Best of Two Worlds: Efficient, Usable and Auditable Biometric ABC on the Blockchain

Neyire Deniz Sarier

Member, IEEE[1]

Abstract

In [1], two generic constructions for biometric-based non-transferable Attribute Based Credentials (biometric ABC) are presented, which offer different trade-offs between efficiency and trust assumptions. In this paper, we focus on the second scheme denoted as BioABC-ZK that tries to remove the strong (and unrealistic) trust assumption on the Reader R, and show that BioABC-ZK has a security flaw for a colluding R and Verifier V. Besides, BioABC-ZK lacks GDPR-compliance, which requires secure processing of biometrics, for instance in form of Fuzzy Extractors, as opposed to (*i*) storing the reference biometric template a_{Bio} in the user's mobile phone and (*ii*) processing of biometrics using an external untrusted R, whose foreign manufacturers are unlikely to adjust their products according to GDPR.

The contributions of this paper are threefold. First, we review efficient biometric ABC schemes to identify the privacy-by-design criteria for them. In view of these principles, we propose a new architecture for *biometric ABC* of [2] by adapting the recently introduced *core/helper setting* of [3]. Briefly, a user in our modified setting is composed of a constrained core device (a SIM card) inside a helper device (a smart phone with dual SIM and face recognition feature), which -as opposed to [1]- does not need to store a_{Bio}. This way, the new design provides *Identity Privacy* without the need for an external R and/or a dedicated hardware per user such as a biometric smart card reader or a tamper proof smart card as in current hardware-bound credential systems. Besides, the new system maintains minimal hardware requirements on the SIM card -only responsible for storing ABC and helper data-, which results in easy adoption and usability without loosing efficiency, if recently introduced key derivation scheme of [4] and the modified ABC scheme of [2] are employed together. As a result, a total overhead of 500 milliseconds to a showing of a comparable non-biometric ABC is obtained instead of the 2.1 seconds in [1] apart from the removal of computationally expensive pairings. Finally, as different from [1], auditing is achieved via Blockchain instead of proving in zero-knowledge the actual biometric matching by the user to reveal malicious behavior of R and V.

Keywords: Identity Privacy, Blockchain, Brands' DLRep, Multi-show Unlinkability, Attribute Based Credential (ABC), Non-transferability, Fuzzy Extractor, face biometrics, user-centric, KYC, GDPR

1. Introduction

Currently, smart-city-related projects are ongoing in various countries. Already in 2017, Inspur, the Chinese cloud computing company formed an alliance with IBM, Cisco and Ericsson to provide smart city solutions. However, recent studies on this topic [5, 6, 7, 8] handle the infrastructure involved in smart-city development (data centers, cameras, sensors and other devices for facial recognition, etc.) from a security perspective by considering the global ambitions of the exporting countries of this technology. For instance, major European companies already have constructed smart cities with the help of Huawei [8], which installed the camera network and other infrastructure for video surveillance systems. In this context, many US and Chines companies developed facial recognition systems integrated into smart city projects that they export to EU and non-EU countries for public security applications. However, there are concerns over the big data that those companies collect, especially when they do not follow the EU regulations on data privacy, namely GDPR [8]. Another example is faceprints of US users collected by TikTok [9], which may transmit them to its servers or data centers outside of USA for storage and/or processing [10]. Thus, sensitive data such as biometrics should be processed on each user's local device in a user-centric manner instead of outsourcing it to external untrusted sensors/devices that may lead to a similar problem present in e-voting [11], namely Secure Platform Problem (SPP) [2].

In this context, the same problem arises in biometric-based non-transferable Attribute Based Credential (biometric ABC) schemes of [1]. Recently, [1] proposed two new generic constructions, where the latter tries to re-

[1]The author is an Assoc. Prof. Dr. of Computer Science and CTO @ BioIDchain (e-mail: denizsarier@ieee.org)

[2]Due to the complexity of the voting task, it is delegated to an external/untrusted voting device leading to SPP, i.e. the untrusted voting computer issue, tackled by Code-voting approach [11].

duce the level of trust to the independent biometric device/sensor/reader R, which is assumed as semi-trusted and deployed in the same premises of the untrusted Verifier V. The origins of ABC can be traced back to 1980s, initially named as Digital/Anonymous Credentials that allow entities/users to get digital credentials from an issuer and to prove ownership of attributes encoded in the ABC to verifiers without revealing any other data. For human credential holders, an ABC contains a number of attributes such as name/surname, age, address, gender, etc. in addition to a biometric attribute either obtained from a fuzzy extractor [12, 13, 2, 14] or in form of a commitment to a biometric template [15, 1]. In any case, an ABC composed of those attributes support minimum disclosure, i.e., only necessary/essential information required for the specific application is revealed. Thus, attributes that are selectively shown lead to strong authentication and privacy even in recently introduced identity management systems built on top of public/permissionless Blockchains such as Bitcoin [2] or Ethereum [16, 17, 18].

Finally, it is crucial for biometric ABC systems to guarantee *Identity Privacy*, namely the privacy of the link between the user's identity (i.e. name/surname) and his/her biometrics, although biometrics is assumed as public data [19]. At the same time, service providers should be assured that users possess valid credentials from an issuer when they authenticate successfully to service enablers, but the credential generator (an authority different than the issuer) and a verifier/provider/enabler (even if they collaborate) could not link the *original* credential of a user to a specific ABC showing.

1.1. Intuition for the security flaw in [1]

In any biometric authentication system, there exists internal/external adversaries that try to reveal the *private* link between the user's identity and his/her biometrics. As opposed to biometric setting, for on-chain applications such as Decentralized Finance (DeFi), where access control is legally mandated to counter illicit activities, it is required to restrict the service to Know-Your-Customer (KYC)-verified users [18]. A common KYC calculation is 2+2, where a minimum of two customer identity attributes (name, surname or address, etc.) is required to be *non-private* for KYC compliance in order to reach an appropriate level of trust, namely a consistent identity score similar to a credit score [20].

Hence, to protect the privacy of the link between the (minimum) KYC identity attributes and biometrics, the biometric verification system either (1) hides the identity, i.e. the users submit their request *anonymously* without any KYC identity attribute involved, or (2) the system aims only for hiding the biometrics, where the users process and submit their biometrics *as encrypted* by using a (potentially homomorphic) scheme pseudonymously, or (3) the users process their biometric data and submit their request both *as encrypted and anonymously*. Most of the biometric verification protocols aiming for high security levels anticipate to obtain from a user an encrypted biometric template processed by the user himself.

Assume that the selective disclosure of BioABC-ZK [1] requires the user to reveal two of his KYC identity attributes a_1, a_2 associated to his/her Name+Surname for KYC-compliance, where a_{Bio} is the biometric attribute that ties the credential to the entity [1]. Then, one of the well exploited notions for secure biometric authentication, namely, *Identity Privacy* that was introduced in [21] and further investigated in [19, 22, 23, 24] for different use-cases, cannot be guaranteed for a colluding Reader R and Verifier V, where the semi-trusted external sensor R of BioABC-ZK [1] captures the raw biometric data and generates the biometric template of the user for V. The untrusted V is responsible for credential verification, and checking the predicates over the attributes depending on the access policy (i.e. for KYC-compliance), and performs the computations of a traditional showing [1].

Therefore, BioABC-ZK [1] cannot simultaneously control the data leakage resulting from the selective disclosure (a traditional showing of a credential to a verifier for the disclosed attribute(s) of Name+Surname) and from the biometric measurement on an external untrusted sensor/reader device R produced by a foreign company that may outsource the latter to the cloud, i.e. for biometric matching operations. Specifically, a colluding Verifier V and Sensor R can associate the facial biometrics -captured and processed- by the semi-trusted R to the selectively disclosed Name+Surname attribute(s) of the user, thus breaking the identity privacy notion, i.e. the link between the identity and biometrics of the user. The scenario worsens if both sensor R and verifier V involve devices manufactured by the same foreign company that does not necessarily adjust its products according to GDPR [8]. One can even deduce an *Impossibility Result* from this worst-case scenario: No two-party (i.e. colluding R and V) biometric ABC protocol that guarantees identity privacy exists.

1.2. Related Work

Apart from financial services/transactions, Blockchain technologies are employed in various sectors ranging from e-health [24] to e-voting [11] and related e-government applications. Thus, in this work we are going to focus on efficient biometric-based non-transferable Attribute-Based Credentials (biometric ABC) with off-chain verification on the Blockchain to achieve cheap, scalable, auditable, usable and practical access control.

1.2.1. Previous work on efficient Attribute-Based Credentials (ABC)

The focus of the paper is efficient ABC schemes that rely on lightweight Non- Interactive Zero-Knowledge (NIZK) arguments such as Sigma protocols. Also, we consider only the traditional setting where the verification is typically done off-chain by a single machine resulting in minimal user/prover costs.

With U-Prove being the most well known representative, Brands Credentials (also denoted as digital credentials) [25] are currently the most efficient credential scheme as shown in [26]. However, Brands' credential system [27, 25] has a serious limitation: showings that are performed more than once can be linked, namely, a Brands ABC can only be used once in an unlinkable way. Despite this disadvantage, it has recently been found real-world applications in Google, Facebook, etc. [3] and recently, in the form of e-voting [11] to prevent double-voting. However, as observed in [28], Brands' schemes allow for efficient issuance of multiple credentials on the same attribute list, where a recent application following this approach is presented for biometric-based identity management on the blockchain in order to achieve auditing, revocation, thaw/suspension, non-transferability of credentials in IIoT/smart industry [2].

1.2.2. Previous work on biometric-based non-transferable credentials requiring a dedicated trusted device

To prevent credential share/lending of credentials, U-prove proposes to split the certified attributes between the smart card and the user's device to force the smart card's involvement. Apart from U-Prove, biometric-based solutions requiring a dedicated trusted device, -e.g. a smart card, which is trusted to capture and process fresh fingerprints upon each credential presentation, and then prove that the measured fingerprint indeed matches the one *encoded* inside the credential- are described in [12, 13, 2, 14]. As opposed to [29, 30, 15], privacy-preserving biometric approaches in Attribute Based Credentials (ABC) do not employ directly embedded biometrics that is stored in each user's tamperproof smartcard [12, 13, 2, 14] resulting in non-transferability, biometric privacy, and GDPR-compliance [3]. The latter two are guaranteed even if the smart card is lost/tamper-proofness is eliminated as biometrics is not required anymore to be used and divulged. For instance, fuzzy vault for fingerprints is evaluated as a secure sketch construction and the anonymous credential scheme of [12] is based on a fuzzy extractor denoted as BKG built from this secure sketch scheme [14].

1.2.3. Previous work on biometric-based non-transferable credentials requiring an external sensor device

In this category, Adams [15] proposed the first efficient ABC with a focus on non-transferability, where the biometrics sensor captures, encrypts and commits to the fresh biometrics before returning the computed values to the verifying authority. Next, the entity computes a Zero-knowledge Proof of Knowledge (ZPK) to prove that the biometric template stored in the credential match the fresh biometric features in the freshly computed commitments sent by the sensor. To provide an efficient system, [15, 13] both rely on the one-show credential approach of Brands, which is also employed in Microsoft's U-Prove [31, 30].

Similar to Adams et al. [15], the authors of [1] recently described BioABC-ZK, the latter of the two generic constructions specifically designed for access control in public transport, restaurants or events, where identification and linkability of users is undesirable. In order to avoid any sensitive biometric data to be leaked to the verifier, the second generic design of [1] splits the verifier into: (i) a semi-trusted sensor device capturing and measuring a user's fresh biometrics Bio_f, and (ii) the untrusted service provider acting as a verifier. Once the user's biometrics Bio_f is measured, this device sends the necessary data to the user and/or the service provider acting as a verifier, and the user computes the ZPK proving that she possesses a credential matching two similar biometrics: the reference template as an attribute a_{Bio} within the credential stored at the user application, i.e. mobile phone, and the fresh template Bio_f. Since the external sensor is semi-trusted, [1] employs a sensor device with minimal operations both from hardware and software aspects to achieve the maximum security and privacy level.

Briefly, [1] defines a bb-ABC system using four actors: An external untrusted reader device R, a user U, a verifier V and an issuer I. R and V are deployed in the same premises and assumed to behave maliciously. Thus, [1] introduces a construction BioABC-ZK that adds auditing capabilities to detect (malicious/inconsistent) behavior/decision of R/V, respectively. BioABC-ZK tries to reduce the trust in R, as R is responsible for recording U's fresh biometrics in addition to the actual biometric matching. Since the user's mobile phone receives the fresh biometric data from R together with a commitment to it and the opening value, the verifier only gets the commitment from R. Next, U proves to V in zero-knowledge that the biometric data attached to his/her credential matches the fresh one. The implementation is based on face biometrics using the template generation scheme of [32] and Pointcheval-Sanders (PS) signatures, where the latter was already suggested in a previous work of [14] in the replacement of the inefficient CL-Credentials employed in [12]. Finally, [1] assumes U as mobile, R as embedded and V as a normal computer being the most powerful among them.

1.2.4. Recent Work on Blockchain based ABC and identity management

For a short summary on non-private anonymous credentials for on-chain verification, where the Certificate Authority (CA) can link users to their wallets when the CA issues an on-chain credential to a user nominated wallet, the reader is referred to [18], which considers on-chain verification of centrally issued ABC by a smart contract with the goal of minimizing the cost of verification given the extreme cost of smart contract execution. The same paper also presents a comparison for the current state-of-the-art anonymous credential systems on Ethereum Blockchain.

