Published in: Journal of Information Security and Applications journal.

This version of the article has been accepted for publication, after peer review and is subject to Elsevier terms of use.

The Version of Record is available online at: http://dx.doi.org/10.1016/j.jisa.2021.102814

ECB2: A novel encryption scheme using face biometrics for signing blockchain transactions

Carmen Bisogni [a,*], Gerardo Iovane [a], Riccardo Emanuele Landi [b], Michele Nappi [a]

[a] Department of Computer Science, University of Salerno, Salerno, Italy
[b] Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milan, Italy

ARTICLE INFO

Keywords:
Biometrics
Smart contracts
Blockchain
Face biometrics
Biometricsignature
Encryption scheme
RSA
Information Fusion
Authentication scheme

ABSTRACT

Blockchain is the technology on the basis of the recent smart and digital contracts. It ensures at this system the required characteristics to be effectively applied. In this work, we propose a novel encryption scheme specifically built to authorize and sign transactions in digital or smart contracts. The face is used as a biometric key, encoded through the Convolutional Neural Network (CNN), FaceNet. Then, this encoding is fused with an RSA key by using the Hybrid Information Fusion algorithm (BNIF). The results show a combined key that ensures the identity of the user that is executing the transaction by preserving privacy. Experiments reveal that, even in strong heterogeneous acquisition conditions for the biometric trait, the identity of the user is ensured and the contract is properly signed in less than 1.86 s. The proposed ECB2 encryption scheme is also very fast in the user template creation (0.05s) and requires at most four attempts to recognize the user with an accuracy of 94%.

1. Introduction

In recent times there was a flood of interest in blockchain due to the new frontiers of its applications [1,2]. The original idea at the basis of this technology was introduced in 1982 by David Chaum [3] but found its applicability only in 2008 thanks to Satoshi Nakamoto,[1] becoming the core system of the crypto coin Bitcoin. Blockchain is, for its definition, a growing list of data blocks linked together and protected by cryptography; a distributed ledger recording transactions between involved parts in a permanent and verifiable manner [4]. No intermediaries are needed, transaction costs are lowered, and the security related to operations is enhanced thanks to the consensus reached among the network nodes. The concept of consensus and security in transactions represents the core of the main critical points of this new technology [5]. The most used consensus algorithm is the Proof-of-Work, in which miners validate transactions by founding blocks through computation [6]. Once written in a block, data cannot be altered without modifying all subsequent blocks, since this would require the consensus of the majority of the nodes in the network. Despite the Proof-of-Work makes malicious activities inconvenient, some nodes of the network can still produce concurrent blocks simultaneously, introducing a "fork" in the chain [7], that is, a malicious deviation of the main chain that guarantees the consistency and integrity of the system. However, the solution at this problem, identified as full ASIC-resistant decentralization [8], results to be an energetically costly process [9,10]. In this case, the Proof-of-Stake consensus mechanism comes into play, allowing the selection of the block creator in the function of its "reliability" [11].

The above set of problems and solutions are inherited by Digital or Smart Contract, which is a blockchain-based software solution aimed at realizing contracts [12]. The main platform that allows its usage is Ethereum [13], whose computational power is paid through the accounting unit called Ether (ETH). The Digital or Smart Contract is used, e.g., for managing electoral systems, financial markets, or intellectual properties, and its validity is guaranteed by the blockchain technology. Its Proof-of-Work mechanism and decentralization are similar to Bitcoin [14], but the amount of information to write on blocks is larger, and users can program contracts by defining custom rules and policies through the Solidity programming language [15]. In the case of the digital coin, another criticism can be observed: double-spending. Though double-spending, malicious users can spend the same currency at least twice [16–18]. Some of the attacks that can be found in this category are the *51% attack*; the *race attack*; the *finney attack*; the *alternate history attack*; the *vector 76 attack*. Penalties represent the common mitigation technique used against the double-spending problem applied considering the amount of time in which a block is hidden from the blockchain network.

