setting up publicly available registers for companies, trusts and other legal entities.

2. It increases the powers of EU Financial Intelligence Units and provides them with access to broad information in order to fulfil their tasks.

3. It limits the anonymity related to virtual currencies, wallet providers but also for pre-paid cards.

4. It broadens the criteria for the assessment of high-risk third countries and improves the safeguards for financial transactions to and from such countries.

5. It sets up central bank account registries or retrieval systems in all Member States.

6. It improves the cooperation related to the exchange of information between anti-money laundering supervisors but also between prudential supervisors and the European Central Bank.

The Basel committee on banking supervision ensures that every bank and financial institution should have three lines of defence regarding AML . The first line of defence includes AML/CTF policies and procedures which concern all member staff. Their content is a basic training on how to handle customers and to follow the instructions given ensuring ethical guidelines. As part of the second line of defence, the chief officer in charge of AML/CFT should have the responsibility for ongoing monitoring of the fulfilment of all AML/CFT duties by the bank. This implies sample testing of compliance and review of exceptional reports to alert senior management or the board of directors if it is believed management is failing to address AML/CFT procedures in a responsible manner. The third line of defence involves internal audits: A bank should establish policies for conducting audits of (i) the adequacy of the bank's AML/CFT policies and procedures in addressing identified risks,

(ii) the effectiveness of bank staff in implementing the bank's policies and procedures; (iii) the effectiveness of compliance oversight and quality control including parameters of criteria for automatic alerts; and (iv) the effectiveness of the bank's training of relevant personnel. Moreover, external auditors also have an important role to play in evaluating banks' internal controls and procedures in the course of their financial audits, and in confirming that they are compliant with AML/CFT regulations and supervisory practice.

The necessity for all regulations is there to mitigate potential financial institutions that would facilitate clients who use financial services for illicit activities. The Basel Committee on Banking Supervision [4] identified that "inadequacy or absence of KYC standards can subject banks to serious customer and counter-party risks." Among others, the Committee identified the following possible risks associated with KYC failures..

- Reputational - a significant threat to financial institutions, as the nature of this industry requires strong level of trust from customers' side in the institution. This risk is defined as "the potential that adverse publicity regarding a bank's business practices and associations, whether accurate or not, will cause a loss of confidence in the integrity of the institution."

- Operational - defined as "the risk of direct or indirect loss resulting from inadequate or failed internal processes, people and system or from external events." This risk relates to improper verification of the customer and inadequate execution of customer due diligence. If it is perceived by the public that a financial institution fails at executing its internal processes accurately, this might represent further disrupt on the institution's business.

- Legal - is defined as "the possibility that lawsuits, adverse judgements or contracts that turn out to be unenforceable can disrupt or adversely affect the operations or condition of a bank." It can be perceived as a direct consequence of operational risk, more specifically improper execution of customer verification and due diligence. As a

consequence, financial institutions can get into lawsuits and face expensive fines. For instance, in 2019, the Financial Conduct Authority fined Standard Chartered Bank £102.2 million for AML breaches in areas of its business [6]. In a global scale, fines related to poor money laundering controls issued in 2019 reached a total of $8.14 billion [9]. Over the past 17 years, AML fines averaged a sum of $155m and median of $2.8m [18]. According to Wayne Johnson, the CEO of Encompass, a company providing KYC solutions: "Since 2015, annual AML penalty figures have been steadily rising each year. Multi-million dollar fines have been commonplace for a while, but we are now seeing more penalties of one billion dollars or over, with two in 2019 alone." While it is hard to analyse the exact causes and circumstances that lead to lawsuits and fines, KYC process is a significant means that, when executed thoroughly, mitigates the factors that would escalate to scenarios with legal consequences.

4 KYC processes

This section provides a comparison between a standard KYC process and one based on the distributed ledger technology. To put things into perspective, we start with the "current KYC" process (Section 4.1), which does not use DLT. Then we describe the "Basic DLT-KYC system" which has been suggested in prior work, specifically, [19] and [20], (section 4.2). We then proceed to identify a security flaw – what we term brittleness – that the basic DLT-KYC system can be susceptible (Section 4.3). We present our improved DLT-KYC system in Section 4.4, which we call *Robust DLT-KYC*, that addresses this security flaw.

4.1 Current KYC

The KYC process begins when a customer wants to start interacting with a financial institution. The customer sends personal documents to the financial institution which verifies them. We can think of the KYC process as consisting of two parts - Customer Identifica-

tion Program (CIP) and Customer Due Dilligence (CDD). The documents required for the CIP part could include a passport, an ID card, or some other official state document. This uniquely identifies the customer and provides basic information about the customer - such as their name, date of birth, nationality, address etc. In order to execute CDD, additional documents about the customer - such as previous bank statements, references etc. are required. The documents required from the customer and the depth of the CDD process would depend on the type of the customer and the level of risk this customer represents.

After reviewing the relevant documents sent by the customer, the financial institution then decides to either accept or decline the customer. Upon being accepted, the customer and the financial institution may start conducting business with each other.

The KYC process is separated for the institution that executes this process and its depth and complexity might vary across different financial firms. Parra-Moyano and Ross [19] introduce an abstract part of the process they name the *core* KYC process. This represents the minimum required verification of the customer as specified by a legal framework that a financial firm executing this process adheres to. All financial institutions that conform to this legislative framework need to execute this minimal verification of the customer and it is thus shared by all of them. We can split the KYC process into its *core* part and additional controls of the customer - these remain exclusive to the financial firm executing the KYC and will not be included in our solution.

The KYC process (in its entirety) is currently executed for each customer individually by each financial institution this customer wants to operate with. This means that if a customer conducts business with several financial institutions, the KYC process must be executed by each of these financial institutions, from scratch, separately. If the average cost of this process for a customer is c, and the number of financial institutions this customer is operating with is n, then the overall cost of these multiple KYC processes would be equal to $c * n$.

Note that in the following sections we will use the *core* KYC process and just KYC

process interchangeably for brevity. When we refer to the KYC process in its entirety, this will be explicitly specified.

4.2 Basic DLT-KYC

The underlying idea in a KYC solution that would leverage DLT, first specified by [19], is that the core KYC process, which is the same for each financial institution operating with a customer with respect to a legislative framework, could be executed only once - by the first institution operating with this customer - and the result of this process would be shared by each institution that would like to operate with this customer in the future. When we now define c to be the average cost of executing the core KYC for a customer, the overall cost associated with the KYC process would be capped at c and would be independent of how many financial institutions operate with the customer. For a given financial institution operating with a customer, this would decrease the costs of KYC for the specific customer from c to c/n on average, where n is the number of all financial institutions this customer operates with.

When a customer approaches the first financial institution (`Bank A`), they need to provide the institution with necessary documents needed for executing the core KYC process. We refer to the set of documents the customer supplies as the document package of the customer. There are now two possible scenarios - either this is the first customer `Bank A` wants to use this Basic DLT-KYC solution for, or `Bank A` already operates with at least one other customer using this solution.

In the former case, `Bank A` does not yet operate on the distributed private ledger, governed by the central authority (CA), and needs a permission to join it. The CA is in charge of giving a financial institution this permission and it is assumed that the CA would assess the financial institution before granting it. This is to allay FIs that might try to scam the system.

In the latter case, `Bank A` already has access to this private ledger and uses the Basic DLT-KYC solution for some other customers. The institution needs an account to operate with a customer using the ledger. It is recommended that the institution opens a new account for each customer it operates with. The account the financial institution uses to operate with the customer, does not reveal any information about the institution, and only the CA knows the identity of a financial institution behind the account it uses on the ledger. In other words, `Bank A` opens a new account on the distributed private ledger to on-board this customer, but this account would not reveal the identity of `Bank A`. Only the CA can identify that `Bank A` stands behind this account and would store this information in its private database, not accessible by financial institutions.

`Bank A` stores the customer's document package in its local database and executes the core KYC. When the institution decides to on-board the customer, it creates a simple customer profile on the blockchain. This profile serves as a digital identity of the customer, is specified by a unique ID and stores the result of the core KYC process. The institution gives the customer their ID and if the institution deems the customer acceptable, the two entities can start conducting business together. The digital identity of the customer does not reveal any personal details about the customer. It specifies the number of institutions the customer is operating with and provides a hash of the customer's document package.

Note that the institution only had to execute the *core* KYC process. We mentioned in 4.1 that each institution might want to execute the KYC process differently, with a various level of complexity. The core process that this solution simulates is the basic legal requirement on the customer verification and due diligence. The institution might want to require additional documents and background checks on the customer and store these in its local database. This is decided by the institution and is executable without any interaction with the blockchain.