[3] Biometrics is considered as sensitive data according to GDPR that requires template protection techniques with provable security (such as *Fuzzy Extractors*) so that leakage of this data is prevented and privacy of biometrics is maintained [2, 14]

Besides, for a short summary of efficient digital credentials on the Bitcoin Blockchain, the reader is referred to [2, 14], which is the path we will follow in this work due to the recently introduced approach in a different setting: Similar to the identity management on the Bitcoin (BTC), exchanging digital assets such as NFTs over the BTC Blockchain is generally avoided because of the high Transaction (TX) fees and deficient programmability. However, recent efforts managed to design a NFT scheme where trades are settled in a single BTC TX contrary to executing complex smart contracts [33]. Although the authors describe their NFT scheme for Bitcoin (BTC) by fixing the TX size to the minimum BTC TX size of 226 bytes, their techniques are essentially independent of the fundamental blockchain technology since the majority of the work occurs off-chain similar to the biometric-based identity management and credential systems of [2, 14].

1.2.5. Evaluation of related work

Since the focus of this paper is on efficient ABC with biometric attributes for non-transferability, we do not consider the remaining categories, namely CL-based [34, 35, 36, 12] and PS-based [16, 17] ABCs, some of which rely on computationally expensive bilinear group operations that are not efficiently supported on Ethereum Virtual Machine (EVM) [18]. Similarly, Hardware Security Modules and embedded secure elements generally do not back up CL-based or BBS+ signatures but only more common signature schemes, for instance, ECDSA [37]. Besides, on-chain verification of Coconut is equivalent to 176 USD on Ethereum in May 2023 [18]. Similarly, BASS is a subsequent work that adds revocation to Coconut since Coconut [16] does not provide auditability, traceability as well as revocation [18]. Even though [18] outperforms the previous PS-based ABCs on the Ethereum Blockchain, the cost of a single on-chain verification of 12USD [18] cannot be assumed as reasonable given the extreme cost of smart contract execution [38]. Besides, [18] is based on more computationally costly general-purpose ZKP, for instance, a Groth16 zk-SNARK [37], thus, requires bilinear pairings and more importantly, a Layer-2 (L2) solution that offloads computation and storage to a more scalable Layer-2 network, which is simply an untrusted server employed to reduce on-chain verification costs. Therefore, credential systems for account-based blockchains with on-chain credential verification and systems for permissioned blockchains are out of scope.

1.3. Motivation

Based on the previous review and focusing on the recently introduced biometric ABC (with a traditional showing) of [1], we define the privacy-by-design criteria for efficient, cheap, scalable, auditable, usable, practical and non-transferable biometric ABC on the Blockchain.

- Even though dedicated hardware per user (i.e. smart card with integrated biometric sensor) does not scale and degrades the usability [1], a user with a (constrained) smartphone should not cause biometric ABC systems to compromise privacy over usability.
- Biometric ABC systems with a traditional showing should guarantee GDPR-compliance and identity privacy in addition to multi-show unlinkability.
- Biometric ABC systems with a traditional showing should avoid storing any biometric attribute a_{Bio} as well as the essential/minimum KYC attributes (such as name, surname, etc.) within the attribute list $\boldsymbol{a} = (a_1, ..., a_n)$ of the credential.
- Biometric attribute(s) should be encoded as a private attribute(s) that is never going to be revealed to any party but its existence guarantees the non-transferability of the credential during the showing.
- If core/helper setting is employed, the helper device (i.e. smart phone) that – captures the raw biometrics, extracts the features, generates the template of the user – is assumed to delete any biometrics once it is finished with all of the computations and returns only the mandatory data.
- If core/helper setting is employed, the system should be efficient in practice and the computational overhead of the core device (SIM card) should be independent of the number of attributes in the credential.
- Malicious behavior of server-side (i.e. external reader/ sensor device R and verifier V) should not be audited and revealed (with extensive computations as in [1]) by the (constrained) user.
- If malicious/inconsistent behavior of server-side is audited through Blockchain, minimal prover costs should be achieved with offchain verification.
- When upgrading the user's smart phone, re-generation of the (original) ABC together with biometric attribute(s)/helper data should be avoided.
- ABC suspension/update/renewal/thaw and revocation should be possible with the same infrastructure.
- ABC protocol should be independent of the underlying blockchain technology similar to the recent proposals [2, 33, 11] designed for completely different settings/use-case scenarios.

For the first item, section 1.1 reveals the consequence of processing sensitive data such as biometrics of users via external readers/sensors right before the traditional (selective) showing of a credential to a colluding Verifier. By this means, the system of [1] is (i) non-private, i.e. it reveals the link between users and their biometrics to the authority granting access, and (ii) it requires additional trust assumptions as a result of relying on an external sensor device to capture raw biometric data before further

processing. To remove additional trust in the latter, we propose an intermediate solution following the approach of [3] in this work. Thus, a new architecture is necessary to achieve the notions presented in the second item.

Considering fuzzy extractors in relation with GDPR-compliance, the recent work of Rathgeb et al. [4] and Hanznik et al. [3] disproved the claim of [1] stating that *producing deterministic outputs from a user's biometrics via fuzzy extractors is not yet feasible.* Besides, the storage requirements of the helper data, and more importantly, the accuracy levels of fuzzy extractors are not considered as a drawback anymore due to the recent results of [4] on face biometrics. Specifically, [4] lists efficient decoding times as low as 100ms for the case of 4 intervals and polynomial sizes ≤ 512, whereas for the case of 8 intervals and polynomial degrees < 1200, [4] achieves decoding times < 500ms. Also, [4] retrieves a key from face biometrics in real time for relevant parameters of False Non-Match Rate $< 1\%$ at a False Match Rate of 0.01%, which is essential for the usability. Hence, a system based on the face fuzzy vault construction of [4] results in a total overhead of 500ms to a presentation of a comparable non-biometric-based ABC.

Moreover, third and fourth items helps to reduce the information leakage through practical measures. Finally, standard assumptions in secure biometric authentication systems (such as liveness detection and immediate erasure of sensitive data once its processing is completed) should also hold for biometric ABC.

For the remaining items, the main concern is the usability, efficiency and practicality of the design. Firstly, Hardware Security Modules and embedded secure elements generally do not support CL-based or BBS+[4] signatures but only more common signature schemes such as ECDSA [37]. Thus, there is a research gap for biometric ABC depending on more efficient crypto-primitives suitable for those devices with limited computational capacity. For instance, auditing of malicious behavior of an external reader device R and/or a malicious verifier is performed in [1] by the (constrained) user at the cost of additional zero-knowledge proof for the inner product resulting in 602 multi-exponentiations and an additional range proof resulting in 200 Pedersen commitments of bits calculated with 200 exponentiations (exp)s + 200 exps for the "real" OR branches, and finally, 200 Pedersen commitments for the simulated OR branches.

To reduce the computational load of the user, auditing can be performed via Blockchain, but, this solution brings its own problems: As in many papers including the recent work of [18], the extreme cost of smart contract execution on Layer-1 Blockchain, i.e. Ethereum is tried to be solved through employing untrusted servers built as a Layer-2 network. Thus, one can employ a hybrid design that minimizes the prover's cost with the help of additional Service Enablers $\mathcal{SE}$s as in current user-centric identity management schemes on the BTC Blockchain [2, 14, 39].

Finally, the last item in our principles for biometric ABC involves a generic design, where the architecture of the protocol is blockchain agnostic. In fact, biometric ABC (with an abstract model) can be implemented using one of the many blockchains with an OP_RETURN-like opcode/transaction. The reason for choosing a Bitcoin-like blockchain is to avoid (unnecessary) smart contracts similar to the recently introduced NFT scheme of [33] where trades are settled in a single BTC transaction contrary to executing complex smart contracts. Currently, NFT marketplaces are designed on top of blockchains with smart contract functionality or Layer-2 solutions due to the lack of programmability for fair exchanges and high transaction fees in Bitcoin. However, the recent work of [33] disproved this wrong assumption and the trade is settled by publishing a single Bitcoin transaction on-chain, contrary to deploying complex smart contracts. The main bulk of work happens off-chain with a practical mechanism for token authenticity resulting in a flexible token authorization system. Following the same approach, we build a new architecture based on the same (but *modified*) ABC of [2] using the *adapted* core/helper setting of [3].

1.4. Contributions

In this paper, we describe a novel biometric ABC protocol that tries to achieve the items of the *privacy-by-design* principles by combining best of two worlds: Adapted core/helper setting of [3] and modified Blockchain-based identity management system of [2] according to the former. Based on a direct instantiation of the recently introduced deep face fuzzy vault of [4], a total overhead of 500 milliseconds to a showing of a comparable non-biometric-based ABC is obtained instead of the 2.1 seconds computed in [1], apart from the removal of computationally expensive pairings. A brief comparison is presented as below.

Table 1: Estimated biometric-based ABC showing times. †: PS-MS of [40], ‡: for U-Prove computed in [26], *: based on [32], **:based on [4]

	Underlying Signing Scheme (σ)	Approx. time of ABC based on σ	Overhead for face
[1]	PS [41]	27.58ms† [40]	2.1s* [1]
New	Brands DLRep [27]	4.38ms‡ [26]	500ms**[4]

To the best of our knowledge, the proposed biometric ABC system is the first scheme that is efficient, non-transferable, secure, transparent, scalable, usable, auditable, user-centric, GDPR-compliant, and most importantly, guaranteeing *Identity Privacy.* Although our system is instantiated on Bitcoin (BTC) Blockchain, the system is blockchain-agnostic: Our system can be built upon any public blockchain, which outperforms BTC with respect to throughput, security, privacy and usability, as the proposed framework is generic and independent of the blockchain

[4]BBS is first suggested by Boneh-Boyen-Shacham (BBS) and related anonymous credentials designs are known as BBS+[37].

platform. Lastly, even though our solution does not require re-generation of credentials when upgrading the user's smart phone, if the alternative framework of Figure 4 is employed, re-issuance of updated credentials are also avoided at the cost of decreased usability.

1.5. Preliminaries

In this section, we summarize the biometric-based non-transferable ABC scheme of [2] by adapting it according to the modified version of core/helper setting of [3]. Another primitive employed in the new proposal is a Fuzzy Extractor (FE). In fact, fuzzy vault is evaluated as a fuzzy (i.e. secure) sketch [12, 14], where the latter implies a FE, since a secure sketch is the building block of a FE [12, 42]. The reader is referred to [42, 19, 14, 12, 4] for the details of biometric cryptosystems and to Appendix for the definition of FE, Brands' DLRep, core/helper setting and the original IIoT identity management protocol of [2].

1.5.1. Background on Face Biometrics

As noted in [2, 14], most of the credential systems (including ABC) do not guarantee true non-transferability, since nothing prevents lending of the smartcard storing a credential, as in the case of lending of credit cards or other credentials. Binding the ABC to the entity/user by means of biometrics such as face, fingerprints, etc. is an effective and practical solution against credential transfer. Thus, biometric-based ABC requiring possession of the ABC owner's face biometrics on the fly ascertains that entities are physically present when their ABCs are used, hindering credential sharing and abuse by theft.

In this context, removing the need for smartcards and special biometric card readers by replacing them with an external Reader/Sensor device R, biometric ABC schemes proposed in [1] are based on the face verification system of MSBSIF-SIEDA [32] that achieves an accuracy of 94.63%, approaching an accuracy of 100% at 0.2 false alarm, i.e. false positive rate (FMR). However, current research in Deep Convolutional Neural Networks (DCNN) shows significant improvements in facial recognition accuracy, which is confirmed by the recent work of [4] that achieves False Non-Match Rate (FNMR) below 1% at a False Match Rate (FMR) of 0.01%, i.e. $<$ 1% FNMR at 0.01% FMR. For the same FMR=0.01%, [32] denotes Genuine Match Rate (GMR = 1 - FNMR) as true positive rate and achieves an accuracy of slightly above 85% (upper lefthand side arrow), hence a false non-match rate below 0.15, i.e. 15% as shown in Figure 1. Thus, for privacy-preserving access control, the system of [4] is a better choice compared to [32] since the former offers additional key derivation feature which can be used for secret key management using biometrics, and thus, in combination with other disclosed (private) attributes of a digital/anonymous credential.

1.5.2. Background on the core/helper setting

The notion of core/helper anonymous credentials is introduced in [3], which considers a constrained core device

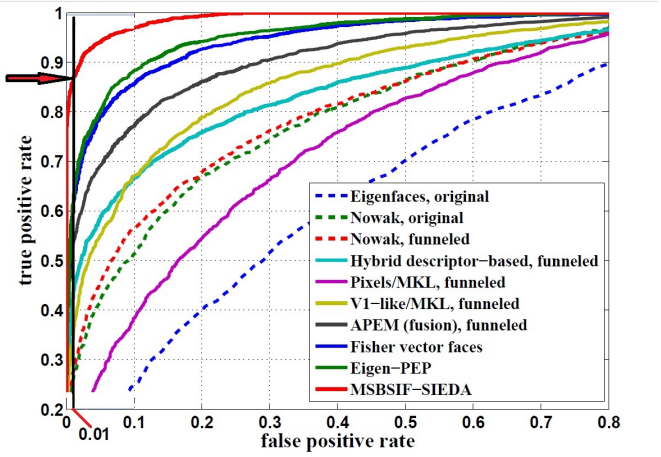

Figure 1: ROC curve of MSBSIF-SIEDA and other state-of-the-art methods [32]

(a SIM card) and a powerful helper device (a smartphone), where the former performs operations regardless of the number of credential attributes and the latter is unable to use the credential without the help of the SIM card. Today's PCs, smartphones are generally equipped with secure elements in form of dedicated hardware modules, e.g. the Trusted Platform Module (TPM) or SIM cards that are designed to handle secrets, i.e. their creation and storage. Initially, we only require a SIM card that is responsible for the storage of an ABC generated by the government. The reader is referred to Appendix A for the details of [3].