* Correspondence to: University of Salerno, Via Giovanni Paolo II, 132 - 84084, Fisciano (SA), Italy.
E-mail address: cbisogni@unisa.it (C. Bisogni).

[1] It is not known if the name indicates a person or a group.

It is clear that, for its nature, digital or smart contracts offer a great opportunity, but they need to be carefully disciplined [19]. A solution that uses a different perspective, which is a super-fast Byzantine Agreement, is Algorand [20]. In this paper, we propose a new signature system for transactions, built explicitly for digital or smart contracts, on which the conducted tests have produced encouraging results. ECB2 is an encryption scheme, designed to be fast and user-friendly, based on the fusion of RSA based encryption and face biometric techniques. The rest of the paper is organized as follows: in Section 2, an overview of the biometric signature and its usage in the blockchain are introduced, and ECB2 and its characteristics are exposed in Section 3; in Section 4, the experimental settings, including the adopted dataset and results, are presented and discussed and, finally, in Section 5, we drew our conclusions and proposals for some further improvements.

2. Related works

The methods involved in the proposed work concern both encryption techniques involved in the blockchain and biometric traits in authentication systems.

Biometric authentication is very common since it is not required for the user to have a physical device or to remember a password and it is less affected by identity thefts [21]. In recent years, the computational capabilities of recent devices allow various authentication systems to benefit from the biometric check [22]. The choice of the involved trait depends on the platform on which the authentication is performed; on smartphones, for example, the main source of information is represented by the camera and, in some cases, by gyroscope and accelerometer [23]. This leads to the usage of biometric traits like ear [24], face [25] or iris [26]. The combination of the latter is also used in biometric authentications [27–29]. By focusing on the biometric key in cryptosystems, the main applications concern IoT [30] or, in general, the cloud [31]. All of them focus on the necessity to hide personal content [32]. In recent years, the advent of blockchain has led to the benefit from biometric authentication also in this field, since personal data in the user's wallet are subjected to malwares able to capture single keys. In this context, biometric authentication helps to prevent those attacks, as proposed in [33], by using face recognition.

Authentication through biometric data for transactions execution based on blockchain is proposed in [34], in which users are authenticated through certificates based on fingerprints and iris templates to support electronic identity, and in [35], where digital or smart contracts are signed through fingerprint biometric data by combining the related encrypted images with user's non-sensitive information. The above authentication approach introduces valuable strengths in terms of security and transparency in executing economic transactions [36] as well as for designing Smart City solutions, and digital identity [37], in which sensible information must be suitably protected. Biometrics as a mean of authentication on the blockchain allows the verification of the user's identity in a distributed way, without the need to establish a central entity that stores sensitive data. Also, it prevents malicious activities, forcing the user to sign transactions with his personal biometric information. The double-spending problem has been addressed through different approaches, among which cooperative P2P systems [38], fair coin deposits [39], fair blind signatures [40], recipient-oriented transactions [41] and multistage secure pool [42]. Despite these solutions, the biometric authentication allows us to further discourage the attacker from carrying out double-spending as well as to punish him permanently inside the blockchain network.

To date, although there is experience related to Information Fusion for encryption, this work represents one of the first solutions for signing transactions biometrically in the context of blockchain technology to achieve suitable protection against the problem of double-spending. With ECB2, the user is prevented to continuously carry out malicious actions, e.g. in a decentralized network that uses Proof-of-Stake-based consensus, by hiding historical punitive measures by migrating to new wallets. By signing transactions with their biometric information, malicious users remain punished so they cannot continually violate the laws of the network. The use of Information Fusion for merging biometric information with cryptographic keys introduces interesting perspectives in terms of blockchain integrity, since to date there are no encryption schemas that allow the proposed signature mode applied to blockchain infrastructure. With the present study, we want to introduce an encryption scheme that aims to improve the security and reliability of transactions in digital or smart contracts by using face biometrics.