When the customer wants to operate with another financial institution, `Bank X`, the core KYC process does not need to be repeated because it was executed by `Bank A`. The customer provides `Bank X` with ID that identifies their customer profile. In case the customer forgets

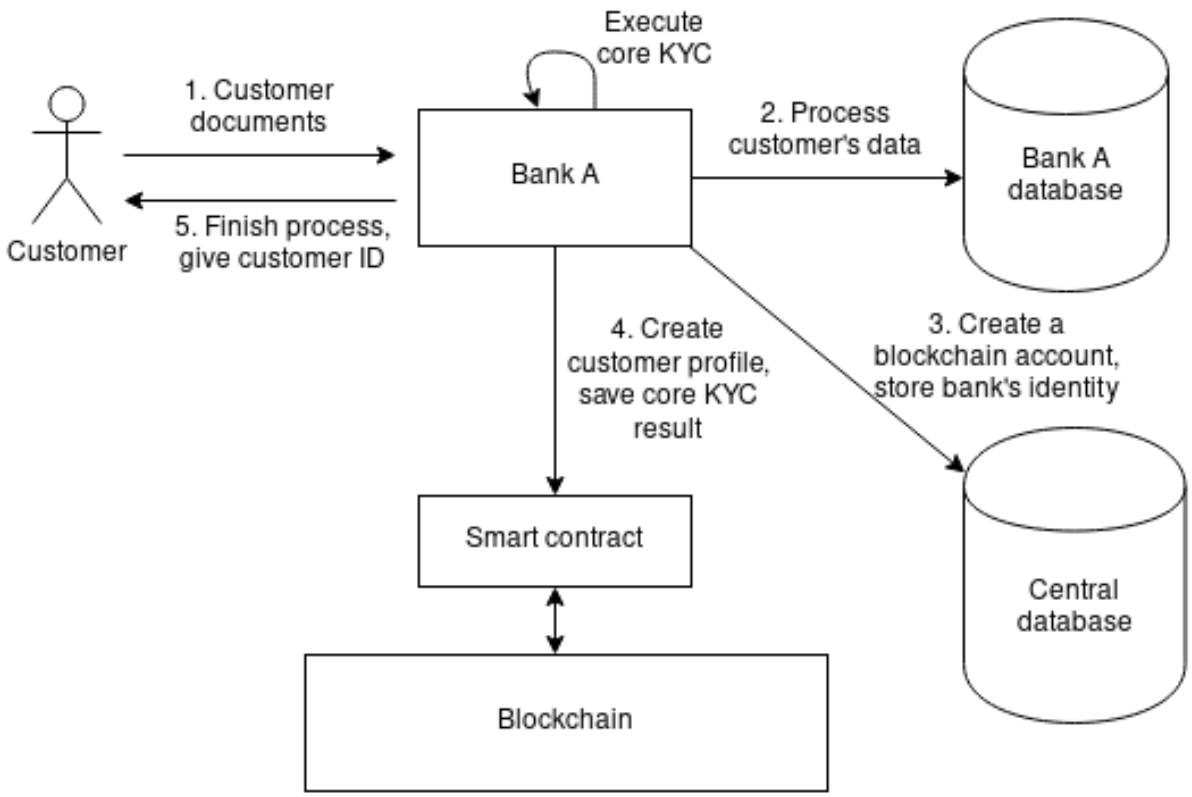

Figure 1: Customer approaches the first financial institution. The central database and blockchain are in control of a trusted third party, such as the regulator.

their ID, they can always retrieve it from any financial institution they have been conducting business with, in this case `Bank A`. Assuming that `Bank X` already has access to the ledger, it can retrieve from the ledger how many institutions operate with this customer and the fee it needs to pay to on-board this customer. After paying the appropriate fee, `Bank X` added to the list of institutions the customer is operating with. `Bank X` and the customer can start conducting business together, effectively speeding up the whole KYC verification process. The fee keeps the system fair, such that the cost of executing the core KYC is equally distributed between all financial institutions that operate with the customer at any instance of time. The figure below outlines this process.

It is expected that the regulatory framework can develop and change over time. When a new legislative requirement is introduced into the process, the core KYC process for the

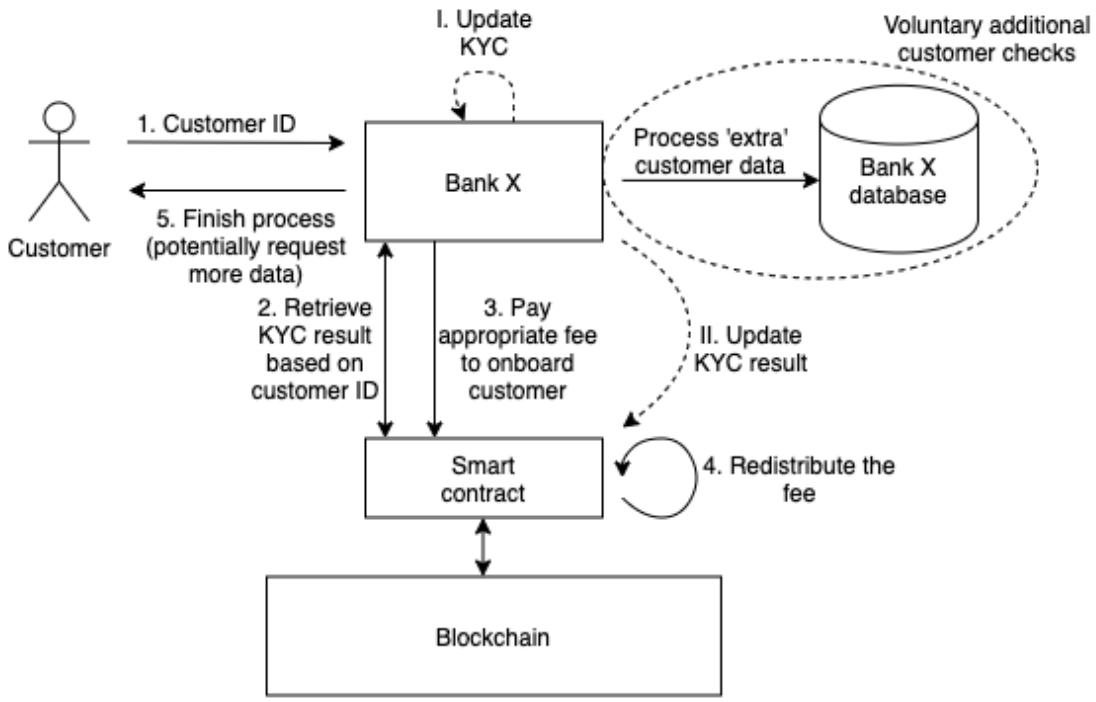

Figure 2: *Basic* DLT-KYC. Customer approaches a new financial institution. The dashed lines and roman numerals indicate a process that might have to be executed only when a legislative change was introduced in the core KYC.

customer has to be updated. If this happens in our simulation before `Bank X` and the customer start operating, then `Bank X` needs to obtain the required customer's documents, update this process and store the updated version of the core KYC on the distributed ledger. Naturally, this update incurs additional cost. To keep the system fair, this cost is equally distributed between all financial institutions operating with the customer. Section A.1 provides more details about how the fees are paid and how they maintain the system fair.

4.3 The brittleness of the Basic DLT-KYC

Consider a customer who wants to open an account at a financial services firm for malicious activities that are in breach of the regulatory framework. This customer can also open an account at other financial institutions - some might be for completely legitimate reasons,

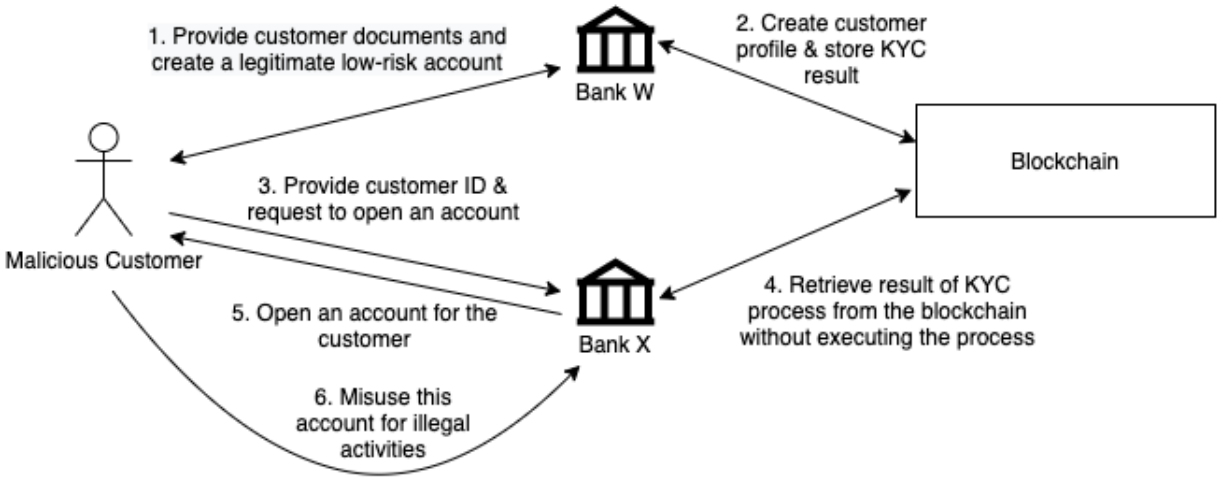

Figure 3: Simulation of the brittleness of Basic DLT-KYC. A malicious customer can open a legitimate, low-risk account at `Bank W` and misuse the system to create a new account for illegal activities at `Bank X`.

others not. It is undesirable for a financial institution to operate with a customer that is known to have misused their account for illicit activities, even when the misused account was facilitated by another financial institution. The customer can attempt to break the Basic DLT-KYC system in the following way: the customer identifies the weakest financial institution, `Bank W`, between all institutions operating on the distributed ledger and attempts to open an account at this institution. Out of all institutions operating on the distributed ledger, this institution is the one most likely to make a mistake in the KYC process and falsely accept this malicious customer by overlooking some details when executing the KYC verification. The customer may now approach any other financial institution, (`Bank X`), and attempt to open an account there. `Bank X` would not re-execute the core KYC process, as it has been executed by `Bank W`, and is more likely to accept this customer. `Bank X` is disincentivized to use the Basic DLT-KYC system, as any new customer might present a risk to the institution and the potential negative consequences might outweigh the benefits this system would bring. It even might be the case that the customer used their account for illicit activities only at `Bank X` and the account created at `Bank W` could have been entirely

legitimate, not raising any suspicions from the staff of Bank W. This scenario can also happen if the customer cannot identify the weakest institution that operates on the ledger using Basic DLT-KYC system. For any financial institution, there is always a non-zero probability that it will not execute the KYC for a customer faultlessly and accept a customer that should have been rejected.

Note that under the privacy condition, all institutions operating on the distributed ledger are anonymous to each other. When `Bank X` wants to start operating with the customer, it cannot identify that `Bank W` executed the core KYC for this customer. This anonymity further exacerbates the risk and uncertainty `Bank X` has when using the DLT-KYC system.