1.5.3. Background on the biometric based ABC of [1]

As stated in [1], the authors of [12, 29] proposed solutions depending on a dedicated trusted device carried by each user, such as a smart card, which is trusted to capture, scan and measure fresh fingerprints for each ABC showing. Besides, their system requires the card to prove that the encoded template stored as part of the ABC matches the fresh biometrics. However, as discussed in section 1.3, the claim that *"a solution requiring dedicated hardware for each entity does not scale and suffers from the same usability limitations as device-bound ABCs as well"* is disproved. Also, the instantiation of BioABC-ZK for face biometrics with a template length of N = 600 requires Pointcheval-Sanders (PS) signatures in a bilinear group, Pedersen commitments and authenticated AES. The ABC showing token is a Schnorr-style ZPK (Σ-protocol), which is transformed using Fiat-Shamir heuristic to obtain a non-interactive protocol. The authors of [1] measured values for the tasks with high costs executed by each of the actors during an ABC showing, i.e. the calculation of Pedersen commitments in the reader device R, and the tasks of the ZPK for the entity and the verifier. Here, a_{Bio} refers to a face template within a credential bound to it, namely, face biometrics represented as an attribute as part of the ABC. This instantiation adds an overhead of approximately 2.1s to a presentation of a comparable non-

biometric-based ABC. Hence, the total overhead is just over 3 seconds. The reader is referred to Appendix C for an overview of the biometric ABC of [1].

2. Adapting the IIoT Identity Management protocol of [2]

To describe our new biometric-based non-transferable ABC (biometric ABC) scheme, we first slightly modify the core/helper setting of [3] so that it is directly applicable to our use-case/scenario.

2.1. Modifying the core/helper setting of [3]

Initially, the core, namely SIM card$_1$ stores (1) the governmentally generated and signed (encrypted) credential c_1, where user-specific X_0 and non-essential attributes denoted as *params*, the helper data P, the associated original credential h, encrypted credential c_1 together with the government's signature σ on c_1 is permanently stored in the SIM card$_1$; Having the two SIM cards installed, the helper device, namely the smartphone can now obtain a credential $h_{\mathcal{E}}$ (computed as in [2]) from the issuer $\mathcal{CIV}$ by sending a request (2), where SIM card$_1$ is informed about a credential issuance via (3), which returns the necessary data to the smartphone so that it can compute $h_{\mathcal{E}}$ together with the issuer. Next, the issuer $\mathcal{CIV}$ computes the secret randomization parameter AP, the secret credential update parameter UP as in [2], which are then passed to the smartphone that completes the computation of $h_{\mathcal{E}}$, $i_{\mathcal{E}}$ together with the issuer via (4). Finally, the credential $h_{\mathcal{E}}$ and index $i_{\mathcal{E}}$ for the authentication path are stored at the smartphone together with UP, AP (5).

For a biometric ABC showing, the smartphone first triggers a request (6) that is also passed to the SIM card$_1$ (7) so that it returns the helper data P to the smartphone for the computation of the biometric attribute using the fresh biometrics captured and extracted by the smartphone. Next, based on the attributes that are required to be shown selectively (as all the other attributes remain undisclosed), the smartphone computes the necessary steps of the showing protocol (8) and sends the computed values to the verifier (9) who either accepts or rejects. Finally, the smartphone computes and stores the updated credential and authentication path (10) to be used in the future authentications. The same computations are performed by the issuer $\mathcal{CIV}$ via (11) which also updates the Merkle Tree for the users who successfully authenticated. An overview of the modification is in Figure 2(b).

Remark 2.1. *We note that the above presented modification to the original core/helper setting of [3] is designed for a SIM card with very limited capabilities, namely a core device equipped only with a small memory. However, in section 5, we extend the capabilities of the core device so that SIM card$_1$ can also compute the (hidden) biometric attribute X_1 using the data/template extracted from the fresh biometrics captured and transferred by the smartphone to the SIM card$_1$ at step (3). Hence, instead of sending the helper data P during a credential presentation, the SIM card$_1$ returns the freshly computed biometric attribute X_1 at step (7) so that the user enters the only missing essential KYC ID attributes (such as Name and Surname) at step (8) as they are not stored as part of the credential.*

In any case, the private attributes such as the biometric data of the entity are not revealed to any party in the system.

Remark 2.2. *Moreover, a further extension to the capabilities of the core device described in Remark 2.1 allows for the computation and storage of the update parameters UP and AP, which requires only two hash computations (as summarized in section 3.3.1) at step (7). Together with the freshly computed attribute X_1 and the fresh update parameters, the SIM card$_1$ returns the stored credential data and path data to the smartphone at step (7) so that the smartphone completes the credential showing at (9) with the verifier $\mathcal{SE}$. Thus, the smartphone is only responsible for the computation of the updated credential at step (10) using the update parameters that it received from the SIM card$_1$ at step (7). As a result, the storage duties of the smartphone is completely eliminated at the cost of a small increase in the storage and computational overhead of the SIM card$_1$. The alternative flow diagram is in Figure 4.*

2.2. Adapting the Participants of [2]

As different from [2], an entity $\mathcal{E}$ is a user possessing a smartphone as the primary helper device with two SIM-card slots, who wants to authenticate to a Service Provider ($\mathcal{SP}$). Here, one SIM-card needs to be reserved as the secure element (core) similar to the setting in [3]. As in [2], all of the parties are assumed to be independent and non-colluding:

Credential Generator ($\mathcal{CG}$) - Central Authority such as the Government that generates the digital twin of the National ID card with biometrics and other attributes defining the user's identity. The core device storing the original governmentally generated biometric ABC and its encryption signed by the government is trusted by everyone.

Credential Issuer and Verifier ($\mathcal{CIV}$) - Any organization, for instance, a bank or a company that has a present relationship with the entity. $\mathcal{CIV}$ provides justification for organizational aspects of $\mathcal{E}$'s identity. As in [2], $\mathcal{CIV}$ is designed to employ a TEE or TPM to run the automatic updates based on $\mathcal{E}$-specific data. $\mathcal{CIV}$ has no knowledge of an $\mathcal{E}$'s complete identity record due to encryption. To achieve multi-show unlinkability against $\mathcal{SE}$, $\mathcal{CIV}$ only records a blinded version of biometric ABC on the Merkle Tree, updates both its location on the tree and its value by randomizing it after a successful credential show. Thus, $\mathcal{CIV}$ does not need to be continuously online resulting in reduced infrastructure costs [2]. Two different platforms take place in our proposal: BTC blockchain for on-chain

storage and a public/private IPFS for offchain data storage of large amount of data using a Merkle tree in order to use the public ledger functionality for integrity of data as in [14, 24, 39, 2].

Service Enabler(s) ($\mathcal{SE}$) - responsible for verifying the records for each $\mathcal{E}$ created by various $\mathcal{CIV}$ and transferring data about these records to $\mathcal{SP}$. The public key $pk_{\mathcal{SE}}$ of $\mathcal{SE}$ should be well-known. $\mathcal{SE}$ and $\mathcal{CIV}$ are (as emphasized before) assumed to be non-colluding.

2.3. Trust Assumptions and Adversarial Model

Each participant is ideally assumed as independent and non-colluding. Except for $\mathcal{CG}$, the remaining authorities are assumed as semi-honest although malicious behavior of some of the authorities are analyzed in section 4.

$\mathcal{CIV}$ is designed as in [2] and only publishes the randomized commitments to credentials into the Blockchain based on the encrypted h issued and justified by $\mathcal{CG}$. As in [2], $\mathcal{CIV}$ has no knowledge of $\mathcal{E}$ except for the X_n, i.e. organizational attribute(s) and minimum KYC ID attributes required to register the user/entity to the organization.

If $\mathcal{E}$ runs the showing protocol to affirm that his/her identity satisfies the requirements determined by $\mathcal{SP}$, some of the X_is ($i > 1$) are revealed to $\mathcal{SE}$. As in [2], $\mathcal{SP}$ does not learn anything about $\mathcal{E}$ registered by the $\mathcal{CIV}$, but $\mathcal{SP}$ is assured that $\mathcal{E}$ satisfies the necessary properties for the service through the intermediary of $\mathcal{SE}$. As in [2], multi-show unlinkability against $\mathcal{SE}$ or their collusion is guaranteed by randomizing both the index $i_{\mathcal{E}}$ and the credential $h_{\mathcal{E}}$ in the Merkle tree following each successful authentication of $\mathcal{E}$ according to the steps in Algorithm 6. Thus, a malicious $\mathcal{SE}$ cannot associate disclosed attributes to a specific $\mathcal{E}$, since there could be many $\mathcal{E}$s that have overlapping/common attribute(s). Besides, the core device of $\mathcal{E}$ is trusted by everyone, thus, collusion of $\mathcal{E}$ and $\mathcal{SE}$ is unrealistic. Also, $\mathcal{SP}$ must rely on $\mathcal{SE}$ to accurately pass the necessary data, namely $\mathcal{E}$ meets the requirements for the service. Finally, distinct $\mathcal{SE}$(s) allow $\mathcal{E}$ to switch to a different $\mathcal{SE}$ for each authentication. Showings can be performed with different $\mathcal{SE}$s without being linkable to each other as in [2].

3. The New Construction: Biometric ABC

As opposed to previous constructions such as [15] that combine the digital credential scheme based on a Blind Signature with biometrics encapsulated as a Pedersen commitment, we directly encode the biometric attribute X_1 into the Brands' DLRep as in [2, 14] so that this private/hidden attribute is not revealed to any actor but its presence guarantees non-transferability of the biometric ABC during the showing/presentation protocol with $\mathcal{SE}$. The abbreviations adapted from [2] are shown in Table 2. An overview of the new construction is in Figure 2.

Remark 3.1. *An identity ID_k defining an entity $\mathcal{E}_k$ denotes the attributes $(X_1, , ..., X_{n-1})$ on the National ID card of $\mathcal{E}_k$, where the first attribute is assigned for the biometrics such as faceprint, fingerprint, etc.. Also, let $h_{\mathcal{E}}$ denote its digital version, where X_1 corresponds this time to the digital version of the biometrics extracted using a fuzzy extractor build upon the deep face fuzzy vault scheme of [4]. The associated helper data P is a user-specific secret parameter of this fuzzy extraction process. We assume that h is generated by the government (credential generator $\mathcal{CG}$), where X_0 is a user-specific secret parameter listed under the params together with the non-essential attributes (i.e. some of X_is).*

The digital version of the National ID card is delivered in form of a DLRep h as computed in Appendix B, section 8.0.1. Next, an ElGamal encryption/Pedersen Commitment is applied on h and the encrypted/committed value c_1 is also signed by the $\mathcal{CG}$. Finally, the secret parameters (P, h, c_1) and the governments signature on the encrypted credential c_1 are stored on the SIM Card.

Table 2: Abbreviations adapted from [2]

$\mathcal{CG}$, $\mathcal{CIV}$	Credential Generator, Credential Issuer
$\mathcal{SE}$, $\mathcal{SP}$	Service Enabler, Service Provider
params	User-specific X_0 and non-essential attributes
P, X_1, h	Helper data, Biometric attribute, Credential
c_1, σ	Encryption of h, $\mathcal{CG}$'s signature on c_1
$\mathcal{E}$, $h_{\mathcal{E}}$	Entity, Credential stored on the Merkle Tree
$i_{\mathcal{E}}$	Authentication Path of $h_{\mathcal{E}}$ in the Merkle Tree
AP and UP	Automatic Update parameters for $i_{\mathcal{E}}$ and $h_{\mathcal{E}}$
$r_{\mathcal{CIV},t}$	Merkle tree root commiting to $h_{\mathcal{E}}$s at time t

3.1. Credential Generation

As in [2], the original credential h of an entity $\mathcal{E}$ is generated by the government ($\mathcal{CG}$) and the resulting value h is first encrypted using $pk_{\mathcal{E}}$, namely ElGamal public key of $\mathcal{E}$ and the resulting ciphertext c_1 is signed using $sk_{\mathcal{CG}}$, namely secret key of $\mathcal{CG}$ for a Schnorr signature scheme. Alternatively, ElGamal Encryption can replaced by Pedersen commitment as suggested in [2]. The credential h, its encryption/commitment c_1 and the signature σ are stored in SIM card$_1$. Since no essential Know-Your-Customer (KYC) identifying data such as Name and Surname together with biometric template or any biometric key/attribute is stored within h, our new scheme requires a fresh biometric reading on each authentication attempt. The data on the SIM card$_1$ is issued by $\mathcal{CG}$, and is trusted to protect the $\mathcal{E}$'s interests. SIM card$_1$ assures that $\mathcal{E}$, $\mathcal{CIV}$ or $\mathcal{SE}$ cannot tamper with (or even read) the secret parameters *params* as well as the helper data P, cryptographic keys and the original credential h.