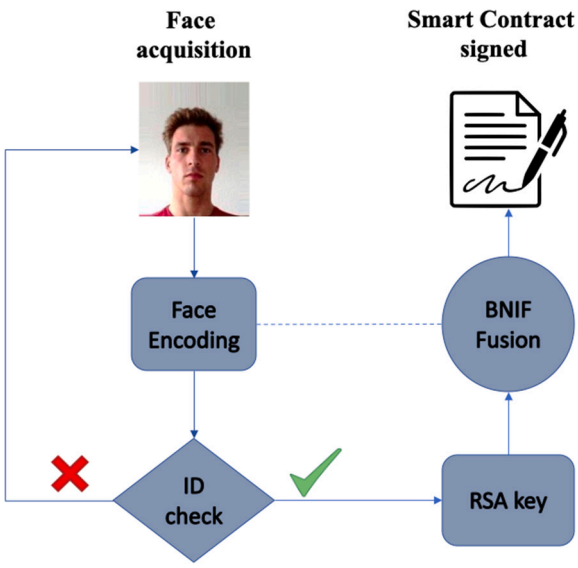

Fig. 1. The overall structure of ECB2.

3. The proposed framework: ECB2

We propose a system through which digital or smart contracts are signed with signatures generated by fusing face biometrics information and RSA private key, showing how this solution contributes to solving the double-spending issue in executing transactions. The operative phases of ECB2 are described in the following sections in terms of face biometrics acquisition and manipulation, private key generation, information fusion, and contract signing procedures. The overall structure of ECB2 can be seen in Fig. 1.

3.1. Face biometrics

To obtain features representative of the face, we followed a classical face recognition pipeline. The difference is that, in this case, we are interested only in the features vector and, for this reason, the steps we will follow are declined as:

- face detection;
- face encoding.

To this step, we will add a step called "face encoding manipulation" to adapt the feature vectors to our system.

3.1.1. Face detection

To make our algorithm able to operate in real-time, we choose the Histogram of Oriented Gradients (HOG) as a face detector. This method was firstly introduced by [43] to perform human detection and, subsequently, it was specifically adapted by [44] for face recognition. By this algorithm, the image is firstly converted in grayscale, since colors are not necessary to perform face recognition. Then, the image is divided into connected regions and, for each of which, the histogram

of edge orientations is computed. A classification algorithm, a Support Vector Machine (SVM), is then trained with positive and negative examples by using sliding windows. In particular, the SVM classifier is trained with the features vectors obtained by positive and negative examples through a method called *hard-negative mining*.

3.1.2. Face encoding

Once the face is detected within the image, we perform the encoding of the latter by using a Convolutional Neural Network (CNN). FaceNet was introduced by [45], and it was trained on LFW, reaching 99.63% of accuracy in recognition tasks, and on YouTube Faces DB, on which it reached 95.12%. This CNN was trained to become able to generate 128 measurements, i.e. an *embedding*, for each person, which is particularly suitable for our goals since it showed very few differences when the images are different, but the subject is the same. This method was born to perform identity recognition through the face. Since we are interested in identity verification, for the nature of the proposed transactions, in the following Section, we introduce our face encoding manipulation.

3.1.3. Face encoding manipulation

To perform identity verification, a template of each user is needed. Since each features embedding presents, for the same user, small differences related to occlusions, illumination variations, pose, expressions, and so on, we propose to use multiple images to recreate a range of values in which an embedding of each user must be included. For this purpose, we choose to use mean and standard deviation. Given N images of the same user of the template, the mean μ_j and the standard deviation σ_j of each feature is defined as

$$\mu_j = \frac{\sum_{i=1}^{N} x_{ij}}{N}, \qquad \sigma_j = \sqrt{\frac{\sum_{i=1}^{N} |x_{ij} - \mu_j|^2}{N}} \tag{1}$$

where x_{ij} is the ith value of the jth image used to create the template. Iterating along with the features values j, we obtain two arrays, one for the mean and one for the standard deviation. To save the latter in a single object, we will create an array in which the mean values are in the odd positions, and the standard deviation values are in the even. The result is the template array of 256 elements:

$$\tau = \mu_1, \sigma_1, \mu_2, \sigma_2, \dots, \mu_{128}, \sigma_{128}. \tag{2}$$