4.4 A Robust DLT-KYC system

Our Robust DLT-KYC system implements a probability with which the execution of the core KYC process needs to be repeated. Referring to terminology used before, `Bank A` - the first FI that the customer operates with - has to execute the core KYC process and store its result on the ledger. When the customer wants to open an account at a new institution, `Bank X`, this institution has to repeat the core KYC for this customer with a certain probability. This probability would be specified by the first institution that on-boarded the customer (`Bank A`), but it is a subject that can be adjusted based on feedback from professionals in financial sector. Perhaps each institution operating with the customer should have an equal say in determining this probability - such a change can be easily altered in the implementation.

`Bank X` does not know whether it has to repeat the core KYC process before it pays the fee to join the institution list of this customer - this depends on random number generator and the institution can only retrieve the probability with which it has to repeat the process. This ensures that `Bank X` cannot reject or accept the customer based on whether it is willing to repeat the core KYC process - a necessary requirement to meet the no-minting condition.

In case `Bank X` does not have to repeat the core KYC, no additional action has to be taken. Otherwise, `Bank X` has to obtain documents from the customer required for the core KYC

process and independently repeat the process. When repeating the process, it is possible that `Bank X` would get to a different result based on the customer verification. It could decide not to operate with the customer, even though the customer already operates with several other institutions. For instance, `Bank X` could find some transaction details that would make the customer too risky to operate with and were overlooked by the financial institution(s) that previously executed the process. `Bank X` can raise an alert that would inform the regulator and/or other financial institutions operating with the customer about this situation. The alarmed parties can react immediately, opening up space for a more in-depth investigation of the customer if it is required. Further actions would depend on the seriousness of the situation and the reason why the independent core KYC processes lead to different results. If necessary, these actions would be taken in cooperation with the regulator.

Note that we might not operate in a regulatory framework that stays constant over time and `Bank X` might have to both repeat and update the core KYC. In this case, `Bank X` would re-execute the core KYC process according to the newest regulatory framework and update the hash of the document package of this customer on the distributed ledger accordingly.

The key aspect of this modification is that any financial institution operating with a customer might be asked to repeat the core KYC process and does so independently. We provide a mathematical analysis of the impact this modification has on the security of the system in section 5.1.

Inspired by the idea of a customer profile, discussed in section 2, that could be used to characterise the behavior of the customer, our solution provides specifics on how this can be achieved. We extend the customer profile, that is uniquely identified by the customer's ID, by adding fields that determine the potential risk a customer presents. A financial institution can review this profile from the distributed ledger and see anticipated risk associated with this customer before executing further actions. For instance, the institution may decide on how many additional controls outside the core KYC process, if any, it has to execute for a

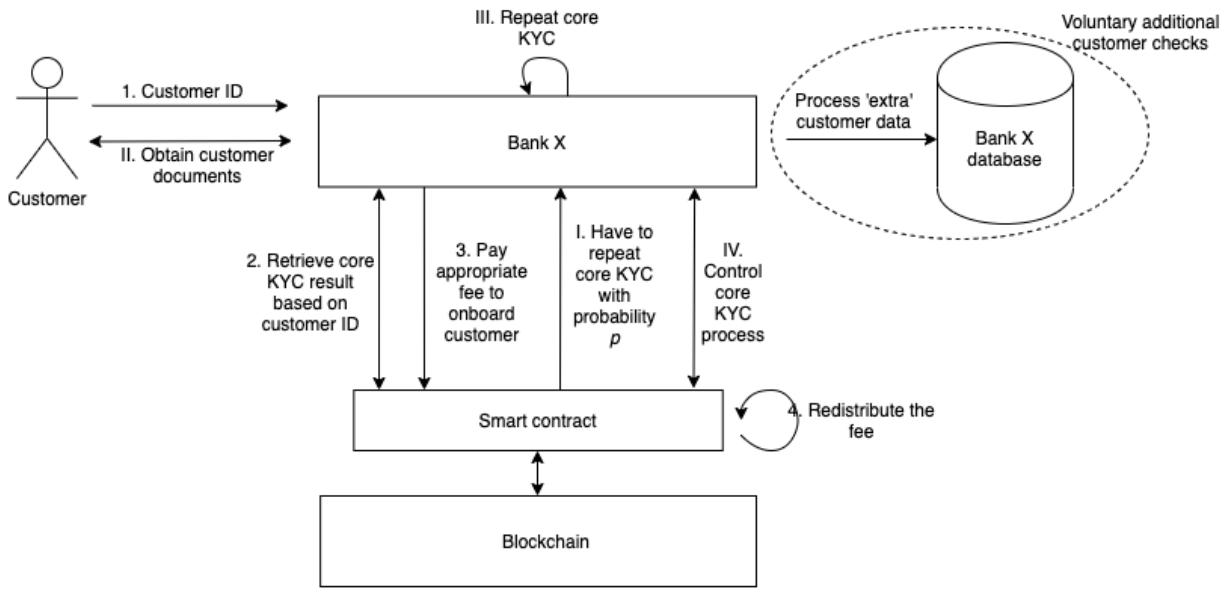

Figure 4: Customer approaches a new financial institution. This institution might have to repeat the core KYC, indicated by roman numerals.

customer before it on-boards them. Similarly, if the customer would present a too high risk, the institution may decide to reject this customer directly. We identify and implement the customer profile with the following fields:

- Customer rating. Each financial institution that operates with a customer may anonymously assign a rating of this customer. The profile would store an average of these ratings. This rating would present how satisfied the institution is with cooperation with this customer. The customer would not have access to this rating. An institution can always change this rating in case the professional relationship between the customer and the firm changes, or in case the rating would be assigned incorrectly.

- Probability for repeating the core KYC process. This probability can be chosen with respect to the level of risk the customer presents. The probability is determined by the first institution that on-boards the customer.

- Number of institutions that executed the core KYC for the customer. The more times the process was executed, the less likely is that the customer might misuse any of their

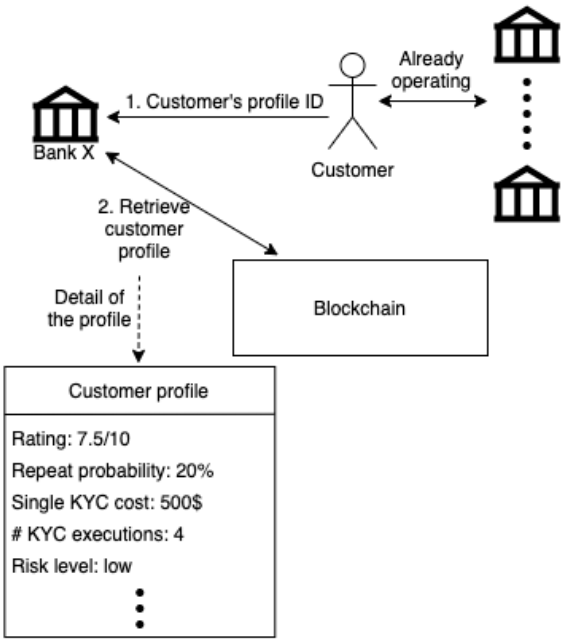

Figure 5: Customer approaches a new financial institution (`Bank X`). The institution views the customer's profile in order to check the associated risk of operating with the customer. This provides grounds for further controls the institution may decide to execute.

accounts at an institution

- Cost of single core KYC execution. This cost indicates how many resources and time has to be allocated to execute the KYC process for a customer. A higher cost indicates more time and resources which would be typically applicable to customers who are higher risk.

Figure 5 outlines how the new institution, `Bank X`, can view a customer's profile before it decides to start operating business with the customer. We identified the basic fields that can be used to estimate the level of risk a customer presents. Additional fields could be included in the profile after consulting professionals from the financial sector.

4.5 Implementation

We have implemented both the basic and robust systems as smart contract distributed applications.[4] The smart contracts were implemented in the Solidity programming language that was originally designed for the Ethereum[5] blockchain. We note that in an actual deployment, a private distributed ledger (blockchain) governed by a central authority would have to be used in practice. This is due to the highly sensitive nature of information about the customers in the financial sector. There are several private distributed ledgers intended to be used in the financial sector, such as Corda[6], Hyperledger Fabric[7] and Quorum[8]. Quorum is a private fork of the public Ethereum network created and open-sourced by JPMorgan[9]. It benefits from its similarity to the main Ethereum network, that provides a rich developers community and various frameworks for full-stack development of distributed applications that can be deployed on this ledger, and the enterprise usability implied by its private nature. These factors indicate that a fork of the Quorum blockchain would be a good candidate for the type of distributed ledger technology (DLT) that could be used if our smart contracts were to be deployed in practice.

4.6 Role of the Central Authority

From a privacy perspective, a DLT-KYC solution would be best served by a deployment over a private distributed ledger (blockchain) governed by a central authority (CA). This CA could be the regulator or it could also be a verified outsourced external company or a consortium of financial institutions that agree to use the system and be subject to the conditions and benefits it offers. We next define the role and scope of the CA including its privileges and responsibilities. We identify the following two requirements.

[4] https://github.com/Matus23/KYC-Ethereum-smart-contracts
[5] https://ethereum.org/
[6] https://www.corda.net/
[7] https://www.hyperledger.org/use/fabric
[8] https://www.goquorum.com/
[9] https://www.jpmorgan.com/global/jpmorgan

- Onboarding of an entity that wants to join the private ledger. In the case of a permissioned private ledger, in which only a verified institution could enter this would be a mandatory step. To maintain privacy, the institutions appear to be anonymous to each other on the ledger, and as a result it is important that each institution's identity on the ledger has been thoroughly vetted.

- Maintaining a database that offers a mapping between accounts on the ledger and real identities of the institutions behind those accounts. This requirement ensures that the identity of any entity on the ledger can always be obtained by the regulator when the regulator needs to proceed to any legal action.

5 Security Analysis: DLT-KYC Robustness

5.1 Mathematical background

Any customer that has an account at a financial institution may decide to use this account for illegal activities, such as money laundering or financing of terrorism. There are numerous measures put in place to avoid this happening, one of which is a thorough KYC process. However, even after an FI executes a careful and adequate KYC verification of a customer and decides to facilitate them, this customer may later misuse their account for illegal activities. In some cases this might occur due to an inadequate verification of the customer during the KYC process and could have been prevented had the KYC been executed with more care. While this is rather unlikely, many financial firms operate with plenty of customers such that this sometimes happens and leads to negative consequences for the firm we specified earlier in section 3.