As opposed to [29, 15, 1], $\mathcal{E}$'s biometrics is not stored on any device such as the SIM card or smartphone, which is otherwise used in checking whether the fresh biometrics of $\mathcal{E}$ matches the one stored on the device as done in

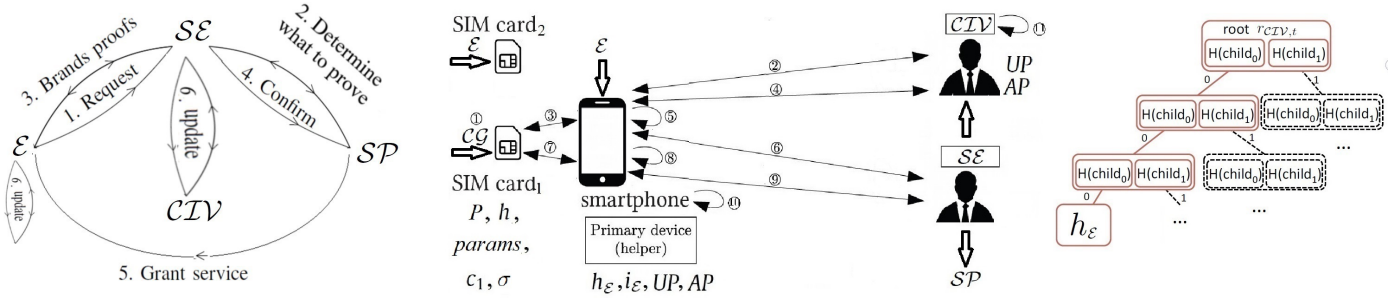

Figure 2: (a) Overview of biometric ABC (b) Overview of biometric ABC issuance and verification (c) The verifications run by $\mathcal{SE}$ to prove that $h_{\mathcal{E}}$ is a branch in the Merkle tree using the intermediate hashes of the branch provided by $\mathcal{E}$. $i_{\mathcal{E}} = 000$ represents the location of $h_{\mathcal{E}}$ on the tree. For immutability through chaining and non-membership proof of the credentials, a commitment to the root of the (sorted) Merkle tree (denoted with $r^{(s)}_{\mathcal{CIV},t}$) is published [2]

Algorithm 1: Biometric ABC Generation

Input: SIM Card, National ID and biometrics of $\mathcal{E}$, $pk_{\mathcal{E}}, sk_{\mathcal{CG}}$
Output: $ID, params, P, h, c_1, \sigma$
Request for original biometric ABC $\mathcal{E} \rightarrow \mathcal{CG}$ (National ID, $\mathcal{E}$)
Biometric Verification $\mathcal{E} \leftrightarrow \mathcal{CG}$:
-$\mathcal{E} \rightarrow \mathcal{CG}$ (Face Image of $\mathcal{E}$)
-$\mathcal{CG} \rightarrow \mathcal{E}$ (Trusted Sensor)
-$\mathcal{CG}$ uses deep face fuzzy vault to retrieve a biometric key from face template $\vec{\mu}$ and the helper data P
Assignment of user-specific data $\mathcal{E} \leftrightarrow \mathcal{CG}$:
-$\mathcal{CG}$ generates user-specific secret X_0 listed in $params$
Credential Binding $\mathcal{CG} \rightarrow \mathcal{E}$:
-$\mathcal{CG}$ runs the fuzzy extractor to compute the extracted random string R, hashes it using a cryptographic hash function H, assigns it to $X_1 = \mathsf{H}(R)$
Biometric ABC Generation $\mathcal{E} \leftrightarrow \mathcal{CG}(ID, params, P, pk_{\mathcal{E}}, sk_{\mathcal{CG}})$:
-$\mathcal{CG} \rightarrow \mathcal{E}$ $(ID, X_0, X_1, ..., X_{n-1}, g_0, ..., , g_{n-1}, g_n)$
-$\mathcal{CG}$ returns $\mathcal{E}$ original credential h on ID, its encryption c_1 using $pk_{\mathcal{E}}$, and σ, i.e. signature on c_1 using $sk_{\mathcal{CG}}$
Storage of biometric ABC (SIM Card, $params, P, h, c_1, \sigma$):
-The original biometric ABC is stored in SIM Card // (1)

[29, 1]. At the end of this phase, we assume that each $\mathcal{E}$ has a digital credential h generated, encrypted/committed and the resulting ciphertext c_1 is signed by $\mathcal{CG}$ as in [2]. Therefore, any industrial organization, bank or company has to trust both the core device and the signature σ on the encryption c_1 of the digital credential h.

3.1.1. Credential Binding

As one can observe from Algorithm 1, $\mathcal{CG}$ binds each credential to $\mathcal{E}$ using a biometric key derivation/retrieval technique based on fuzzy extractors [43, 12]. As different from [2], in this work, we employ the deep face fuzzy vault of [4], which can be evaluated as a secure sketch forming the basis of Fuzzy Extractors FE. Briefly, FE allows one to extract randomness from w to reproduce it later using a different value w' close to the original w. During the Showing Protocol, the original w is obtained with the help of P given that $\mathsf{dis}(w, w') \leq d$ and R is extracted to compute X_1. The details of Credential Binding are presented in Algorithm 1 and in Appendix A, where we assign R to X_1 that represents the biometric attribute of $\mathcal{E}$. Since the core device of $\mathcal{E}$ stores only the helper data P, data leakage from the stored original credential h about X_1 or biometrics of $\mathcal{E}$ is impossible as opposed to [1].

3.2. Credential Issuance

A summary of credential issue is given in Algorithm 2 and detailed description is presented in Appendix B.

Algorithm 2: Biometric ABC Issuance

Input: Smartphone, SIM Card, $ID, g_n, X_n, pk_{\mathcal{E}}$
Output: $h_{\mathcal{E}}, i_{\mathcal{E}}, UP, AP$, Merkle Tree
Request for biometric ABC $\mathcal{E} \rightarrow \mathcal{CIV}$ (KYC ID, c_1, σ):// (2)
-Smartphone informs SIM Card about credential issuance, which returns h, c_1, σ, P to the smartphone // (3)
Credential Verification $\mathcal{CIV} \leftrightarrow \mathcal{E}$ (KYC $ID, g_n, pk_{\mathcal{E}}, c_1, \sigma$):
-$\mathcal{CIV}$ requests the KYC ID of $\mathcal{E}$ and verifies the signature σ on c_1, encrypts the new organizational attribute X_n, i.e. $g_n^{X_n}$ using $pk_{\mathcal{E}}$ and returns c_2.
Blind Attribute Merging $\mathcal{CIV} \leftrightarrow \mathcal{E}$ $(X_n, g_n^{X_n}, c_1, c_2)$:
-$\mathcal{CIV}$ sends $c_2, X_n, g_n^{X_n}$ to $\mathcal{E}$ so that both compute/check $c_3 = c_1 \cdot c_2$ corresponds to the encryption of $h' = h \cdot g_n^{X_n}$
Blinded DLRep $\mathcal{CIV} \leftrightarrow \mathcal{E}$ (BlindDLRep):// (4)
-$\mathcal{E}$ computes BlindDLRep together with an additional Schnorr Signature as in Appendix B, section 8.0.3
-$\mathcal{E}$ returns $\mathcal{CIV}$ BlindDLRep together with a Σ protocol to prove KYC ID and a PoK computed as in section 8.0.3 $PoK\{\gamma, h' | (\beta_1 = z^{\gamma}) \wedge (\beta_2 = h'^{\gamma}) \wedge (c_3^{\gamma} \in Enc(h'^{\gamma}))\}$
-$\mathcal{CIV}$ verifies Σ and PoK, assigns the biometric ABC, i.e. $(z^{\gamma}, h'^{\gamma})$ of $\mathcal{E}$ as $h_{\mathcal{E}}$
Storage of biometric ABC on the Merkle Tree $(h_{\mathcal{E}}, i_{\mathcal{E}})$:
-$\mathcal{CIV}$ stores $h_{\mathcal{E}}$ at the position $i_{\mathcal{E}}$ on the Merkle Tree
Assignment of Update Parameters $\mathcal{CIV} \rightarrow \mathcal{E}(UP, AP)$:
-$\mathcal{CIV}$ returns $(h_{\mathcal{E}}, i_{\mathcal{E}})$ and the secret parameters, i.e. Authentication Path AP and credential Update UP // (4)
-$\mathcal{E}$ stores $h_{\mathcal{E}}$, index $i_{\mathcal{E}}$, UP, AP on the smartphone // (5)

After storing the blinded DLRep h'^{γ} of $\mathcal{E}$ on the Merkle tree as $h_{\mathcal{E}}$, $\mathcal{CIV}$ returns the secret randomization parameter for the authentication path AP to $\mathcal{E}$ together with the credential update parameter UP. The details of credential storage and publication of BlindDLReps are given in Appendix B, sections 8.1 and 8.2.

3.3. Credential Showing

Biometric ABC presentation is described in Algorithm 3, the details of which is given in Appendix B, section 8.3.

The original digital credential h produced by $\mathcal{CG}$ is stored at the SIM card, which does not store any biometric

Algorithm 3: Biometric ABC Showing Protocol

Input: $pk_{\mathcal{SE}}$, branch of Merkle tree, BTC Transaction with identifier TX_{id}, Smartphone, SIM Card, ID, X_n
Output: Grant of service or Reject
/* The entity generates a new $pk_{\mathcal{E}}$ to open the secure communications channel $\text{TLS}_{pk_{\mathcal{E}},pk_{\mathcal{SE}}}$ */
Request for service $\mathcal{E} \rightarrow \mathcal{SE}$ (Name of Service, $\mathcal{SP}$) // (6)
Prove possession of info $\mathcal{E} \leftrightarrow \mathcal{SE}$:
-$\mathcal{SE} \rightarrow \mathcal{E}$ (Info to prove: Required KYC ID attributes, $\{\mathcal{CIV}\}$, SessionID) // (6)
-$\mathcal{E} \rightarrow \mathcal{SE}$ (TX_{id}, Merkle tree branch, SessionID) // (6)
-Smartphone informs SIM Card about credential show, which returns $params, P$ to the smartphone // (7)
-Smartphone uses P and the fresh biometrics to extract the hidden biometric attribute X_1 // (8)
-$\mathcal{E}$ enters the minimum/required KYC ID attributes and prepares the first move of credential show using $params, X_1$ and the entered attributes // (8)
Credential Verification $\mathcal{E} \leftrightarrow \mathcal{SE}$ (Showing protocol run of section 8.3 in Appendix B, SessionID) // (9)
-$\mathcal{SE}$ uses the branch and the public Merkle tree root $r_{\mathcal{CIV},t}$ to check that $h_{\mathcal{E}}$ is in the tree
-If verified, $\mathcal{SE}$ communicates $\mathcal{SP}$, otherwise reject $\mathcal{E}$
Confirm for service $\mathcal{SE} \rightarrow \mathcal{SP}$: (SessionID)
Grant of service $\mathcal{SP} \rightarrow \mathcal{E}$
Confirm for update $\mathcal{SE} \rightarrow \mathcal{CIV}$: (SessionID, $h_{\mathcal{E}}$, $i_{\mathcal{E}}$)
Inform of update $\mathcal{SE} \rightarrow$ other $\mathcal{SE}$s: (SessionID, $h_{\mathcal{E}}$, $i_{\mathcal{E}}$)

Algorithm 4: Updating the index and credential after an authentication

Input: $i_{\mathcal{E}}$, $h_{\mathcal{E}}$, PRF, password$_{\mathcal{E}}$, UP, AP, counter k
Output: Updated $\{i_{\mathcal{E}}, h_{\mathcal{E}}, UP, AP$, counter $k\}$
$\mathcal{E}$ updates entity index for the kth authentication:// (10)
-If $k = 1$, compute $(i_{\mathcal{E}})^* = \text{PRF}(\text{H}^k(AP))$
-Else, compute $(i_{\mathcal{E}})^* = \text{PRF}(AP)$
$\mathcal{E}$ updates entity credential for the kth authentication: // (10)
-If $k = 1$, compute $U = \text{PRF}(\text{H}^k(UP))$ and $(h_{\mathcal{E}})^U = (\text{BlindDLRep})^U = (z^\gamma, h'^\gamma)^U$
$= (z^{U\gamma}, h'^{U\gamma}) = \prod_{j=0}^{n} g_j^{X_j U\gamma} = (\overline{z}, \overline{h'})$
-Else, compute $U = \text{PRF}(UP)$ and $(h_{\mathcal{E}})^U = (\overline{z}, \overline{h'})^U$
$\mathcal{E}$ replaces $h_{\mathcal{E}}$ with $(h_{\mathcal{E}})^U$ and $i_{\mathcal{E}}$ with $(i_{\mathcal{E}})^*$ // (10)
$\mathcal{E}$ computes the new update parameters: // (10)
-Assign $UP = \text{H}(UP)$ and $AP = \text{H}(AP)$
-Increment k by 1

Algorithm 5: Updating the index and credential of an entity after an authentication

Input: (SessionID), $i_{\mathcal{E}}$, branch of Merkle tree, $h_{\mathcal{E}}$, PRF, password$_{\mathcal{E}}$, UP, AP, counter k
Output: Updated {Merkle tree, $i_{\mathcal{E}}, h_{\mathcal{E}}, UP, AP$, counter k}
$\mathcal{CIV}$ updates entity index:// (11)
-Identical to Algorithm 4
$\mathcal{CIV}$ updates entity credential:// (11)
-Identical to Algorithm 4
$\mathcal{CIV}$ updates Merkle Tree: // (11)
-Insert a random value at the position $i_{\mathcal{E}}$
-Record $(h_{\mathcal{E}})^U$ at the position $(i_{\mathcal{E}})^*$
$\mathcal{CIV}$ updates Database entry for the Entity: // (11)
-$\mathcal{CIV}$ replaces $h_{\mathcal{E}}$ and $i_{\mathcal{E}}$ identical to Algorithm 4
-$\mathcal{CIV}$ computes UP, AP, k identical to Algorithm 4

image or template similar to the smartphone. Thus, even if the integrity of the SIM card storing h is breached, recovering the biometrics or any biometric attribute of $\mathcal{E}$ is impossible. Because of the automatic updates and change of the session key $pk_{\mathcal{E}}$ prior to each authentication attempt, each authenticating $\mathcal{E}$ appears as a first time user to $\mathcal{SE}$.