When the user declares his identity and provides his image to be recognized since the image is single, the mean values will coincide with the features obtained by the CNN, and the standard deviation values will be zeros:

$$id = \mu_1, 0, \mu_2, 0, \dots, \mu_{128}, 0. \tag{3}$$

The details about the identity verification, i.e. the Identity Check, are presented in Section 3.3.

3.2. BNIF algorithm

The obtained face encoding vector is meant to be fused with the RSA private key to obtain robust authentication for signing digital or smart contracts. We use the RSA algorithm because it is the most common public-key cryptography algorithm and, when implemented with suitable key length, currently 2048 bits, it has proved to be computationally difficult to attack. Information fusion is performed according to the Hybrid Information Fusion (BNIF) algorithm proposed in the study conducted by Iovane et al. [46,47], whose approach allows signatures characterized by eligible randomness and robustness against crypto-analytical attacks. This algorithm constructs a signature through biometric and numerical information, transforming the face encoding vector and the RSA modulus into one matrix, which is required to be squared and characterized by an order which varies according to the components related to the two fused vectors.

The BNIF algorithm works as follows:

- Be $a \in Z^m$ and $b \in Z^n$ two vectors, where a is the biometric component and b the product of two prime numbers. Be $s \in Z$:

  $$s = m + n,$$

  with m and n, respectively, the dimensions of the vectors a and b.
- Be $q \in Z$:

  $$q = \left\lceil \sqrt{s} \right\rceil,$$

  a whole number containing s root.
- The padded vectors a_1 and b_1 respectively related to the biometric and the numerical component are defined as:

  $$a_1 \in Z^{m_1},$$
  $$b_1 \in Z^{n_1};$$

  with dimensions $m_1 = m + nz_1$ and $n_1 = n + nz_2$, where $nz_1 = q - mod\,(m, q)$ and $nz_2 = q - mod\,(n, q)$. The padding is performed according to the components of the private key, which identifies the indexes of the values to insert. The first component identifies the index of the first padding value directly, while successive indexes are computed by cyclically adding the current component to the previously retrieved index.
  Vectors a_1 and b_1 are then fragmented into blocks to instantiate the rows of the union matrix U. The number of blocks extracted from the above vectors are:

  $$nblocc\_a_1 = m1/q,$$
  $$nblocc\_b_1 = n1/q.$$

  Furthermore, the overall padding is defined as:

  $$pad = nz_1 + nz_2.$$

- Blocks are inserted into the union matrix U according to the RSA private key. For this purpose, the vector that is chosen first is given by the first component of the key (module 2), which also specifies the number of blocks to insert. Similarly, the number of blocks related to the remaining vector is identified by the following component, and the procedure continues in this sense alternately. In case all the blocks of one vector have been already inserted, the algorithm proceeds by inserting all the remaining blocks of the other vector subsequently.
- In case the result of the above process is a rectangular matrix, the algorithm inserts the required number of rows, or columns, to obtain a squared matrix, according to the following rules:

  $$pad\_tot = q^2 - (m + n),$$
  $$diff = Pad - pad\_tot$$

  which determine the following implications:

  $$diff < 0 \Longrightarrow add\ a\ line,$$
  $$diff = 0 \Longrightarrow no\ padding,$$
  $$diff > 0 \Longrightarrow add\ a\ column.$$

  Also, in this case, the values to insert for the additional padding are identified by the private key. The value of the first component of the key indicates the element of the first line from which the first padding component is taken; the value of the second component of the key identifies the element of the first column from which the second padding component is taken, and so forth.
- The change of columns is guaranteed by the post-product of the matrix U with a permutation matrix P, which is defined according the private key and composed by combining the following matrices:

$$P_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad P_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad P_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$P_3 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad P_4 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad P_5 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

where the ith matrix is selected by the kth component (module 6) of the private and is centered on the main diagonal of the matrix of final permutation. Furthermore, in case the union matrix U cannot be divided by three, it is necessary to compute further padding of the matrix P by adding another diagonal block.