In order to simplify the terminology, we say that the 'KYC was executed correctly' when the KYC process was executed adequately and thoroughly with respect to a regulatory framework and even if the customer turns out to misuse their account for illegal activities,

this could not have been anticipated and prevented during the KYC process. We say the 'KYC was executed incorrectly' when there was made a mistake in the core KYC process such that if the customer turns out to misuse their account for illegal activities, this could have been prevented by a more thorough KYC process execution.

We say that our system is for a given customer secure when the customer does not represent any risk to the institutions operating with the customer based on the core KYC process. In other words, we say the system is for the customer secure when even if the customer uses their account at any of the institutions for illegal activities, this could not have been prevented by the core KYC process.

> *Question: What is the probability that, for a given customer, the core KYC process was executed correctly with respect to a regulatory framework? In other words, what is the probability that the system, for a given customer, implies no risk that could have been prevented by the core KYC verification process, to any of the financial institutions the customer operates with?*

Our analysis is based on the following assumptions.

Assumption 1 (Independence)**.** *For a given customer, every financial institution that wants to operate with the customer and has to execute the core KYC process would do so independently of the other financial institutions.*

Assumption 2 (Equal execution)**.** *All financial institutions would adequately execute the core KYC process, with respect to a regulatory framework, with the same probability. We call this probability p_{ok}. While it is unlikely that each institution would execute the KYC process adequately with the same probability, we can think of p_{ok} as the average probability of adequately executing the core KYC process and a hyper-parameter that can be tuned and chosen based on estimates made by professionals in the financial sector to model security of the system.*

Assumption 3 (Cooperation)**.** *When a financial institution that repeats or updates the core KYC process comes to a different conclusion than the preceding institution(s) that executed the process for this customer, it shares this finding with these institutions and does not keep it for itself.*

Let us call the average probability that an FI executes the core KYC correctly p_{ok}. The probability that a future FI will have to repeat the core KYC process is p_{rep}. Finally, let $list\_size$, also abbreviated as LS, be the number of institutions operating with a customer.

After the core KYC for a customer is executed by the first financial institution, we know that each financial institution that would like to start operating with the customer in the future will either have to repeat the core KYC process with probability p_{rep}, or just enter the list without repeating it with probability $1-p_{rep}$. Let R be a random variable expressing how many times the process was repeated and LS be a random variable expressing the number of institutions that facilitate the customer.

The probability that the core KYC process was repeated k times, given that the customer operates with $n+1$ institutions, behaves according to the Bernoulli distribution[10] and is determined by the following equation.

$$P(R = k \mid LS = n+1) = p_{rep}^{k}(1-p_{rep})^{n-k}\binom{n}{k} \tag{1}$$

The first financial institution has to execute the core KYC. There are n additional financial institutions that need to re-execute the core KYC with probability p_{rep}. The first term of the product, p_{rep}^{k}, expresses the probability that the process was repeated k times. This would mean that the remaining $n-k$ times, the process was not repeated - expressed by the second term $(1-p_{rep})^{n-k}$. The extra term $\binom{n}{k}$ expresses the number of possible combinations how k core KYC processes can be repeated out of n trials.

Let us introduce a new random variable, OK, that expresses how many times the core

[10] `https://en.wikipedia.org/wiki/Bernoulli_distribution`

KYC was repeated correctly. The probability that the core KYC process was repeated correctly i number of times, given that it was repeated k number of times, is given by:

$$P(OK = i \mid R = k) = p_{ok}^{i}(1 - p_{ok})^{k-i}\binom{k}{i} \tag{2}$$

In order to find out if the first institution executed the core KYC correctly, there needs to be at least one financial institution repeating the core KYC that does so correctly and points out to this problem. This institution would either obtain the same result, or it would spot the difference between the two processes. Given that there are k institutions repeating the core KYC, this probability can be written as:

$$\begin{aligned} P(OK \geq 1 \mid R = k) &= 1 - P(OK = 0 \mid R = k) \\ &= 1 - (1 - p_{ok})^{k} \end{aligned} \tag{3}$$

We need to obtain the probability that the core KYC was repeated correctly i times as a function of the size of the institution list (LS), not as just as a function of the number of core KYC repetitions (R). We need this expression because random variable R depends on p_{rep}, which is a hyper-paremeter that would be set by the institutions. On the other hand, the number of institutions operating with a customer is only dependent on the customer's incentive to operate with other institutions and cannot be tuned by the system.

The probability that the core KYC was adequately repeated i times out of k repeated

attempts, where $i \leq k$, when the customer operate with $n+1$ institutions is:

$$\begin{aligned} P(OK = i \mid LS = n+1) &= \sum_{k=0}^{n} P(OK = i, R = k \mid LS = n+1) \\ &= \sum_{k=0}^{n} P(OK = i \mid R = k, LS = n+1) P(R = k \mid LS = n+1) \\ &= \sum_{k=0}^{n} P(OK = i \mid R = k) P(R = k \mid LS = n+1) \\ &= \sum_{k=0}^{n} p_{ok}^{i} (1 - p_{ok})^{k-i} \binom{k}{i} p_{rep}^{k} (1 - p_{rep})^{n-k} \binom{n}{k} \end{aligned} \tag{4}$$

The above equation used the sum and product rule in probability theory [10]. An important observation is that $P(OK = i \mid R = k, LS = n+1) = P(OK = i \mid R = k)$, as when we know how many times the core KYC was repeated, the overall number of institutions the customer operates with is not relevant.

We can simplify the above relation, as we need to find out the probability of repeating the core KYC correctly at least once, instead of exactly i times. Putting equation 3 and 4 together, we obtain:

$$P(OK \geq 1 \mid LS = n+1) = \sum_{k=1}^{n} [1 - (1 - p_{ok})^{k}] p_{rep}^{k} (1 - p_{rep})^{n-k} \binom{n}{k} \tag{5}$$

Note that the sum here goes from $k = 1$ instead of $k = 0$, as was the case in equation 4. In order to have at least one correctly repeated core KYC process, the process itself has to be repeated at least once in the first place. Thus, for the case $k = 0$, the probability $P(OK \geq 1 \mid R = 0) = 0$ by definition.

If the first institution executed the core KYC incorrectly, our scheme is only secure if there is at least one of the following institutions that correctly re-executed the core KYC. If the first institution executed the core KYC correctly, then the following institutions that repeated the process could make a mistake in the process and the system would still be secure.

It would be secure as it would continue to rely on the the first execution of the process and the consequent ones would be recognised as incorrect. Using equation 5, the probability that the system is safe (s), given than the customer operates with $n+1$ institutions, is mathematically written as:

$$\begin{aligned} P(s \mid LS = n+1) &= P(OK \geq 0 \mid LS = n+1)p_{ok} + P(OK \geq 1 \mid LS = n+1)(1-p_{ok}) \\ &= p_{ok} + (1-p_{ok}) \sum_{k=1}^{n} [1-(1-p_{ok})^k] p_{rep}^k (1-p_{rep})^{n-k} \binom{n}{k} \end{aligned} \tag{6}$$

This holds true as $P(OK \geq 0 \mid list\_size = n+1) = 1$ by definition.

5.2 Results of the analysis

We present some model situations coming from our mathematical analysis to demonstrate that our system is on average more cost-efficient and secure than the KYC scheme that is currently put in practice. We say on average because our model situations operate with the average probability p_{ok} that a financial institution executes the core KYC for a customer correctly. The security is also enhanced in comparison to the basic DLT-KYC system, but there is a compromise between the cost-reduction and enhanced security our solution offers.

The first diagram simulates a situation in which a financial institution is assumed to correctly execute the core KYC process in 90% of the cases and incorrectly in the remaining 10% cases. The core KYC process has to be repeated with a 50% probability. This applies to each financial institution operating with the customer, except for the first institution that has to execute the core KYC unconditionally. The y axis on the left shows the probability that the system is for a given customer secure. This is achieved when at least one of the institutions executed the core KYC for the customer correctly. The y axis on the right shows the price of the core KYC for a single financial institution as a fraction of the average cost of executing the core KYC for the customer.

Note that the average probability of executing the core KYC adequately in 90% is only