Every time $\mathcal{E}$ authenticates successfully, the helper device, namely the smart phone as well as $\mathcal{CIV}$ automatically increment a counter k, compute the newly updated ABC overwriting the present one and the newly updated authentication path on the Merkle Tree using a PRF, Cryptographic Hash Function, AP and UP as in [2]. The password shared between $\mathcal{E}$ and $\mathcal{CIV}$ for the PRF and related computations are identical to section 8.4 of Appendix B. Lastly, we count on the helper device for the correct computation of the update parameters. The details are given in Algorithms 4 and 5.

3.3.1. Updating the Credential-related Data

After $\mathcal{CIV}$ receives the message of "Confirm for update" from $\mathcal{SE}$, if $\mathcal{E}$ is not suspended, $\mathcal{CIV}$ updates the Merkle tree path and ABC of $\mathcal{E}$ as in Algorithm 5, whereas the steps performed by each entity are described in Algorithm 4. For simplicity, we assume in Algorithm 4 that a single user/entity $\mathcal{E}$ is authenticated between two Bitcoin (BTC) blocks although the same procedures are repeated for the remaining entities during the same time interval.

Recall that in Algorithm 2, $\mathcal{CIV}$ returns $\mathcal{E}$ the update parameters for the secret ABC and the authentication path, UP and AP, respectively, so that both parties can compute at the kth authentication the parameters, the new path and updated the credential. Following the Pseudo Random Function (PRF) computation, both $\mathcal{CIV}$ and $\mathcal{E}$ replace UP with $\text{H}(UP)$ (equal to $\text{H}^k(UP)$ for the kth authentication to update its value for $k = 2, 3...$) identical to the parameter AP.

During the credential issuance, BlindDLRep involves the blinding factor γ as described in section 8.0.3 of Appendix B. This way, the government $\mathcal{CG}$ and the service enablers $\mathcal{SE}$s cannot link the generation and showing phases, even if they collude. This corresponds to the *Unlinkability* property achieved in U-prove as well [31]. However, the non-colluding $\mathcal{CIV}$ can access the organizational attributes of $\mathcal{E}$. Thus, $\mathcal{CIV}$ not only updates the credential BlindDLRep via $(\text{BlindDLRep})^U$ but also randomizes the authentication path. This update mechanism following a successful authentication results in *Multi-show unlinkability* against credential verifiers ($\mathcal{SE}$ in our system), which is a lacking feature in U-prove.

After computing all the updated ABCs denoted with $h_{\mathcal{E}}^j$s in Algorithm 6, $\mathcal{CIV}$ calculates the anonymity set with size sum to check whether sum denoting the total number of authenticated entities between two consecutive blocks is larger than the system threshold τ in order to guarantee the unlinkability of those entities against the malicious $\mathcal{SE}$s. Next, as described in Algorithm 6, $\mathcal{CIV}$ publishes one of the Bitcoin transactions as shown in Figure 3 below.

Algorithm 6: Merkle Tree root computation and storage on the blockchain

Input: Randomized and Updated Branches of Merkle tree, SessionIDs, $h^j_{\mathcal{E}}$, $i^j_{\mathcal{E}}$, $cm_{\mathcal{CIV},t-1}$,$cm^s_{\mathcal{CIV},t-1}$
Output: BTC Transaction with identifier TX_{id}
Check for entity anonymity:
-Sum up the total number of authentications of $h^j_{\mathcal{E}}$s at different $\mathcal{SE}$s for epoch $[t-1, t]$
-Check whether the sum is $0 < sum < \tau$, if so, update empty branches of Merkle tree with random values, otherwise do nothing /* To provide unlinkability and untraceability of an entity against a malicious $\mathcal{SE}$ */
Update Merkle tree:
-Construct Merkle tree via updated & remaining branches
-Compute $r_{\mathcal{CIV},t}$, $cm_{\mathcal{CIV},t}$ = H ($cm_{\mathcal{CIV},t-1}$, $r_{\mathcal{CIV},t}$)
-Optionally, $r^s_{\mathcal{CIV},t}$, $cm^s_{\mathcal{CIV},t}$ = H ($cm^s_{\mathcal{CIV},t-1}$, $r^s_{\mathcal{CIV},t}$) for the sorted Merkle tree root commitments
Distribute Merkle tree:
-$\mathcal{CIV}$ publishes a TX on the Blockchain as in Figure 3.
-$\mathcal{CIV}$ distributes the Merkle tree to $\mathcal{SE}$s.

3.4. Auditing

Auditing can be performed offchain by entities and service enablers using the Merkle Tree and Bitcoin transactions as described in Algorithm 7. This way, malicious or inconsistent behavior of $\mathcal{CIV}$ can be checked both by entities and service enablers.

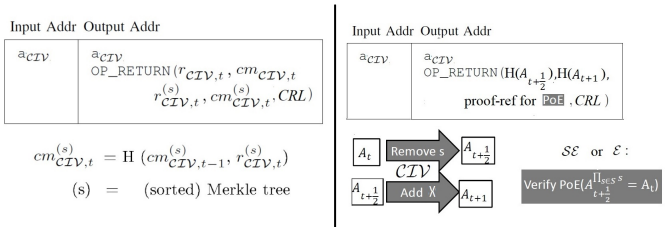

Figure 3: Bitcoin Transactions (BTC TX)[2]:
Left-hand side for Merkle Tree- | Right-hand side for Accumulator-based IIoT identity management system of [2]

Algorithm 7: Auditing

Input: Distributed Merkle Tree
Output: Deleted/replaced/updated $h_{\mathcal{E}}$s
Verification after distribution of Merkle Tree of Algorithm 6:
-$\mathcal{SE}$ can verify that $\mathcal{CIV}$ correctly deleted the used/spent $h_{\mathcal{E}}$s from the (sorted) Merkle tree by checking it using an exclusion proof in $O(\log N)$
-$\mathcal{E}$ can verify that $\mathcal{CIV}$ correctly deleted his used/spent $h_{\mathcal{E}}$ identical to $\mathcal{SE}$
Query for the current Merkle Tree $\mathcal{E} \leftrightarrow \mathcal{SE}$:
-$\mathcal{E}$ queries any $\mathcal{SE}$ to check whether $\mathcal{CIV}$ correctly updated and replaced his credential on the tree any time after an authentication

3.5. Credential Renewal/Thaw/Suspension/Revocation

Lastly, permanent revocation of an entity is performed as in Algorithm 8. The details of the Algorithms for credential renewal due to an organizational attribute change and a thaw process following a suspension are described in Appendix B, section 8.4.

Algorithm 8: Revocation

Input: Current Credential Revocation List CRL
Output: Updated CRL
$\mathcal{CIV}$ revokes an identity/entity permanently:
-Insert a random value at the position $i_{\mathcal{E}}$
$\mathcal{CIV}$ updates CRL:
-Compute $CRL^U = H(CRL) \cup h_{\mathcal{E}}$ and $CRL = H(CRL^U)$
$\mathcal{CIV}$ updates/distributes the Merkle Tree as in Algorithm 6:
-Distribute the updated list (CRL) to $\mathcal{SE}$s
-Continue identical to Algorithm 6

4. Security Analysis of the New System

First, we evaluate the BioABC-ZK [1] based on the following Lemma. The reader is referred to Appendix C for a summary of BioABC-ZK [1].

Lemma 4.1. *A protocol cannot guarantee Identity Privacy for a compromised user or colluding Reader R and Verifier V if the user stores the biometric ABC together with any essential KYC ID attributes (such as name, surname) and biometrics that are processed by those V and R, respectively.*

For the second generic construction (BioABC-ZK) of [1], Identity privacy in the sense of the above Lemma cannot be guaranteed since identity attributes are part of **b**= (a_{Bio}, **a**), where **a** = $(a_1, ..., a_n)$ and the biometric template a_{Bio} that is used during the registration/credential issuance is stored next to the credential's signature σ as shown in Figure 10 of Appendix C. For increased usability, the user of BioABC-ZK does not require to carry additional smart card, biometric card reader, etc. to store/process those sensitive data. Since the proof is trivial, we omit the details. Besides, even if a careless user has lost his phone, an outside attacker cannot obtain any essential credential data in our proposal, due to the direct application of the third privacy-by design principle in section 1.3.

4.1. Identity Privacy

In this security notion, we focus on internal adversaries, namely $\mathcal{SE}$ or colluding $\mathcal{SE}$s, who perform the offchain verification. Specifically, a malicious $\mathcal{SE}$ tries to break the privacy of the sensitive relationship between $\mathcal{E}$'s identity and its biometrics and recover the biometrics/identity link of $\mathcal{E}$. For simplicity, we denote c_1 as c in $Exp_{\mathcal{A}}(\lambda)$ below.

Experiment $Exp_{\mathcal{A}}(\lambda)$
$(k, ID_k, \overrightarrow{\mu_k}^0, \overrightarrow{\mu_k}^1, (ID_j, \overrightarrow{\mu_j})_{\{j \neq k\}}) \leftarrow \mathcal{A}(1^\lambda)$
$\overrightarrow{\mu_k} = \overrightarrow{\mu_k}^\beta \xleftarrow{R} \{\overrightarrow{\mu_k}^0, \overrightarrow{\mu_k}^1\}$, $(params) \leftarrow (1^\lambda)$
$\{\text{SIM Card}(P_l, h_l, c_l, \sigma_l) \leftarrow \mathsf{GEN}(params_l, ID_l, \overrightarrow{\mu_l})\}_l$
$\mathtt{TX}_{\mathtt{PUBLISH}}(r_1, cm_1, CRL_1) \leftarrow \{\mathsf{ISSUE}(\text{BlindDLRep}_l, i_l)\}_l$
For each BTC Block:
$res \leftarrow \mathsf{SHOW}(\text{SIM Card}(), \overrightarrow{\mu}, h_{\mathcal{E}}, i_{\mathcal{E}}, UP, AP)$
If $res = 1$, $\mathsf{UPDATE}(UP, AP, h_{\mathcal{E}}, i_{\mathcal{E}})$
$\mathtt{TX}_{\mathtt{PUBLISH}}(r, cm, CRL)$
$\beta' \leftarrow \mathcal{A}(Challenger; \mathsf{Authentication})$
if $\beta' = \beta$ return 1 else return 0

$\mathcal{A}$'s advantage in the above game is $\Pr|[\beta = \beta'] - 1/2|$.

A protocol satisfies Identity Privacy against a malicious $\mathcal{SE}$ if every PPT adversary $\mathcal{A}$ has negligible advantage in $Exp_{\mathcal{A}}(\lambda)$ described as below. For simplicity, we assume that a single entity $\mathcal{E}$ with biometrics $\overrightarrow{\mu}$ is authenticated between two BTC blocks, where the upper part of the game (i.e. above the BTC Block phrase) simulates the setup and enrollment of users, whereas the rest of the game is related to the credential show/update steps simulated by the challenger. The attacker $\mathcal{A}$ is a PPT running against the biometric ABC and a *Challenger* $\mathcal{C}$ simulates the environment for $\mathcal{A}$. The attacker can access any number of BTC Blocks and Authentication attempts till he outputs a guess β'. Although the game resembles the identity privacy game in [24], $Exp_{\mathcal{A}}(\lambda)$ differs in the simulation of credential generation GEN, issuance ISSUE, showing SHOW and UPDATE phases, where the latter two are represented under the process of Authentication.

Theorem 4.1. *Identity Privacy against a malicious $\mathcal{SE}$ or colluding $\mathcal{SE}(s)/\mathcal{SP}$ is guaranteed based on the security of the secret parameters, security of ElGamal Encryption and the Blinded DLREP Schemes.*

Proof. The proof is similar to section 5.3 of [24] with a malicious $\mathcal{SE}$ and honest entities that are not compromised by an attacker. Due to the correct behavior of each $\mathcal{E}$, ZKPs provided by each $\mathcal{E}$ are omitted. Besides, the malicious $\mathcal{SE}$ does not generate any ZKP and $\mathcal{SE}/\mathcal{SP}$ is allowed to obtain the partial or full biometric data of any $\mathcal{E}$. Even then, identity privacy is guaranteed based on the security of the secret parameters and the original credential (generated by the government) stored in $\mathcal{E}$'s core device and the security of the Blinded DLRep. Clearly, the new proposal cannot guarantee the notion for a colluding $\mathcal{SE}$ and $\mathcal{E}$ that are both malicious as a malicious entity reveals his/her secret parameters to a malicious $\mathcal{SE}$ in addition to the credential data. For simplicity, assume that the credential is composed of a single attribute X corresponding to biometrics $\overrightarrow{\mu}$ due to the following Lemma [31].

Lemma 4.2. *DL-REP is at least as hard as Discrete Logarithm (DL), namely finding a DL representation of h is at least as hard as the DL problem.*

In view of the above Lemma, a simple reduction shows that using an oracle that is able to construct a (non-trivial) DL representation of h, breaking the DL problem is easy. The reader is referred to [31] for the details.

We can also allow for the helper device, namely the smart phone to leak some *partial data* related to credential/identity such as update parameters UP, AP, or even (public) identity data such as name, surname, etc. -resulting in an even stronger security model- as long as the core device, namely the SIM Card securely stores the user-specific secret data, *params*, the helper data P and the digital twin of the National ID Card, i.e., h, c, σ. This stronger security model simplifies the challenge for identity ID_k with $\overrightarrow{\mu_k} = \overrightarrow{\mu_k}^{\beta}$ that adversary $\mathcal{A}$ has to process as below:

Let c^* be the challenge ElGamal Encryption using the challenge public key $pk^*_{\mathcal{E}}$ associated to the challenge biometrics X^*, namely $\overrightarrow{\mu^*}$, where $h^* = g^{X^*}$. In fact, h^* is a commitment on X^* that is encrypted in c^* as in Verifiable Encryption of Discrete Logarithms scheme of [44].