- The product of the two matrices is defined as:

$$F = UP.$$

- Finally, the hybrid face code is obtained by arranging the matrix F along the lines.

As we have seen, the BNIF code depends on the RSA private key, the biometric component, and the numerical code, and the resulting signature seems random to an external malicious user. The original owner of the biometric component can be recognized only through the private key to successfully reverse the BNIF algorithm. In particular, the private key allows transporting the permutation matrix P easily, considering the transpose of single diagonal blocks and remembering that:

$$P_3 = P_4^T$$

and

$$P_4 = P_3^T,$$

whereas for i $= 0, 1, 2, 5$, being symmetric matrices:

$$P_i = P_i^T.$$

union matrix U is given by:

$$U = FP^T;$$

and deleting the possible padding, it is possible to retrieve both the biometric component and the product of the two prime numbers.

3.3. User identification and smart contract execution

To sign a Digital or Smart Contract through the obtained BNIF vector, it is required to identify the contractual parts, compute their related hybrid face codes, and perform final encryption before instantiating the contract. The Current Biometric Input, i.e. the face encoding vector the user acquires for signing the transaction, is compared with a template that summarizes the user's face biometric characteristics, as described in Section 3.1.3. The Biometric Component used for the information fusion with the RSA private is the face encoding vector extracted from the template at the time of user identification. The obtained BNIF vector is then encrypted to protect the signature when instantiating the Digital or Smart Contract.

The procedure for signing the contract, as shown in Fig. 2, is declined into the following phases:

- *Biometric Input Acquisition*: acquisition of the current face encoding vector related to the user involved in signing the Smart Contract. In the event that the user has not signed any contract in the past, before acquiring the current face encoding vector, the construction of the reference template is carried out. In order to guarantee a high level of security and to be sure that the user do not use photo of other subjects to create a template associate to his/her identity, we require the online creation of the template. On the other hand, online template creation led to an high homogeneity of the features involved, because of the same light, accessories, background. For this reason, to improve the further recognition tasks through the heterogeneity of data, we ask the subject to also add some of his/her photo as much as possible different from the actual condition (e.g., outdoor if indoor). We also plan to make this implementation possible also in a secondary moment if the user does not possess' other photo at the present. From a technical point of view, the user's reference template is constructed following the steps presented in Section 3.1.3.
- *Identity Check*: the current face encoding vector is compared with the user reference template in order to verify its identity. If the identification is successfully passed, the Biometric Component will consist of the vector of the expected values related to the biometric components in the reference template. The Identity Check follows the definition, for each component x_i associated to the current face encoding vector, of the following function:

$$v(x_i) = \begin{cases} 1, & \text{if } \mu_i - \sigma_i - 0.005 \leq x_i \leq \mu_i + \sigma_i + 0.005; \\ 0, & \text{otherwise.} \end{cases}$$

where μ_i and σ_i are, respectively, the mean value and the standard deviation related to the ith biometric component in the user reference template. The value ± 0.005 was empirically established as an additional extension to handle outliers. When $v(x_i)$ turns out to be 1, it means that the component x_i is recognized successfully. To make the system able to recognize a user even if the related image presents several environment differences compared to its template, we empirically set the minimum number of components recognized as 80 over 128.
- *RSA key generation*: generation of the product of two prime numbers, the modulus, through the RSA algorithm. The key is chosen according to the 2048 bits length.
- *Information Fusion*: the information fusion between the Biometric Component and the RSA key is carried out according to the BNIF algorithm presented in Section 3.2.
- *Signature encryption and contract instancing*: encryption of the resulting BNIF vector with the SHA-256 algorithm and instancing of the Smart Contract with the obtained signature.