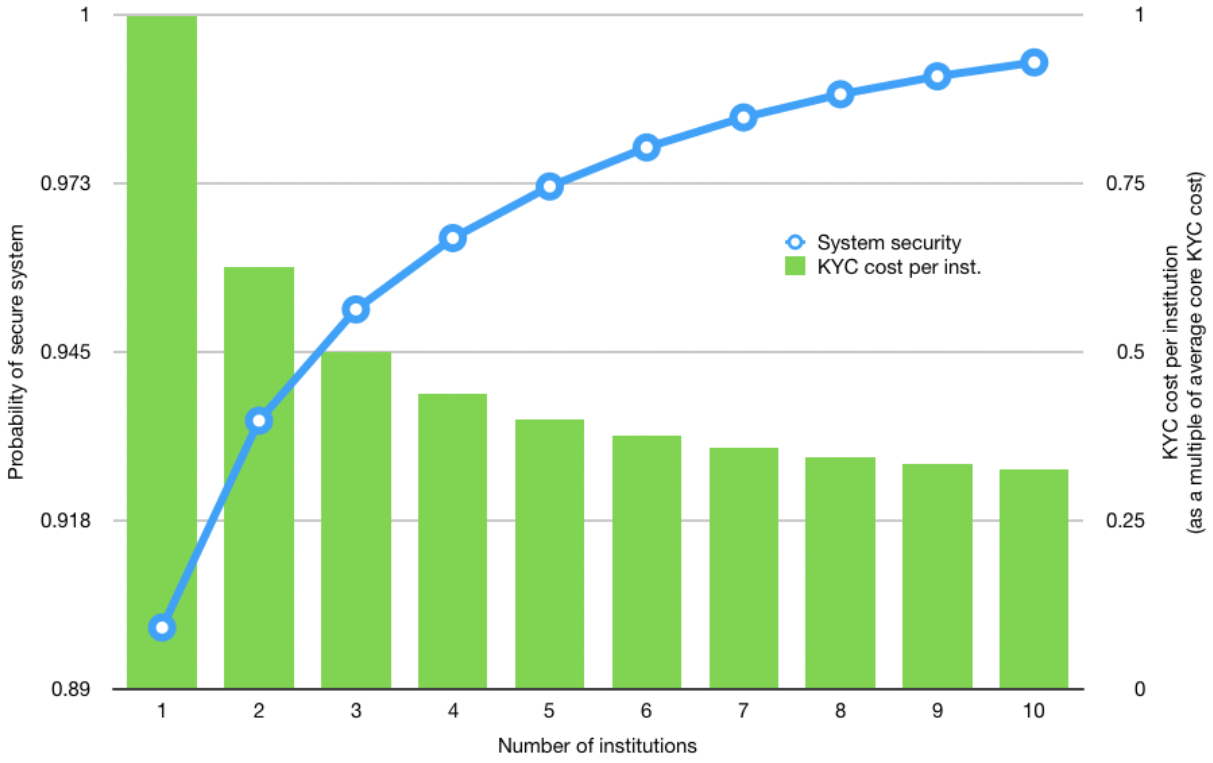

Figure 6: Security of the system for a customer. An institution executed on average core KYC adequately in 90% cases. The Core KYC needs to be repeated with 50% probability.

an estimate and is most likely well below the real value. However, this value well illustrates how the security of the system increases. As the number of institutions a customer operates with increases, the average cost of executing the core KYC for an institution decreases and the security of the system increases. This shows that our system is both more financially efficient and more secure than the current KYC system. The current system, where the core KYC is executed by each financial institution individually, would have the system security stuck at 0.90 and the cost per institution at 1 (equal to c), for any number of institutions the customer operates with. In other words, the current KYC system presents no improvement as the customer starts operating with more institutions.

The probability of repeating the core KYC process is a hyper-parameter that would be selected based on opinions of professionals representing the financial institutions. It could

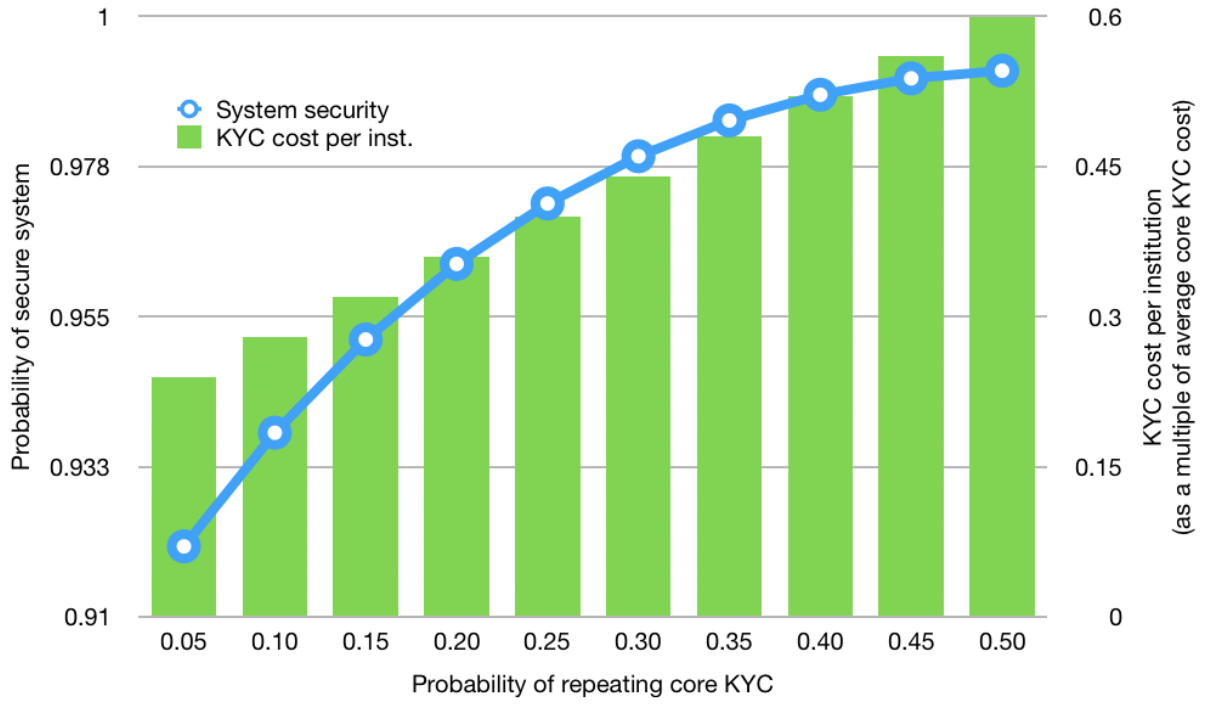

Figure 7: Security of the system for a customer. Institution on average executed core KYC adequately in 90% cases. Customer operates with 5 institutions.

be different for each customer and be dependent on how much risk the customer represents. The second diagram illustrates a situation where the customer is operating with 5 financial institutions. As before, a financial institution executes the core KYC process adequately in 90% cases. The x axis now represents the different probabilities of repeating the core KYC process. The y axis on the left and on the right remain unchanged and represent the security of the system and the cost associated with the core KYC per institution respectively.

We can see that the security of the system as well as the cost incurred for the core KYC per institution increases as the probability of repeating the core KYC increases. This is because for a fixed number of institutions, the process will be repeated more often when the probability of repeating it is higher, and repeating the core KYC process increases the overall cost incurred by this. Figure 7 demonstrates the compromise between increased security of the system and decreased cost reduction we mentioned earlier. If we used the current KYC

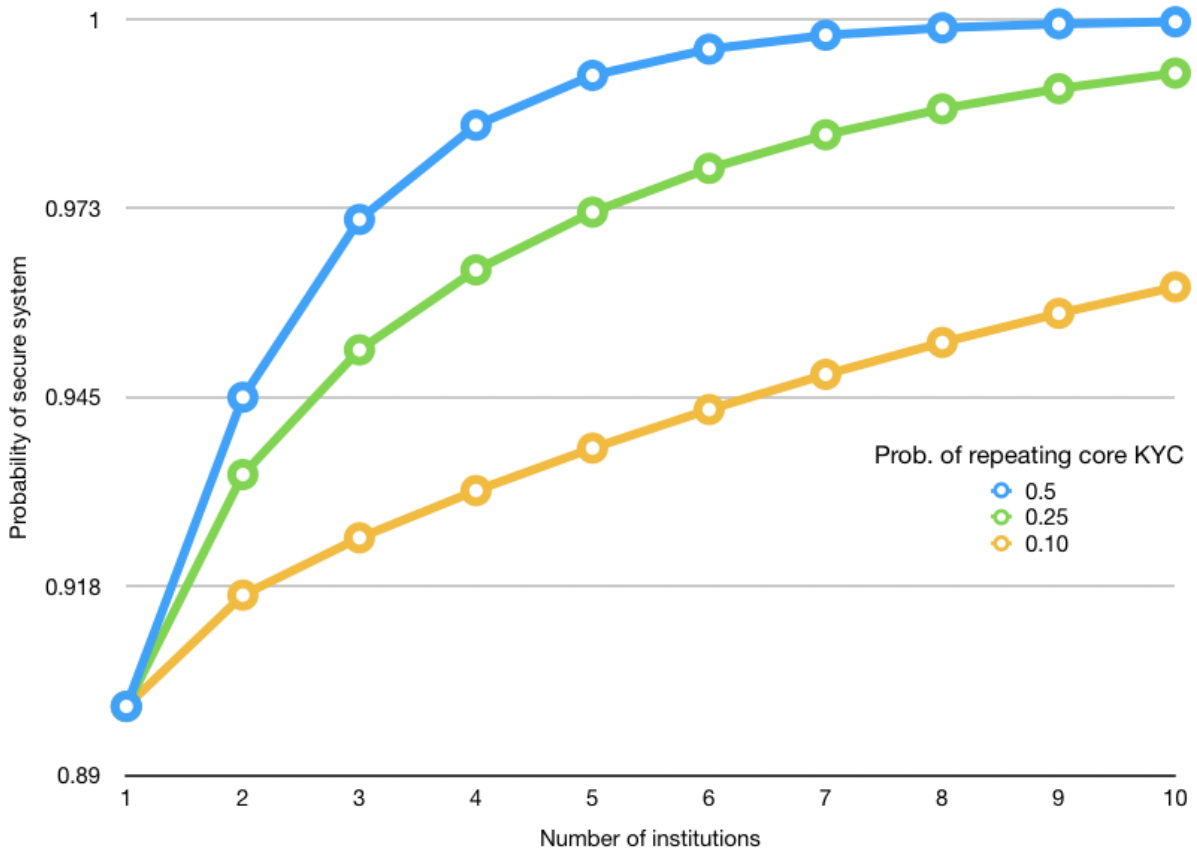

Figure 8: Security of the system for a customer. An institution executed on average core KYC adequately in 90% cases.

system, the cost would be 1 (equal to c) and the system security would be 0.9.

The final figure compares the security of the system for various probabilities of repeating the core KYC process. We observe this as a function of the number of institutions the customer operates with. The x axis shows the number of institutions and the y axis the security of the system. It can be observed that when the probability of repeating the core KYC process is 50%, the security of the system quickly asymptotically approaches 100%.

6 One-Tier problem

Let `Bank X` be an FI that operates with many clients, is in charge of a large capital, has been on the market for many years and has a strong reputation to uphold. Imagine `Bank X` would like to start operating with a new customer using our Robust DLT-KYC system. Assume that the customer only operates with one other FI, `Bank A`, which executed the core KYC. When `Bank X` is requested to pay a fee to on-board the customer, without having to update or independently repeat the process, it effectively relies on the result of the core KYC process executed by `Bank A`. While this enables effective sharing of the costs incurred by this process, it also presents a certain risk to `Bank X`. `Bank X` cannot identify that `Bank A` executed the process - it only knows that one FI, that operates on the ledger, executed the core KYC for this customer.