Lemma 4.3. *An attacker that has non-negligible advantage against the identity privacy experiment can be used to break the verifiable encryption of discrete logarithms.*

We prove the Lemma by using a reduction which is given $\overrightarrow{\mu^{\mathbf{0}}_{\mathbf{k}}}, P^0$ and $\overrightarrow{\mu^{\mathbf{1}}_{\mathbf{k}}}, P^1$ that results in the two randomly extracted values X_0, X_1, where the challenger of the verifiable encryption scheme randomly picks one value as X^* to compute h^*, c^*. The reduction uses $\overrightarrow{\mu^{\mathbf{0}}_{\mathbf{k}}}, P^0$ and $\overrightarrow{\mu^{\mathbf{1}}_{\mathbf{k}}}, P^1$ to simulate the parameters and the SIM Card(), and h^*, c^* to simulate the challenger of the identity privacy game in coordination with its own challenger, namely the challenger of the verifiable encryption scheme. The adversary ends the identity privacy game by returning β', which is also returned by the reduction to his challenger. For the case analyzed in section 5, where A SIM card with computational capabilities of [3] is considered, the following part of the identity privacy game should be modified as:

For each BTC Block:

$res \leftarrow$ SHOW(SIM Card(), $\overrightarrow{\mu}$)

Lemma 4.4. *Our proposal achieves multi-show unlinkability against the $\mathcal{SE}$ if $\mathcal{CIV}$ and $\mathcal{SE}$ do not collude and if the update mechanism secure.*

The proof is identical to Lemma 5.3 of [2].

Lemma 4.5. *Our proposal achieves biometric-based non-transferability.*

Extracted biometric key resulting from the key release of the deep face fuzzy vault [4] that is encoded as a biometric attribute X_1 as in Figure 3 of [14], must be freshly computed for the interactive ZKPoK protocol of the credential showing. Here, the helper device must guarantee the Liveness assumption as required in any biometric system similar to the other arguments presented in [2].

Lemma 4.6. *Attribute privacy is guaranteed if the core device and the showing protocol are secure.*

The attributes of the original ABC generated by $\mathcal{CG}$ are never disclosed to $\mathcal{CIV}$, hence, privacy against a malicious $\mathcal{CIV}$ is identical to the proof in [2]. As long as $\mathcal{E}$ and $\mathcal{CIV}$'s database are not hacked simultaneously, non-traceability of an entity against a malicious $\mathcal{SE}$ is guaranteed similar to the security against malicious $\mathcal{CIV}$ as discussed in [2].

Lemma 4.7. *Unlinkability of the credential generation and showing processes performed by the government $\mathcal{CG}$ and the service enablers $\mathcal{SE}$s, respectively, is guaranteed.*

The lemma implies that $\mathcal{CG}$ and $\mathcal{SE}$s cannot associate credential issuance and showing phases even if they collude. The proof is identical to Lemma 5.4 of [2].

Corollary 4.1. *Our proposal achieves non-malleable and unlinkable ABC in the sense of Idemix for an independent and non-colluding $\mathcal{CIV}$, in addition to biometric-based non-transferability.*

Another example of non-malleable and unlinkable credential system Idemix, which is an ABC scheme not considered as a self-blindable credential [45]. Credential Showing of Idemix is substantially different from the ones of the other ABC schemes summarized in [45], since an Idemix entity blinds the CL signature (A, e, v) partially, resulting in $(\bar{A}, e, \bar{v})$, and sends $\bar{A}$ to the verifier. Next, the user and the verifier engage in an interactive ZKP, where the entity proves that he knows $e, \bar{v}$, and his private key, without disclosing any of them.

A similar pattern occurs in [2] and thus, in our new proposal, but with additional actors/platforms, i.e. Blockchain and credential generator $\mathcal{CG}$, where the latter is required to generate and biometrically bind the original digital credential h to the user similar to the generation of National ID cards. Hence, the user blinds the digital twin of his National ID card but before engaging in an interactive zero knowledge-proof with the credential issuer $\mathcal{CIV}$, the entity proves to $\mathcal{CIV}$ that he knows an encryption c_1 of h (For simplicity, we denote c_1 as c in $Exp_{\mathcal{A}}(\lambda)$, which is also signed by the $\mathcal{CG}$), and then in an interactive and complex zero knowledge-proof described in section 3.5.2 of [2], he proves his identity $\mathcal{CIV}$ using his blinded credential denoted as BlindDLRep with selective disclosure, i.e. without disclosing any biometric attribute, secret attributes and h. After this, $\mathcal{CIV}$ stores the credential $h_{\mathcal{E}}$ and updates it after each successful authentication in parallel to the user $\mathcal{E}$.

5. Discussion

5.1. A SIM card with computational capabilities of [3]

In our initial design shown in Figure 2, we assume a SIM card with minimum capabilities: It is only responsible for storing the original credential, its encryption signed by the government and the helper data of the fuzzy extractor. However, as one can observe from the original core/helper setting of [3], the core device (i.e. SIM card) generates and stores the private key required to present credentials. Since it is equipped with limited capabilities, namely small memory/computational power, it can only produce "partial show tokens".

If we apply the same framework as described in Remark 2.1 and 2.2, an alternative architecture is obtained as shown in Figure 4. The tradeoff is increased security in case of a lost/compromised helper device, which is now only used for computations of the updated credential at the cost of decreased usability since the core device is expected to handle secrets and to be more powerful than the initial scheme of Figure 2. Besides, the alternative solution in Figure 4 does not require re-issuance of updated/current credential $h_{\mathcal{E}}$ when upgrading the user's smart phone, whereas our initial proposal only prevents re-generation/issuance of the original credential h. In any case, the essential KYC attributes (such as Name and Surname) are entered by the user at each credential showing in addition to the fresh computation of the (hidden) biometric attribute instead of storing them as part of the credential as in [1]. Hence, a careless user who has his phone stolen or has lost it, does not endanger any sensitive data since they are processed on-the-fly in both cases.

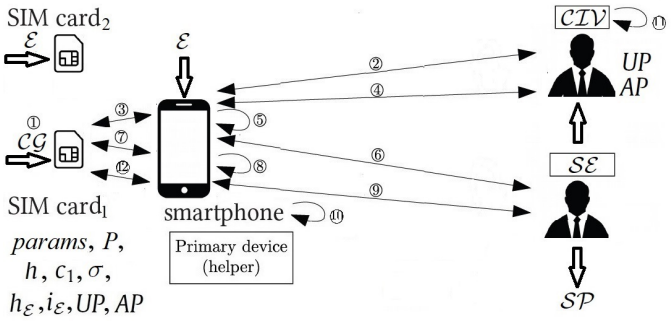

Figure 4: Overview of the Alternative Framework

5.2. Computational Cost

An estimate for the efficiency of the new proposal can be computed based on the exact timings implemented and published by the authors of the deep face fuzzy vault [4]. The authors conduct the runtime measures on a single core of an Intel Core i5-8250U CPU at 1.60 GHz achieving decoding times less than 500 ms for the best configuration, i.e. a FNMR less than 1% at a FMR of 0.01%. The result of the key retrieval procedure is the input to our fuzzy extractor and cryptographic hash function that outputs the value for the biometric attribute X_1 similar to the application presented in Figure 3 of [14]. Hence, the overall fuzzy extraction can be performed in real time on a compatible smartphone [5] for related parameters of fuzzy vault [4] that is crucial for the system's usability.

Apart from the biometric key retrieval process, the traditional credential show of our new proposal has comparable computational cost to U-Prove for the ZK-proof generation times since both schemes depend on DLRep of Brands [27] although our proposal -as opposed to U-Prove [31, 30]- does not combine it with Schnorr's Blind Signature. In fact, Figure 5 presents the ZK proof generation times for $n = 10$ credential attributes, where two of them are revealed, i.e. disclosed, which is the minimum for KYC applications as discussed in section 1.1. Even though our proposal guarantees a security level in the sense of Idemix according to Corollary 4.1, the efficiency is almost 20 times less than the implementation of Idemix for the same number of credential attributes as summarized in Figure 5.

[5] In fact, the latest Geekbench 6 scores prove that Apple's new A17 Pro chip found in the iPhone 15 Pro and Pro Max, challenges AMD and Intel [46]

The comparison of U-Prove against Idemix is presented in [26] (approx. 72ms), and another recent implementation conducted in a laptop with an Intel Core i7-9750H CPU at 2.60 GHz and 16GB of RAM is presented in [40] for a comparison of PS-MS against Idemix (approx. 88ms). The implementations of PS-MS are based on PS-signatures [41] and include variations on the number of revealed attributes r during ZK-proof generation for a fixed number of credential attributes, i.e. $n = 10$ as shown in Figure 5.

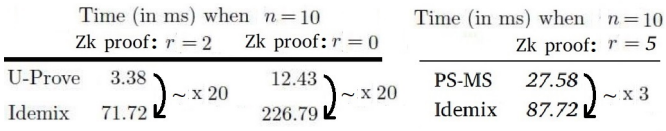

Figure 5: Comparison of *U-prove to Idemix* versus *PS-MS [40] to Idemix* for various number of revealed attributes r, PS-MS: Pointcheval–Sanders Multi-Signatures based on PS-signatures [41]

Based on the implementation results computed for the ABC schemes that are build upon Brands DLRep ([30, 2, 14]), PS-signatures ([16, 17, 40, 1]) and CL-signatures (Idemix [47])), we present a final overview of the computational cost based on the same attribute parameters for compatibility in Figure 6. The details of Figure 5 and 6 are in Appendix C. As one can observe from the right-hand side of Figure 6, Brands DLRep based ABC schemes outperforms the other proposals. We note that the total user cost during credential show equals to the sum of ZK Proof generation and the overhead for face biometrics as shown in Table 3.

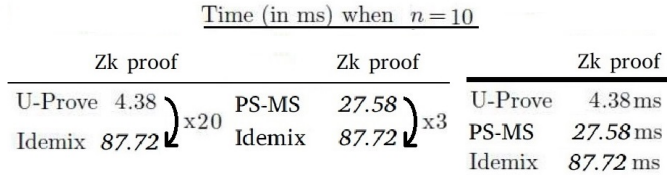

Figure 6: Comparison of *U-prove to Idemix* versus *PS-MS [40] to Idemix* for a fixed number of credential/revealed attributes, PS-MS: Pointcheval–Sanders Multi-Signatures based on PS-signatures [41], ZK-proof generation time for U-Prove is adapted and approximated

5.3. Limitations

A recent research conducted in *Which?* labs reveals that 40% of smartphones that target the cheaper to mid-range end of the market are equipped with face recognition technology that can be spoofed even with a 2D printed photo [48]. Although standard biometric systems assume liveness detection, if the user's smart phone cannot guarantee this feature, employing multimodal biometrics, -for instance fingerprint biometrics as an additional attribute to the ABC- would prevent a spoofing attack. The reader is referred to [42, 19, 49, 50, 14] for fingerprint-based fuzzy extractors. Clearly, the overhead could double resulting in approximately 1 seconds instead of 500ms of face biometrics, even then, the total overhead for face and fingerprint attributes is smaller than the 2.1 seconds in [1] computed for face biometrics alone.

6. Conclusion

In this work, a new biometric ABC scheme is proposed, which achieves two conflicting goals simultaneously: Identity Privacy without sacrificing *efficiency* and *usability*. The new system is by design generic; it can function on existing Blockchains with an OP_RETURN-like opcode/TX such as Bitcoin. In addition to the fulfillment of the privacy-by-design criteria, the new scheme does not require any implementation, more importantly, any complex smart contracts minimizing the TX costs. Lastly, a future work can be implementation of the new system using other public Blockchains that can outperform Bitcoin Blockchain to achieve improved overall performance.

References

[1] J. García-Rodríguez, S. Krenn, D. Slamanig, To pass or not to pass: Privacy-preserving physical access control, Computers & Security 136 (2024) 103566.

[2] N. D. Sarier, Efficient biometric-based identity management on the blockchain for smart industrial applications, Pervasive and Mobile Computing 71 (1) (2021) 101322.

[3] L. Hanzlik, D. Slamanig, With a little help from my friends: Constructing practical anonymous credentials, in: ACM CCS'21, ACM, 2021, p. 2004–2023.

[4] C. Rathgeb, J. Merkle, J. Scholz, B. Tams, V. Nesterowicz, Deep face fuzzy vault: Implementation and performance, Computers & Security 113 (2022) 102539.

[5] WIRED, China Is the World's Biggest Face Recognition Dealer, available at https://www.wired.com/story/china-is-the-worlds-biggest-face-recognition-dealer/ (January, 2023).

[6] DW Shift, China's tech takeover: Do we need to worry?, available at https://www.youtube.com/watch?v=lpCW-IUL4JI (November, 2023).

[7] J. E. Hillman and M. McCalpin, Watching huawei's "safe cities", available at https://www.csis.org/analysis/watching-huaweis-safe-cities (November, 2019).

[8] A. Ekman, China's Smart Cities: The New Geopolitical Battleground, in: Études de l'Ifri, French Institute of International Relations, 2019.

[9] S. Perez, Tiktok claims it's not collecting us users' biometric data, despite what privacy policy says, `https://techcrunch.com` (Articles by Date: September 14, 2022).

[10] A. Simmons, Examining tiktok's potential engagement with data brokers, `https://techpolicy.sanford.duke.edu` (November 14, 2022).

[11] N. D. Sarier, Efficient and usable coercion-resistant e-voting on the blockchain, Cryptology ePrint Archive, Paper 2023/1509, `https://eprint.iacr.org/2023/1509` (2023).