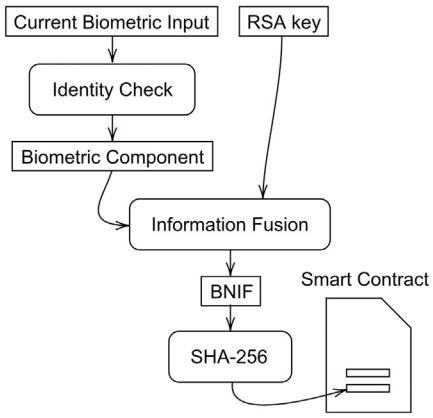

Fig. 2. The Smart Contract signing procedure. The Biometric Component is related to the template of the subject with which to make comparisons for the identification.

Transactions between two parts can be performed through the Smart Contract by generating BNIF signatures for buyer and seller to sign the contract. Both clients instantiate an RSA private key, acquire the related face encoding vector, and produce the signature, the BNIF vector, which will be used to sign the Smart Contract. The buyer can deposit cryptocurrencies into the account specified in the contract, and funds are transferred to the seller at the time both parties accept the execution of the transaction. Transferred coin packs are tagged to prevent double-spending and to be able to take legal measures in case of malicious attempts. Transparency is allowed thanks to the BNIF keys identifying the clients involved in the transaction.

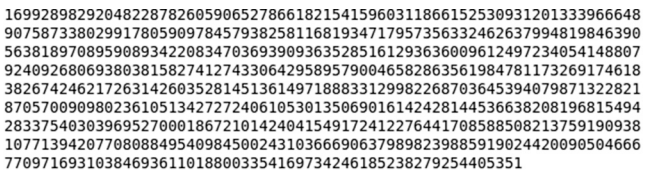

Fig. 3. Alice's RSA key.

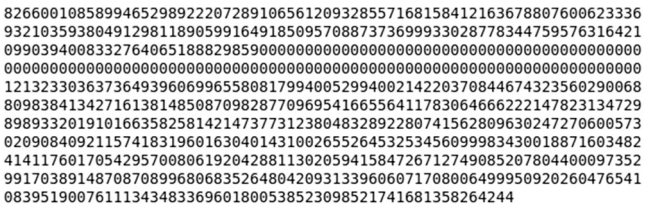

Fig. 4. Alice's BNIF vector. In this example the sequences of zeros are due to certain features of the Biometric Component whose values are close to zero and thus subjected to truncation by means of conversion to integer.

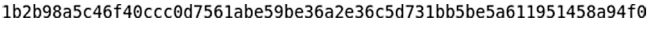

Fig. 5. Alice's SHA-256 hash.

The procedure regarding the implementation of the Smart Contract, assuming Alice wants to transfer funds to Bob, acts according to the following steps:

1. Alice instantiates her RSA_A key, acquires the face encoding, and performs information fusion to compute the $BNIF_A$ (see Figs. 3 and 4);
2. Bob instantiates his RSA_B key, acquires the face encoding vector and performs information fusion to compute the $BNIF_B$;
3. Alice stores her $BNIF_A$ key with SHA-256, producing the hash SHA_A (see Fig. 5);
4. Bob stores his $BNIF_B$ key with SHA-256, producing the hash SHA_B;
5. Alice instantiates the legal part of the Smart Contract and signs it with her SHA_A signature;
6. Bob instantiates the legal part of the Smart Contract and signs it with his SHA_B signature;
7. the Smart Contract goes into the blockchain for the execution;
8. Alice deposits the coins in the Smart Contract account;
9. both Alice and Bob accept the transaction;
10. cryptocurrencies are transferred from Alice's to Bob's wallet (see Fig. 6).