The issue is that some FIs might be better at executing this process than others, and by relying on the execution of the core KYC process by another institution, they put their own reputation on the line. This is further aggravated due to the anonymity between the FIs that must be preserved to protect the privacy of the customers. This is what we call the *one-tier problem* - the assumption that all FIs do presumably equal job of executing the core KYC process and that they will rely on each other's work.

The Robust DLT-KYC system partially tackles this issue when a customer operates with many FIs. In such a case, due to the law of large numbers, the core KYC process would have been repeated by sufficiently many FIs of various sorts such that the risk of on-boarding this client would be minimised. However, it is important to consider that most clients might not operate with sufficiently many FIs and a different approach has to be taken.

To address this problem, we propose to create a profile for each institution present on the ledger that would provide information about the capabilities of this institution executing the KYC process. Our smart contract implementation creates a profile that provides a rating of a financial institution and list of customer profiles the institution operates with. The rating represents an institution's quality of adequately executing the core KYC process with respect

to the regulatory framework. The existence of a rating would imply that FIs operating on the ledger would no longer be in a single tier, breaking the assumption that they all execute the core KYC process correctly with the same probability.

When `Bank X` wants to now operate with a new customer, it can view the list of institutions that executed or updated the core KYC process for this customer, together with the ratings of these institutions. Based on this information, `Bank X` can anticipate the level of trust it can have in the work of these institutions. In our example, `Bank X` would be able to see `Bank A`'s rating, without knowing the identity of the bank, and could decide on how reliable the core KYC process is, as executed by `Bank A`. If `Bank X` would deem `Bank A` unreliable, it could decide to repeat the core KYC process or execute additional controls of the customer, presenting an extra cost that minimises the potential risk. If it would deem `Bank A` highly reliable, it might decide not to execute any additional controls of the customer, fully utilising the cost sharing of the single core KYC process.

The rating of a financial institution is the average of ratings that were assigned to this institution by other institutions operating on the blockchain. The rating `Bank A` assigns to `Bank B` presents how satisfied `Bank A` is with `Bank B`'s execution of the core KYC for customers they operate together with and for whom `Bank B` executed, updated or repeated the core KYC process. A financial institution (`Bank A`) can rate another financial institution (`Bank B`) when the following conditions are met:

- `Bank A` and `Bank B` must be operating with at least one mutual customer. Let S be the set of customers that operate with both institutions. If the set S is an empty set, the rating cannot be assigned.

- `Bank B` must have executed, updated, or repeated the core KYC for at least a single member of set S. Set S represents customers that operate with both institutions, but does not specify whether `Bank B` was involved in any way in verifying the any members of this set.

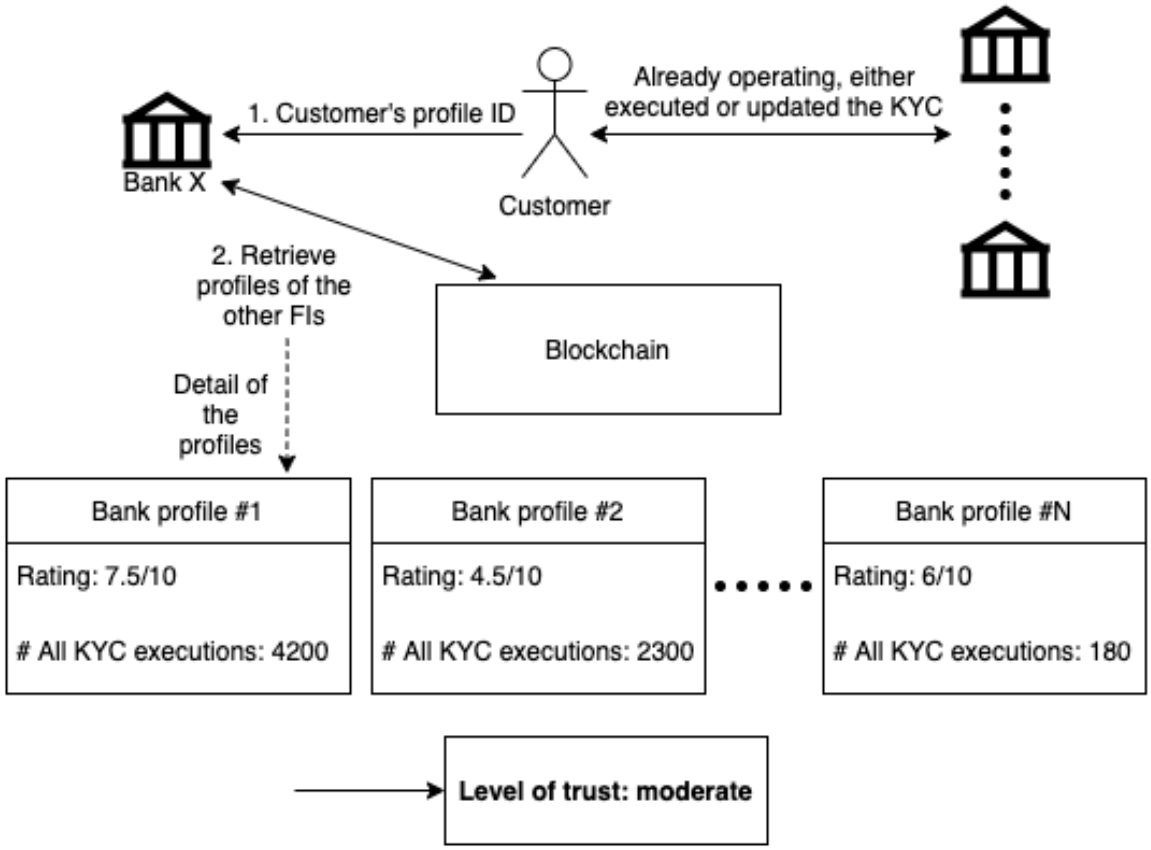

Figure 9: Customer approaches a new financial institution `Bank X`. `Bank X` can view which institutions executed or updated the core KYC for this customer and ratings of these institutions. Based on this information, it can forecast a level of trust it puts into the work of these institutions and decide on additional customer controls outside the blockchain solution.

- `Bank A` can only give `Bank B` a single rating, no matter what is the size of S. `Bank A` can review and change this rating based on the current situation.

In Robust DLT-KYC system the institutions maintain their privacy by opening up a unique account on the ledger for each customer they operate with. In this modified version, we propose that an institutional profile would be created that includes a list of all customer profiles an institution operates with. This does not directly reveal identity of the FI, but a pattern between accounts the institution uses and the customers it operates with could be found that would unravel identity of the institution.

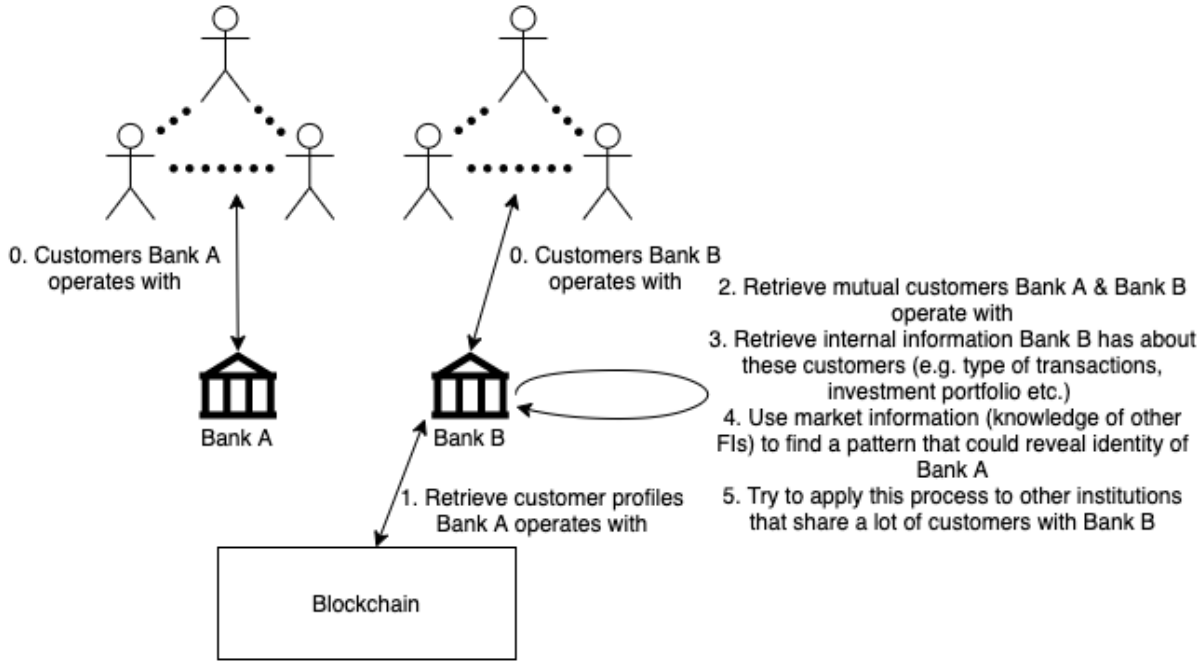

Figure 10: This figure illustrates how `Bank Z` may try to unravel the real identity of `Bank Y` based on its blockchain profile. It can try to unravel the identity of any other institution on the blockchain - the more customers, and the more unique the customers are, the higher the probability of success. `Bank Y`, and any other institution on the blockchain, can try to unravel the identity of `Bank Z` or another institution.

Whether this scenario would be real or just hypothetical has to be consulted with professionals from financial sector. Due to the potential of this risk, this solution is not included with our Robust DLT-KYC implementation, but is outlined in this separate section and given its own separate smart-contract implementation[11].