[12] M. Blanton, W. M. P. Hudelson, Biometric-based non-transferable anonymous credentials, in: ICICS'09, Vol. 5927 of LNCS, Springer, 2009, pp. 165–180.

[13] D. Bissessar, C. Adams, D. Liu, Using biometric key commitments to prevent unauthorized lending of cryptographic credentials, in: PST'14, IEEE, 2014, pp. 75–83.

[14] N. D. Sarier, Comments on Biometric-based Non-transferable Credentials and their Application in Blockchain-based Identity Management, Computers & Security 105 (2021) 102243.

[15] C. Adams, Achieving non-transferability in credential systems using hidden biometrics, Security and Communication Networks 4 (2) (2011) 195–206.

Table 3: Comparison of recently introduced biometric-based ABC schemes. ^{x}: Chapter 3 of [27], †: PS-MS of [40], ‡: for U-Prove computed in [26], *: based on [32], **:based on [4], $^-$: False Non Match Rate (FNMR) for a fixed False Match Rate of 0.01%, NC: Non-Comparable

	Underlying Signing Scheme (σ)	Approx. time for ZK Proof Generation of ABC based on σ	Overhead for face	FNMR$^-$ for face	GDPR-Compliance	Identity Privacy	TX Cost
[1]	PS [41]	27.58ms† [40]	2.1s* [1]	<15%*	No	No	NC
New	Brands DLRep[27]^x	4.38ms‡ [26]	500ms**[4]	<1%**	Yes	Yes	Low

[16] A. Sonnino, M. Al-Bassam, S. Bano, S. Meiklejohn, G. Danezis, Coconut: Threshold issuance selective disclosure credentials with applications to distributed ledgers, in: NDSS'19, 2019.

[17] Y. Yu, Y. Zhao, Y. Li, L. Wang, X. Du, M. Guizani, Blockchain-based anonymous authentication with selective revocation for smart industrial applications, IEEE TII.

[18] D. Rathee, G. V. Policharla, T. Xie, R. Cottone, D. Song, Zebra: Snark-based anonymous credentials for practical, private and accountable on-chain access control, Cryptology ePrint Archive, Paper 2022/1286, https://eprint.iacr.org/2022/1286 (2022).

[19] N. D. Sarier, Biometric cryptosystems: Authentication, encryption and signature for biometric identities, Ph.D. thesis, Bonn University, Germany (2013).

[20] C. Farmer, 5 ways GBG Identity Score helps know your customer, https://www.gbgplc.com/en/blog/5-ways-gbg-identity-score-helps-know-your-customer/ (Retrieved on December, 2023).

[21] J. Bringer, H. Chabanne, M. Izabachène, D. Pointcheval, Q. Tang, S. Zimmer, An application of the goldwasser-micali cryptosystem to biometric authentication, in: ACISP'07, Vol. 4586 of LNCS, Springer, 2007, pp. 96–106.

[22] N. D. Sarier, Security notions of biometric remote authentication revisited, in: STM'11, Vol. 7170 of LNCS, Springer, 2012, pp. 72–89.

[23] K. Simoens, J. Bringer, H. Chabanne, S. Seys, A framework for analyzing template security and privacy in biometric authentication systems, IEEE TIFS 7 (2) (2012) 833–841.

[24] N. D. Sarier, Privacy preserving biometric authentication on the blockchain for smart healthcare, Pervasive Mob. Comput. 86 (2022) 101683.

[25] S. A. Brands, A technical overview of digital credentials, http://credentica.com (Consulted on 2019).

[26] M. Chase, S. Meiklejohn, G. Zaverucha, Algebraic macs and keyed-verification anonymous credentials, in: ACM CCS'14, ACM, 2014, pp. 1205–1216.

[27] S. A. Brands, Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy, MIT Press, 2000.

[28] S. Brands, L. Demuynck, B. D. Decker, A practical system for globally revoking the unlinkable pseudonyms of unknown users, in: ACISP'07, Vol. 4586 of LNCS, Springer, 2007, pp. 400–415.

[29] R. Impagliazzo, S. M. More, Anonymous credentials with biometrically-enforced non-transferability, in: WPES '03, ACM, 2003, pp. 60–71.

[30] C. Paquin, G. Zaverucha, U-prove cryptographic specification v1.1, in: Technical Report, Microsoft Corporation, 2011.

[31] G. Alpar, U-Prove Cryptography, in: Available at http://www.cs.ru.nl/ gergely/objects/u-prove.pdf, Accessed on: August, 2019.

[32] A. Ouamane, M. Bengherabi, A. Hadid, M. Cheriet, Side-information based exponential discriminant analysis for face verification in the wild, in: FG'15, IEEE, 2015, pp. 1–6.

[33] M. Kiraz, E. Larraia, O. Vaughan, Nft trades in bitcoin with off-chain receipts, in: ACNS'23 Workshops, Vol. 13907 of LNCS, Springer, 2023, pp. 100–117.

[34] J. Camenisch, A. Lysyanskaya, An efficient system for non-transferable anonymous credentials with optional anonymity revocation, in: EUROCRYPT'01, Vol. 2045 of LNCS, Springer, 2001, pp. 93–118.

[35] J. Camenisch, A. Lysyanskaya, Signature schemes and anonymous credentials from bilinear maps, in: CRYPTO'04, Vol. 3152 of LNCS, Springer, 2004, pp. 56–72.

[36] J. Camenish, A. Lysyanskaya, Dynamic accumulators and application to efficient revocation of anonymous credentials, in: CRYPTO'02, Vol. 2442 of LNCS, Springer, 2002, pp. 61–76.

[37] M. Babel, J. Sedlmeir, Bringing data minimization to digital wallets at scale with general-purpose zero-knowledge proofs (2023). arXiv:2301.00823.

[38] M. Rosenberg, J. White, C. Garman, I. Miers, zk-creds: Flexible anonymous credentials from zksnarks and existing identity infrastructure, in: IEEE SP'23, 2023, pp. 790–808.

[39] N. D. Sarier, Privacy preserving biometric identification on the bitcoin blockchain, in: CSS'18, Vol. 11161 of LNCS, Springer, 2018, pp. 254–269.

[40] J. García-Rodríguez, R. T. Moreno, J. B. Bernabe, A. Skarmeta, Implementation and evaluation of a privacy-preserving distributed abc scheme based on multi-signatures, JISA 62 (2021) 102971.

[41] D. Pointcheval, O. Sanders, Reassessing security of randomizable signatures, in: CT-RSA'18, Vol. 10808 of LNCS, Springer, 2018, p. 319–338.

[42] N. Li, F. Guo, Y. Mu, W. Susilo, S. Nepal, Fuzzy extractors for biometric identification (2017).

[43] Y. Dodis, L. Reyzin, A. Smith, Fuzzy extractors: How to generate strong keys from biometrics and other noisy data, in: EUROCRYPT'04, Vol. 3027 of LNCS, Springer, 2004, pp. 523–540.

[44] M. Stadler, Publicly verifiable secret sharing, in: EUROCRYPT'96, Vol. 1070 of LNCS, Springer, 1996, pp. 190–199.

[45] J. H. Hoepman, W. Lueks, S. Ringers, On linkability and malleability in self-blindable credentials, in: WISTP'15, Vol. 9311 of LNCS, Springer, 2015, p. 203–218.

[46] M. J. White, The iPhone 15's chip challenges Intel's fastest desktop CPU - but there's a catch, https://www.digitaltrends.com/computing/apple-a17-pro-beats-intel-and-amd/ (2023).

[47] IBM, Specification of the identity mixer cryptographic library (revised version 2.3.0), in: IBM Research Report RZ 3730, 2010.

[48] A. Axworthy, Face recognition on 40% of new phones easily spoofed with a printed photo, https://www.which.co.uk/news/article/face-recognition-mobile-phones-axNDM2P9VvyO (2023).

[49] A. Arakala, J. Jeffers, K. Horadam, Fuzzy extractors for minutiae-based fingerprint authentication (2007).

[50] W. Yang, J. Hu, S. Wang, A delaunay triangle-based fuzzy extractor for fingerprint authentication, in: TrustCom'12, IEEE, 2012, pp. 66–70.

Author Bio: N. Deniz Sarier received her M.Sc. degree in Media Informatics from RWTH Aachen, and PhD degree in Computer Science from b-it, cosec of Bonn University, Germany in 2007 and 2013, respectively. As an Assoc. Prof. of Computer Science, her research interests include biometric security, Blockchain, identity management, public-key cryptography, in particular, integration of biometrics into cryptographic/blockchain applications.

7. Appendix A

7.1. Background on the core/helper setting

Briefly, [3] leverages smartphones in conjunction with smart cards, to let both jointly present shared credentials. Their scheme is shown to be efficient in practice, and in particular ensures that the computational overhead of the core device is independent of the number of attributes in the credential. Our system achieves the same independency although we adapt the setting for our own use-case. From the security perspective, the authors of [3] considers the helper device to be potentially malicious, although a corrupted helper device can always break system's privacy.

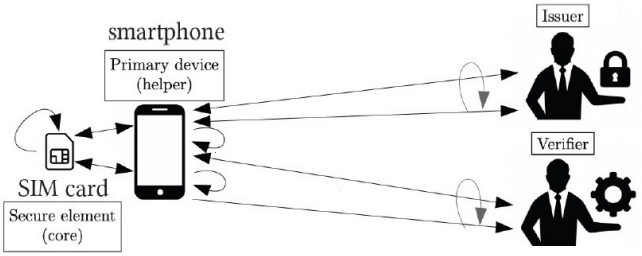

Figure 7: Overview of Core/Helper Setting [3]

7.2. Fuzzy Extractors and Bitcoin Blockchain

The reader is referred to [14] and Appendix C of [11] for a detailed analysis on Fuzzy Extractors and their integration into Brands DLRep. Although our system is instantiated using Bitcoin, the system can employ any blockchain platform (preferably with a higher transaction rate) that allows the sender of a Transaction (TX) to store arbitrary data in TX as in [11, 2]. The reader is referred to a detailed analysis on Bitcoin dust value limit and a Bitcoin instantiation of a similar system to [11] and [2].

8. Appendix B: Efficient biometric-based identity management on the Blockchain

8.0.1. Brands selective disclosure scheme (DLRep)

Brands [27] presented the digital credentials scheme, where the same credential, signatures and parameters are used in each instance of the showing protocol resulting in a single-show credential system. Thus, Brands credentials are linkable, and identical to Bitcoin, pseudonymity instead of anonymity can be achieved for them. Similarly, U-Prove [30] does not allow unlinkable reuse of credentials: To unlinkably use a credential again, a user must get it reissued. However, from the efficiency point of view, Microsoft U-Prove [30] (based on Brands' work [27]) is evaluated as the most efficient construction. Brands selective disclosure scheme [27] enables selective disclosures involving an identity with $n-1$ fields, $(X_1, ..., X_{n-1})$. (For example, X_1 may represent a user's biometric attribute, X_2 her name, and X_{n-1} be the nationality etc.). Let q be a prime number and $\mathbb{G}$ a group of order q. Let $g_0, g_1, ..., g_n \in \mathbb{G}$, where g_n is reserved for additional attributes merged by other organizations to the credential. Note that the number of the parameters n generated by the government could be increased depending on the organizational requirements of $\mathcal{E}$. For simplicity, we arranged a single parameter g_n for a single additional attribute X_n to be merged later by the organization. X_0 prevents an attacker with a priori knowledge on the X_j attributes of an entity from performing a dictionary attack where she guesses values for the remaining X_js [2]. The reader is referred to Appendix B of [11] for the definition of DLRep and the associated credential show protocol. We reserve X_n for the organizational attribute inserted by the $\mathcal{CIV}$ and X_1 for the biometric attribute, which is extracted by the government from the entity's biometrics as described in [14, 11, 2].

8.0.2. Blind Attribute Merging

Due to the governmentally produced ABC and signed (encrypted) ABC data, any organization, bank or company has to trust both the device and the signature on organization specific attributes describing the entity, $\mathcal{CIV}$ blindly adds the specific attributes to the digital credential h as follows: First, the $\mathcal{CIV}$ determines the new attributes associated to the entity, for instance the new attributes could be the role, the position, the department,etc. of the entity. For simplicity, lets assume that $\mathcal{CIV}$ adds a single attribute to h. First, $\mathcal{CIV}$ shares the attribute information X_n with the entity and sends the encryption c_2 of the $g_n^{X_n}$ using the public key of $\mathcal{E}$ to the device of $\mathcal{E}$, which uses $h = \prod_{j=0}^{n-1} g_j^{X_j}$ with g_n generated by the government during credential generation phase to be used for additional attributes to compute $h' = \prod_{j=0}^{n} g_j^{X_j} = h g_n^{X_n}$. The device of the entity returns the encryption $c_3 = c_1 \cdot c_2$ to $\mathcal{CIV}$. Due to the multiplicative homomorphic property of ElGamal encryption, $\mathcal{CIV}$ can easily verify the correctness of the computation, which corresponds to the encryption of h'. Next, the device of $\mathcal{E}$ performs the following.

8.0.3. Blinded DLRep Scheme

Let $(g_0, g_1, ..., g_n)$ be the parameters of the DLRep commitment. Consider an additional parameter $z \in \mathbb{Z}_q$ where $z \neq 1$. Let $h' = \prod_{j=0}^{n} g_j^{X_j}$ with respect to $(g_0, g_1, ..., g_n)$. Then the values $(\beta_1, \beta_2) = (z^\gamma, h'^\gamma)$ with $\gamma \overset{R}{\leftarrow} \mathbb{Z}_q$ can also be viewed as a commitment to the same $(X_0, X_1, ..., X_n)$. Here, we need an additional Schnorr signature that proves knowledge of γ such that $\beta_1 = z^\gamma$.