From an operative point of view, transactions between buyer and seller consist of the following phases:

- *Deposit*: performed by the buyer and executed when the seller is waiting for the payment; this phase regards the verification of the buyer BNIF signature and the deposit of the coins into the Smart Contract account.
- *Delivery Confirm*: performed by the buyer and executed when the seller is waiting for the delivery fund confirmation; this phase regards the transfer of the cryptocurrencies to the seller's wallet and the completion of the contract.

3.4. Double spending prevention

The double-spending problem is faced through blockchain on the tamper-proof ledger with biometric authentication. The user registers on the blockchain with the client acquiring the face encoding vector

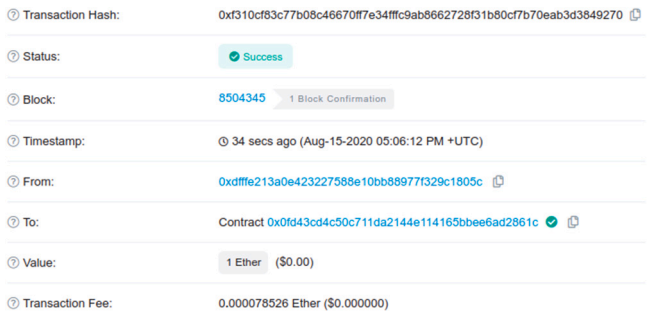

Fig. 6. Alice's fund deposit.

and producing the Biometric Component, which is then fused with the generated RSA key through the presented BNIF algorithm. Finally, the computed signature is enrolled in the blockchain, and the user can perform the desired transactions.

The enrollment process includes the following steps:

1. acquisition of the Current Biometric Input with the user client, RSA private key generation and final information fusion to obtain the BNIF signature;
2. user authentication based on the BNIF signature and miners validation;
3. BNIF signature saving in the blockchain.

In case of invalid authentication, the user cannot sign the transaction. The user can sign transactions with his private key if and only if he has been successfully authenticated with the BNIF signature. The wallet owner only possesses both the Biometric Component and the RSA private key, and both are necessary to invert the BNIF algorithm. On the blockchain, both the product of two prime numbers and the RSA key related to the BNIF are saved and encrypted according to a voting-oriented SHA-256 secret key. Biometric Component and the RSA private key related to the contract owner can be seen if and only if the miners vote for this process. In this way, privacy is guaranteed as long as the user no double-spends his money and the nodes do not vote, in case of a malicious attempt, for the BNIF reverse process.

3.5. Correctness and time complexity

The correctness of the encryption algorithm, as already discussed in the prodromal studies [46,47], is identified in its robustness to crypto-analytical attacks. The BNIF strings subjected to the NIST [48] statistical analysis resulted in random configurations of fixed-size binary sequences, and therefore perceived as noise to intruders. The NIST test was carried out on 100 signatures, evaluating the overall frequency and blocking of the binary digits, cumulative sums, total sum of the sequential executions, and the longest execution. Results showed that all the test instances provided p-values are larger than 0.01 and that the BNIF signature can be considered as a random sequence of symbols.

As far as the time complexity is concerned, the encryption algorithm follows a trend, in the worst case, of $\mathcal{O}(d^2)$, where d is the number of rows in the union matrix U defined in Section 3.2.

4. Experiments and results

In this Section, we will introduce the system configuration and the results obtained for the biometric component and the contract signing.