7 Future Work

Several studies [14, 21, 22] suggested that using DLT could bring significant benefits in the general processes related to KYC/AML. At present, recording transactions associated with customers' accounts on the distributed ledger is out of scope of this paper. Our DLT-KYC system stores only transactions that are required for effective cost-sharing between FIs op-

[11] `https://github.com/Matus23/KYC-Ethereum-smart-contracts`

erating with the same customer universe. The system is not intended for storage of the transactions between a customer and a financial institution; as the ledger is accessible to all parties operating on it, revealing this sensitive information is not desirable for customer privacy. The ledger could be adjusted to provide on-chain transactions in a privacy-preserving manner, but this is by no means trivial and would require further investigation as well as cooperation with experts from the financial sector on how it could be best achieved. Alternative mechanisms that use secure multiparty computation for conducting cross organization transaction pattern searching is another avenue that would benefit this direction, cf. [16].

Lomazzo [12] suggests that a blockchain can be used to on-board new customers while artificial intelligence can be used to determine customer's interests and recommend a type of account and portfolio they could invest in based on these findings. Our paper outlines how a user profile, stored on the ledger, can be used to estimate the risk level the customer presents. An FI that would like to start business with a customer can use this risk level to determine the extent of additional controls it may have to apply. The institution might decide to lessen these controls when the customer presents a low level of risk, and dwell on these controls when the customer presents a high level of risk. We mentioned that the customer profile on the ledger might be extended by other fields, such as based on customer's activity, volume of transactions, etc. If we could provide a customer profile with sufficiently many features, artificial intelligence and machine learning techniques could be used even during the on-boarding phase. This could further lower the costs by automating tasks that currently have to be done by human operators. The challenge would be to build a strong customer profile, ensure privacy and anonymity of the profile, and to gather data about sufficiently many customers to implement this in a testing environment.

In the context of the one-tier problem, we identified the assumption that all FIs have equal capability of executing the KYC process may be problematic in practice. We offered some remedy by introducing a profile of each FI on the ledger that would provide some insight about the institution and its ability to execute the KYC process. However, this

reduces the level of privacy offered by the system. Further research would be requried to address this problem in a privacy-preserving manner.

Regarding the sharing and management of the customer KYC documents, they are always stored in a private database maintained by an institution. When the same customer starts operating with other FIs, the customer can either directly provide these institutions with the documents again, or a system where the customer's documents would be shared can be used. Sharing the documents would present better user experience for the customer, and there are several ways this could be achieved: (1) using a database governed by the central authority, (2) using a distributed database system such as those suggested for the health care industry, cf. [23, 24], that control access via a credit system for queries , (3) using IPFS[12] or some other peer-to-peer protocol for distributed storage. Each solution has its pluses and minuses - it is important to consider the extra cost and possibly increased responsibilities of the CA or of other parties that would be in charge of the database system.

Finally, with respect to the possible misuse of our system by malicious (or hacked) FIs, a first measure is to include several checks in the smart contracts to provide authorization of the FIs and restrict the actions they can take to limit their leverage to what is the minimum required. However, an FI can still misuse the system by retrieving a customer profile, after it is given ID of this profile by the customer, and use it to start operating with the customer without paying the appropriate fee. In such a case, the regulator can see that there is no record on the ledger that would enable the institution to operate with the customer and if a random control finds this out, the institution would be severely penalized. This mechanism provides strong incentive for an FI not to misuse the system this way, but it relies on the random controls that would be executed by the regulator. Were this solution put in practice, an interface that would enable an FI to retrieve the customer profile only after it paid the appropriate fee could be implemented.

[12] `https://ipfs.io/`

8 Conclusion

The KYC process is one of emerging use-cases in the setting of financial services that can benefit by the use of DLT. When put in practice, our Robust DLT-KYC system can reduce compliance costs while offering improved security (in the sense of robustness) compared to previous proposals. The proposal would still decrease the waiting time for KYC due to distributing the load across a set of participating institutions and organizations, improving in this way, customer experience. At the same time, the system improves monitoring of customers by creating a customer profile on the distributed ledger. This profile can be used to anticipate the risk level of the customer, and to determine the amplitude of additional controls the financial institution decides to execute based on the level of risk the customer presents. By identifying customers that present a higher level of risk, the institutions can concentrate their staff on controlling such customers, potentially combating money laundering and financing of terrorism more effectively.

Finally, cooperation between the financial institutions and the regulator on a distributed platform can bring other long-term benefits, such as automating the KYC process by using artificial intelligence, and presents a benefit to all parties - the regulator, financial institutions and their customers. We outlined several issues that need further investigation, including addressing the one-tier problem, offering better privacy and detecting failures of the no-minting property. We expect that our smart-contract implementation, and the accompanied security and performance analysis demonstrate the immediate practicality of our approach as well as motivate further research and development in this area.

References

[1] Hong Kong Monetary Authority. Whitepaper 2.0 on distributed ledger technology. 2017.

[2] Roman Beck, Michel Avital, Matti Rossi, and Jason Bennett Thatcher. Blockchain technology in business and information systems research, 2017.

[3] Matthew Britton. Could blockchain solve the kyc/aml challenge?, September 2016.

[4] Colin Powell Charles Freeland. Customer due diligence for banks. 2001.

[5] FATF. Fatf 40 recommendations, 2004.

[6] FCA. Fca fines standard chartered bank £102.2 million for poor aml controls, 2019.

[7] FCEN. History of anti-money laundering laws, Accessed in March 2020.

[8] FDIC. Bank secrecy act, anti-money laundering, and office of foreign assets control, 1970.

[9] Pedro Goncalves. Money laundering fines total $8.14bn in 2019, 2020.

[10] Arno Onken Iain Murray. Bayesian inference and prediction, 2019.

[11] Shani Koren. Know your customer, 2018.

[12] Jessica Lomazzo. Fintech's impact on wealth management. *Ivey Business School*, 2016.

[13] YVONNE Lootsma. From fintech to regtech: The possible use of blockchain for kyc. *Fintech To Regtech Using block chain*, 2017.

[14] Matthias Memminger, Mike Baxter, and Edmund Lin. Banking regtechs to the rescue. *URL www. bain. com/Images/BAIN_BRIEF_Banking_Regtechs_to_ the_Rescue. pdf*, 2016.

[15] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. Technical report, Manubot, 2019.

[16] Dimitar Jetchev Kevin McCarthy Jakob Odersky Alexander Petric Abson S. Tang Nicolas Gama, Mariya Georgieva. Detecting money laundering activities via secure multipartycomputation for structural similarities in flow networks, accessed in 2020.

[17] Vimalkumar Pachaiyappan and R Kasturi. Block chain technology (dlt technique) for kyc in fintech domain: A survey. *International Journal of Pure and Applied Mathematics*, 119(10):2108, 2018.

[18] Matt Packer. Five minutes on... the surge in anti-money laundering fines, 2019.

[19] José Parra-Moyano and Omri Ross. Kyc optimization using distributed ledger technology. *Business & Information Systems Engineering*, 59(6):411–423, 2017.

[20] José Parra-Moyano, Tryggvi Thoroddsen, and Omri Ross. Optimized and dynamic kyc system based on blockchain technology. *Available at SSRN 3248913*, 2018.

[21] Neepa Patel. Blockchain kyc/aml utilities for international payments, 2017.

[22] Refinitiv. A blockchain enabled kyc solution: New horizon or falsedawn?, 2018.

[23] Michael Siegenthaler and Ken Birman. Privacy enforcement for distributed healthcare queries. In *2009 3rd international conference on pervasive computing technologies for healthcare*, pages 1–6. IEEE, 2009.

[24] Michael Siegenthaler and Ken Birman. Sharing private information across distributed databases. In *2009 eighth ieee international symposium on network computing and applications*, pages 82–89. IEEE, 2009.

[25] Prince Sinha and Ayush Kaul. Decentralized kyc system. *International Research Journal of Engineering and Technology (IRJET)*, 5(8):1209–1210, 2018.

[26] Nedyalko Valkanov et al. Smart compliance or how new technologies change customer identification mechanisms in banking. *Economics and computer science*, (2):12–19, 2019.

[27] M. Walport. Distributed ledger technology: Beyond block-chain, 2016.

[28] Iza Wojciechowska. What is kyc and why does it matter?, 2019.

A Appendix - Fulfilling the KYC conditions

In order for our Robust DLT-KYC system from section 4.4 to be usable, it is important that it fulfills the KYC conditions outlined in section 1.

A.1 Proportionality

The proportionality condition ensures equal cost distribution between all institutions operating with a customer. It does so as follows: when the first financial institution executes the core KYC process for a new customer, it has an associated cost with this process. Let us call c the average cost of executing the core KYC process for a customer. In our repeated scheme, when a customer would like to operate with a new financial institution, `Bank X`, this institution has to repeat the core KYC with probability p_{rep}. As the process might need to be repeated, updated, or both, we create a new variable c_{agg} which is the aggregate cost incurred by executing possibly several core KYC processes for this customer. If there are k institutions operating with a customer, in order for the proportionality condition to hold, each must have had a cost of c_{agg}/k. There are four possible scenarios `Bank X` could be facing before operating with the customer.

1. Neither KYC update, nor repetition of the process is required.
2. Update the core KYC without repeating the process.
3. Repeat the core KYC without any update.
4. Repeat the core KYC and execute an update.

Let us start with the first scenario. Assume there is only one institution operating with the customer, `Bank A`, which executed the core KYC. This costed c and so the aggregate cost is the same, $c_{agg} = c$. In order to on-board the customer, `Bank X` has to pay a fee of c_{agg}/k, where k is the number of financial institutions already operating with the customer plus this new institution, `Bank X`. In this case, $k = 2$. After `Bank X` pays this fee, it is equally distributed between all of the $k-1$ institutions that were already operating with the customer. In our scenario, this means only `Bank A`. Both institutions have paid $c_{agg}/2$ and `Bank X` can now on-board the customer with no additional costs. The system is fair.