Let us define a new DLRep scheme, which we will call the blinded DLRep scheme, where BlindDLRep $= (z^\gamma, h'^\gamma)$. Finally, using well-known Σ protocols, it is easy

to see that the same set of relations that can be proven about values inside a DLRep can be proven about values inside a blinded DLRep [2].

The entity returns BlindDLRep together with the proof that it corresponds to h', i.e. $(log_z h' = log_{\beta_1}\beta_2)$ to the $\mathcal{CIV}$ for credential storage on the Merkle tree generated by $\mathcal{CIV}$. Since the issuer $\mathcal{CIV}$ does not know h' but only its encryption/commitment c_3, we have to construct a complex ZKP proof for the equality of the statement $log_z h' = log_{\beta_1}\beta_2$, i.e. a composite zero knowledge proof argument for proving equality of two logarithms and c_3 corresponds to the encryption of h'. Here, the device of the entity re-randomizes c_3 using γ by raising c_3 to the power of γ and the $\mathcal{CIV}$ checks that c_3^γ corresponds to the ElGamal encryption of $\beta_2 = h'^\gamma$ using the following ZKP:

$PoK\{\gamma, h' | (\beta_1 = z^\gamma) \wedge (\beta_2 = h'^\gamma) \wedge (c_3^\gamma \in Enc(h'^\gamma))\}$

After verifying the ZKP, $\mathcal{CIV}$ stores the credential (z^γ, h'^γ) of the entity $\mathcal{E}$ on the Merkle tree as $h_\mathcal{E}$.

Moreover, Brands [27] shows that if the discrete logarithm problem is difficult, *DLRep* h (and thus h') is one-way and collision intractable, preventing forgery attacks. Clearly, replacing h with a DLRep h' having more attributes does not affect the security. The proof for blinded DLRep (z^γ, h'^γ) and the remaining proofs are in [2].

8.1. Credential Storage

We remind that the original digital credential issued by the government is stored at the core device. However, the user device does not store any biometric image or template data as opposed to [1, 29, 15, 30]. In particular, even if the integrity of the device holding the credential is breached, it is still impossible to recover the biometric data of the user as in [12]. We employ a Merkle tree with 2^{256} leaves the paths to which are given using each bit of a SHA-256 hash to indicate which binary branch to take at each node. We denote the entity's position in the tree by $i_\mathcal{E}$.

8.2. Publication of BlindDLREPs

$\mathcal{CIV}$ publishes the root $r_{\mathcal{CIV},t}$ of a Merkle tree as a Bitcoin transaction that commits to the state at time t of the credentials for each of the entities as shown in Figure 2. To compute $r_{\mathcal{CIV},t}$, $\mathcal{CIV}$ must both add new entities to the tree and edit the information for existing entities after updating their BlindDLRep as a resut of a successful authentication at the $\mathcal{SE}$. We want to randomize the path to an entity's entry and update the BlindDLRep after each successful authentication so that the entity cannot authenticate using a previous, out-of-date authentication path in the tree and to provide multi-show unlinkability against the $\mathcal{SE}$. The details of the transaction TX_{id} with transaction id containing the OP_RETURN script is shown in Figure 2 and 3. The cost of membership and non-membership proofs through Merkle tree is $O(\log(N))$, where the latter requires a sorted tree. For auditing through blockchain, we publish both the commitment to the root of the unsorted (and optionally sorted) Merkle tree denoted as $r_{\mathcal{CIV},t}$ (and $r^s_{\mathcal{CIV},t}$), respectively. The latter can be used for verifying/auditing the used/spent credentials after each successful authentication of entities if necessary. To provide immutability through hash chaining, $cm_{\mathcal{CIV},t} = \mathrm{H}\ (cm_{\mathcal{CIV},t-1},\ r_{\mathcal{CIV},t})$ as the unsorted Merkle tree root commitment and optionally the respective commitments for the sorted tree are also published. Finally, a commitment to the current Credential Revocation List (CRL) that lists the permanently revoked credentials $CRL^U = \mathrm{H}(CRL) \cup h_\mathcal{E}$ is also published as CRL.

8.3. Credential Showing

The entity proves possession of attributes to $\mathcal{SE}$ using $h_\mathcal{E}$ =BlindDLRep^U, where using well-known Σ protocols, it is easy to see that the same set of relations that can be proven about values inside an updated blinded DLRep $h_\mathcal{E}$ as done in section 8.0.3 can be proven for the values inside a blinded DLRep (z^γ, h'^γ) by replacing γ with $U\gamma$, where z, U and γ all known to the entity.

8.4. Credential Renewal/Thaw/Suspension

As discussed in section 3.3.1, $\mathcal{CIV}$ returns the entity the secret credential update parameter UP so that both parties can compute U and update the stored credential via (BlindDLRep)^U after each successful authentication automatically. Here, PRF denotes any secure Pseudo Random Function and H is SHA-256 with H^k denoting SHA-256 applied k times. A pseudorandom function (PRF) is a keyed function, hence we can take a keyed hash function for the PRF, where the secret key is shared between the entity and $\mathcal{CIV}$ to be stored in the database of $\mathcal{CIV}$ such as a password that the entity enters during each authentication attempt. Hence this password is not stored on the entity's device as in [2]. To update the organizational attributes merged by the $\mathcal{CIV}$ during the credential issuance phase, we process identical to [2]. Similarly, after an unsuccessful authentication attempt of the entity due to a suspended credential use, we process identical to [2]. $\mathcal{CIV}$ can track the number of authentication attempts of the entity using the value k, hence the entity may have the option to authenticate at most k-times. After this k-time period, the entity is automatically suspended, if a Thaw request is not made by the entity to the $\mathcal{CIV}$ as in [2]. Here, we count on the security of the user device [2].

Algorithm 9: Renewal of entity credential due to a modified organizational attribute

Input: $i_\mathcal{E}$, branch of Merkle tree, $h_\mathcal{E}$
Output: Updated Merkle tree, $i_{\mathcal{E}^N}, h_{\mathcal{E}^N}, UP, AP$
Entity/Organization informs $\mathcal{CIV}$ about the attribute change/update
$\mathcal{CIV}$ confirms the new attribute
$\mathcal{CIV}$ repeats credential issuance phase with the new attribute(s): Identical to Algorithm 2
$\mathcal{CIV}$ updates Merkle Tree:
-Insert a random value at the position $i_\mathcal{E}$
-Record $(h_\mathcal{E})^N$ at the position $(i_\mathcal{E})^N$
$\mathcal{CIV}$ updates Database entry for the Entity:
-$\mathcal{CIV}$ replaces $h_\mathcal{E}$ and $i_\mathcal{E}$ with $(h_\mathcal{E})^N$ and $(i_\mathcal{E})^N$

Algorithm 10: Thaw process following a suspension

Input: counter k, $i_{\mathcal{E}}$, branch of Merkle tree, $h_{\mathcal{E}}$
Output: Updated Merkle tree, $(i_{\mathcal{E}})^T, (h_{\mathcal{E}})^U, UP, AP$
$\mathcal{CIV}$ compares $\mathcal{E}$'s counter to the system threshold, i.e. k-times
-If equal, $\mathcal{CIV}$ assigns $k = 0$ and $\mathcal{E}$ is suspended
$\mathcal{E}$ makes a Thaw request after a # of rejects
$\mathcal{CIV}$ verifies the request and in case of an attribute change,
$\mathcal{CIV}$ runs credential issuance phase with the new attribute(s):
-Identical to Algorithm 2
Otherwise, $\mathcal{CIV}$ requests $\mathcal{E}$'s latest counter value and
$\mathcal{CIV}$ synchronizes $\mathcal{E}$'s database entry:
$\mathcal{E}$'s counter differs from the system threshold by d
-Repeat d times:
$\mathcal{CIV}$ computes $(i_{\mathcal{E}})^T = \text{PRF}(AP)$
$\mathcal{CIV}$ computes U=PRF(UP) and $(h_{\mathcal{E}})^U = (\overline{z}, \overline{h'})^U$
$\mathcal{CIV}$ assigns UP=H(UP) and AP=H(AP)
-Increment k by 1
$\mathcal{CIV}$ updates Merkle Tree:
-Insert a random value at the position $i_{\mathcal{E}}$
-Record $(h_{\mathcal{E}})^T$ at the position $(i_{\mathcal{E}})^T$
$\mathcal{CIV}$ updates Database entry for the Entity:
-Replaces $h_{\mathcal{E}}$ and $i_{\mathcal{E}}$ with $(h_{\mathcal{E}})^U$ and $(i_{\mathcal{E}})^T$

8.5. Credential/Identity Revocation

To revoke the credential permanently, $\mathcal{CIV}$ will replace the $(\text{BlindDLRep})^U$ at the position $i_{\mathcal{E}}$ by a random value as in the case of an automatic update performed after each successful authentication. This way, a revoked entity cannot prove his identity to $\mathcal{SE}$ to access a service provided by the $\mathcal{SP}$ similar to the case where the entity cannot prove his identity against an out-of-date/suspended credential (i.e. his previous credential). The main difference to the suspension operation is that revocation inserts a random value to the last valid index of the entity, where suspension leaves this index unchanged until a valid Thaw request comes in. For the revoked credentials, $\mathcal{CIV}$ keeps a Credential Revocation List CRL that is published on the Blockchain as a commitment as shown in Algorithm 8.

9. Appendix C

In this section, we present the original results of the implementations in [26, 40] as show in Figure 8 and 9.

	Time (in ms) when $(n,c,r) =$ (10,2,2)	(10,10, 0)	Credential size (in bits)
U-Prove	3.38	12.43	1024
Idemix	71.72	226.79	5369
Bilinear CL	20.98	28.32	$512n + 768$

estimated presentation proof generation cost. U-Prove and bilinear CL use 256-bit elliptic curve parameters, Idemix uses a 2048-bit modulus. n is the number of attributes, r is the number of *revealed* attributes in a presentation proof, and c is the number of *committed* attributes.

Figure 8: Comparison of U-Prove to Idemix [26]

Method	NS	NAttr	NRevAttr	Zk proof	Zk verify
PS-MS	2	10	5	*27.58*	*40.15*
PS-MS	10	10	5	*27.58*	*40.15*
Idemix	–	10	5	*87.72*	*85.31*

Figure 9: Comparison of Idemix to PS-MS [40] : PS-MS: Pointcheval–Sanders Multi-Signatures based on PS-signatures [41], NS: Number of signers, NAttr: Number of Attributes n, NRevAttr: Number of Revealed Attributes r

In [1], templates are denoted by Bio, where a_{Bio} refers to a biometric template to which a credential is bound, i.e. the template included as an attribute in the credential.

ParGen(1^λ). Return $pp \leftarrow\$\ PS.\text{ParGen}(1^\lambda)$

Key Generation and Issuance.

I.KeyGen(pp). Return $(\text{sk}, \text{pk}) \leftarrow\$\ PS.\text{KeyGen}(pp)$

I.IssueCred($\text{sk}, a_{Bio}, \boldsymbol{a}$). $\boldsymbol{b} = (a_{Bio}, \boldsymbol{a})$.
Return $\sigma \leftarrow\$\ PS.\text{Sign}(\text{sk}, \boldsymbol{b})$

U.VerifyCred($\text{pk}, \sigma, a_{Bio}, \boldsymbol{a}$). $\boldsymbol{b} = (a_{Bio}, \boldsymbol{a})$.
Return 1 if $PS.\text{Verify}(\text{pk}, \boldsymbol{b}, \sigma) = 1$, else return 0

Presentation.

U.GenEph(pp). Return $\text{sk}_{ae} \leftarrow\$\ AES.\text{KeyGen}()$.

R.GenEphUser(Bio_f, ri_U). Parse $ri_U = \{\text{sk}_{ae}\}$.
Compute $(C_{Bio_f}, V_{Bio_f}) \leftarrow\$\ BitPC.\text{Commit}(Bio_f)$,
$C_{Bio_f} = (C_1, \ldots, C_N)$, $V_{Bio_f} = (r_1, \ldots, r_N)$
Return $ro_U = AES.\text{Encrypt}(\text{sk}_{ae}, \{C_{Bio_f}, V_{Bio_f}, Bio_f\})$.

U.Present($\text{pk}, \sigma, a_{Bio}, \boldsymbol{a}, \phi, ro_U, ri_U, ctx$).
Parse $\{C_{Bio_f}, V_{Bio_f}, Bio_f\} = AES.\text{Decrypt}(\text{sk}_{ae}, ro_U)$.
If decryption fails, return $\perp$ $\boldsymbol{b} = (a_{Bio}, \boldsymbol{a})$.
Given $a_{Bio} = (e_i)_{i\in[N]}$, $Bio_f = (f_i)_{i\in[N]}$.
Choose random blinding values $w, z \leftarrow\$\ \mathbb{Z}_r$.
Take σ as (a', σ_1, σ_2) and compute $(\sigma_1', \sigma_2') = (\sigma_1^w, (\sigma_2\sigma_1^z)^w)$.
compute an Schnorr-style proof:
$pt \leftarrow\$\ \text{NIZK}[(z, (e_i), \boldsymbol{a}, (f_i), (r_i), s, r) :$

Verification.

V.InputGen(pt). $ri_V = \varepsilon$.

R.GenEphVerifier(Bio_f, ri_V). Return C_{Bio_f}

V.Verify($\text{pk}, pt, \phi, ro_V, ctx$). Parse $C_{Bio_f} \leftarrow ro_V$
If pt verifies correctly return 1. Else, return 0

Figure 10: Summary of BioABC-ZK: A credential σ on attributes $\mathbf{a} = (a_1, \ldots, a_n)$, where a_{Bio} is the biometric attribute that binds the credential to the user and a_i are her identity attributes [1].