Assume now there are already $k-1$ institutions operating with the customer and that all these $k-1$ institutions have equally shared the cost of executing the core KYC. This means that each institution has paid $c_{agg}/(k-1)$. Note that aggregate cost may no longer be equal to the cost of a single core KYC. In order for `Bank X` to on-board the customer,

the institution needs to pay c_{agg}/k. This new contribution of value c_{agg}/k is then equally distributed between the $k-1$ institutions that are already operating with the customer. The cost that a financial institution already operating with the customer will face after this new institution joins in is the following:

$$\begin{aligned} \frac{c_{agg}}{k-1} - \frac{c_{agg}}{k}\frac{1}{k-1} &= \frac{c_{agg} * k - c}{k(k-1)} \\ &= \frac{c_{agg}(k-1)}{k(k-1)} \\ &= \frac{c_{agg}}{k} \end{aligned} \tag{7}$$

We can see that `Bank X` pays c_{agg}/k and the system remains fair for each institution also after `Bank X` joins in.

The second scenario from the list requires `Bank X` to update the core KYC process without repeating the part that was already executed. This update incurs an additional cost c_{upd} for `Bank X`. For brevity, let us assume there are already $k-1$ institutions operating with the customer that have so far equally shared the price, by having paid $c_{agg}/(k-1)$. In order to on-board the customer, `Bank X` first pays c_{agg}/k which is distributed equally between the $k-1$ institutions. After it executes the update of the core KYC, it uploads hash of the updated document package on the distributed ledger. Putting the hash on the ledger does not represent any cost, but executing the process itself incurred cost c_{upd} that was entirely covered by `Bank X`. When the hash of the document package gets updated on the ledger, the contract automatically creates a debt of value c_{upd}/k for all $k-1$ institutions that were already operating with the customer, not including `Bank X`. The overall cost for `Bank X` then is:

$$\begin{aligned} \frac{c_{agg}}{k} + c_{upd} - \frac{c_{upd}(k-1)}{k} &= \frac{c_{agg}}{k} + \frac{c_{upd} * k - c_{upd} * k + c_{upd}}{k} \\ &= \frac{c_{agg} + c_{upd}}{k} \end{aligned} \tag{8}$$

All other institutions had paid c_{agg}/k before the update was executed and c_{upd}/k afterwards. This shows that the proportionality condition is met. In order to meet the proportionality condition in the future, it is necessary to update the aggregate cost and the cost of executing a single core KYC for the customer. Written as an expression, $c_{agg} \leftarrow c_{agg} + c_{upd}$ and $c \leftarrow c + c_{upd}$.

The third scenario is mathematically identical to the second one. The difference is that instead of the cost of update, c_{upd}, `Bank X` has to cover the cost of executing the core KYC c. Proof of meeting this condition follows equation , with the update cost c_{upd} replaced by c.

One difference is that `Bank X` does not need to update hash of the document package, only compare the result it obtains with the result stored on the distributed ledger. The aggregate cost is increased by c.However, c - the cost of a single core KYC execution - remains the same.

The last scenario requires both an update of the core KYC and re-execution of the process in its previous form. This is equivalent to re-execution of the core KYC from the scratch with respect to the newest legislative framework.

Assume that the customer is currently only operating with one financial institution, `Bank A`, when it approaches `Bank X`. `Bank A` executed the core KYC which incurred price c. The aggregate cost is currently also c. In order to on-board the customer, `Bank X` has to pay $c/2$ that is received by `Bank A`. It then needs to repeat the core KYC up to the updated legislative framework. This covers the cost c of repeating the core KYC up to the previous standards plus the cost of the update c_{upd}. When `Bank X` updates hash of the document package, the contract creates a debt of value equal to the cost incurred by `Bank X` for re-executing the updated core KYC, which is $(c + c_{upd})/2$. `Bank A` owes this debt to `Bank X`. After these transactions take place and `Bank A` pays to `Bank X` the debt, the total cost incurred for `Bank A` is:

$$c + \frac{c + c_{upd}}{2} - \frac{c}{2} = c + \frac{c_{upd}}{2} \tag{9}$$

The incurred cost for `Bank X` is:

$$c + c_{upd} + \frac{c}{2} - \frac{c + c_{upd}}{2} = c + \frac{c_{upd}}{2} \tag{10}$$

We can see the two costs are equivalent and the proportionality condition holds true. The aggregate cost and the cost of executing a single core KYC would need to be updated again accordingly: $c_{agg} \leftarrow c_{agg} + c + c_{upd}$, $c \leftarrow c + c_{upd}$.

To prove this condition in general, let us now assume the customer is already operating with $k - 1$ institutions and that each institution has so far equally distributed this cost. When the customer wants to operate with a new institution, `Bank X`, this institution has to pay c_{agg}/k to on-board the customer. It obtains access to the customer's documents and requests additional documents required for the updated part of the core KYC. `Bank X` re-executes the core KYC up to the newest legislative framework which incurs cost $(c + c_{upd})$. It updates hash of the document package for the customer, by which it creates a debt of value $(c + c_{upd})/k$ to each of the $(k - 1)$ institutions already operating with the customer.

When all institutions pay this debt back, the total cost for `Bank X` is the following:

$$\begin{aligned}\frac{c_{agg}}{k} + c + c_{upd} - \frac{(c + c_{upd})(k-1)}{k} &= \frac{c_{agg} + k(c + c_{upd}) - (c + c_{upd})(k-1)}{k} \\ &= \frac{c_{agg} + c + c_{upd}}{k}\end{aligned} \tag{11}$$

The cost for an institution that was already operating with the customer after `Bank X` joined in is:

$$\begin{aligned}\frac{c_{agg}}{k-1} + \frac{c + c_{upd}}{k} - \frac{c_{agg}}{k(k-1)} &= \frac{k * c_{agg} + (k-1)(c + c_{upd}) - c_{agg}}{k(k-1)} \\ &= \frac{(k-1)(c_{agg} + c + c_{upd})}{k(k-1)} \\ &= \frac{c_{agg} + c + c_{upd}}{k}\end{aligned} \tag{12}$$

We can see the two costs are equal and the proportionality condition is met.

A.2 Privacy

The privacy condition has two aspects: privacy of the financial institutions and privacy of the customers. The privacy of financial institutions requires the following: First, each institution on the distributed ledger is anonymous to other institutions operating on the ledger and only known to the regulator. Second, an institution cannot know what customers another institution is operating with.

The first criterion is accomplished by not storing the identity of an institution on the distributed ledger, but in a private database only accessible by the regulator.

The second criterion is fulfilled by requiring that each institution uses a unique blockchain account to operate with each customer. The following situation outlines why the second criterion is important. Let us assume `Bank A` and `Bank B` are two financial institution operating on the blockchain with multiple customers. If `Bank B` could identify the blockchain accounts `Bank A` uses to operate with its customers, there would be a large overlap in the customers `Bank A` and `Bank B` are operating with, it could reveal a pattern that would enable the institutions to identify each other. By ensuring that `Bank B` cannot identify the blockchain accounts `Bank A` uses for operating with its customers, this scenario is avoided.

The privacy of customers ensures that the customer's identity and personal details and documents the customer submitted for the core KYC process cannot be compromised. In addition, it needs to be ensured that a financial institution cannot identify a customer based on their customer profile. The distributed ledger only stores a hash of the document package

of the customer. This is accessible to entities on the blockchain - financial institutions and the regulator. It does not store any documents about the customer, which guarantees there cannot be a leak of the customer's personal information. The customer holds their ID and when they want to start operating with a new institution, they give the institution this ID. Only institution that was given this ID knows the real customer's identity behind a digital profile. Otherwise, an institution can find out how many institutions operate with a customer based on the digital profile, but the identity is not revealed. The regulator is the only party that has access to the real identity of each account on the blockchain. This fulfills the privacy condition.

A.3 Irrelevance

The irrelevance condition is about establishing there is no extra incentive for any institution to either execute the core KYC or let another institution do so. When a customer approaches the first financial institution, this institution needs to execute the core KYC unconditionally to comply with the regulatory framework. When the customer approaches another financial institution, there is no certainty on whether the institution has to re-execute the process or not. There is only an existing probability that the institution has to repeat the process, but the outcome depends on a random number generator. A financial institution cannot be certain about a single customer, but when operating with multiple customers, it can estimate the number of core KYC processes it needs to execute according to the law of large numbers[13]. Similarly, a financial institution has no impact on whether another institution needs to do the core KYC process for a customer. If a financial institution refuses to do the process when required, it is unable to operate with the customer. This fulfills the irrelevance condition.

A.4 No-minting

The no-minting condition ensures that a financial institution cannot simulate having executed the core KYC verification process without actually doing so. Any time an institution has to update and/or repeat the core KYC process, there is a digital footprint left on the distributed ledger that confirms this occurred. This footprint, and thus a right to claim some financial compensation, has to be preceded by a previous request for this from the smart contract. This ensures an institution cannot claim it executed or updated the core KYC process when it was not requested to do so.

[13] `https://en.wikipedia.org/wiki/Law_of_large_numbers`

The challenge is to assure that when the core KYC has to be repeated, updated, or both, this is adequately executed. Controlling the adequacy of the process can not be directly done using the distributed ledger, because the ledger does not store sufficient details about this process so as not to violate the privacy condition. The no-minting condition is then fulfilled implicitly by an incentive mechanism. A financial institution has a clear incentive to execute the core KYC adequately in compliance with the regulatory framework. If it does not, and the regulator later unravels it, the financial institution could face serious legal and reputational consequences outlined in section 3. This is also the mechanism that is used in the current, non-DLT KYC process. Nothing prevents a financial institution from directly on-boarding a new customer, but it could face negative consequences if the customer misuses their account for illegal activities and this is found out.

We propose that the regulator could do controls of the institutions that had to execute or update the core KYC. These controls can become more efficient due to the lower number of cases where they would be required. The number of cases would decrease because the DLT-KYC scheme does not require the core KYC to be executed by each institution a customer operates with, but allows for mutual cooperation and sharing of the result of the process between the institutions.

View publication stats