4.1. Biometrics dataset and results

The dataset used to create the identities is the popular CelebA [49]. CelebA is a publicly available dataset composed of 202 599 celebrity face images, with 10 177 identities and 40 annotated attributes for each image. This dataset results competitive for recognition tasks because images come from various environments like indoor, outdoor, different backgrounds and present pose, illumination and age variations, and significant occlusions. Some examples of CelebA images can be found in Fig. 7. Starting from this heterogeneous dataset, we randomly selected 1000 identities and, for each identity, 10 random images, for a total of 10 000 face images. These 10 000 images were split into 5 sets of data. The first set is composed of 6 images per subject and was used to create the templates as described in Section 3.1.3. Each one of the remaining 4 sets contained one image per subject and were used in identity verification, for a total of 4000 images available for transaction tests. Considering that the subjects will be combined two per time, and a subject cannot perform a transaction with itself, using combination formulas, more than 20 million combinations can be obtained and tested.

As claimed in Section 3.3, we empirically set the acceptability threshold to 80 correspondence over 128. This threshold was obtained to minimize the False Acceptance Rate (FAR) considering an acceptable False Rejection Rate (FRR). The FAR is defined as the percentage of impostors that are wrongly identified as genuine users. On the other hand, FRR is defined as the percentage of genuine users who are denied to authorize the transaction. It is clear that for the characteristics of ECB2, a higher FAR is much more dangerous than a higher FRR. In other words, it is preferable that we ask a user to repeat the identification rather than authorize an impostor to perform a transaction. After extensive experiments conducted on the subset of CelebA defined above, we decided to set the threshold to 80. In this case, the FAR results to be 0.6% and the FRR is 32%. This means that the verification system has an accuracy of 99.4%. At most every third time, the user is asked to repeat the verification using conditions more similar to his template. Even if it might seem restrictive for the user, it should be considered that these results were obtained over the CelebA dataset, which, for its nature, presents considerable inter-class variation. It is fair to assume that in a more controlled environment during the template and input acquisition, the FRR will be significantly lower. The same we can claim for the FAR also increased from inter-class variation, which minimizes the intra-class variations. In other words, our accuracy of 99.4% in verification could only benefit from the use of a more controlled dataset, as well as the FRR.

4.2. Results on contracts signing

Tests conducted over 1000 users show that, on average, the percentage of successfully recognized users is 98%; identification is successful for 65% at the first attempt, 22% at the second, 8% at the third, and 3% at the fourth. In Table 1, standalone tests regard Identity Check phases performed over four biometric instances for each user. In comparison, the multiple attempts test shows the distribution of users who completed the Identity Check by submitting one or more biometric instances.

We noticed that the number of attempt necessary to recognize the user is strongly dependent by his/her images stored in the template. In particular, due to the strong heterogeneity of CelebA, a considerable set of subjects have photos with significant differences. For example, if the subject is a swimming champion, we can find both the photos in which he/she has the swimming cup and the goggles, than his/her photo in formal clothes. Those are case limits present in the Dataset on which we test our algorithm. For this reason, we can undoubtedly claim that in real application scenario, it will perform almost as good as the presented experiments, since a hat and sunglasses worn simultaneously are one of the few causes of the three attempts request.

Table 1
Identity Check tests related to four configurations conducted over 1000 users. Most users are successfully identified within the first two attempts (87%).

	Standalone tests			
	Test 1	Test 2	Test 3	Test 4
Identified users	0.67	0.65	0.68	0.67
Not identified users	0.33	0.35	0.32	0.33
	Multiple attempts test			
	1st att.	2nd att.	3rd att.	4th att.
Identified users	0.65	0.87	0.95	0.98
Not identified users	0.35	0.13	0.05	0.02

Table 2
Execution times related to the preliminary stages of the contractual process. Upper bounds are due to the additional activities of the operating system during the experiments.

Process phase	Execution time (ms)
Template creation	50
Identity Check (including camera acquisition)	1860
Identity Check (excluding camera acquisition)	0.9
RSA key generation	240–1750
Information Fusion	2.7–5.3

Experiments regarding the execution time of the signature process, as shown in Table 2, show that the most computationally expensive phases are those of RSA key generation and camera acquisition. However, since the private key and the BNIF are kept by the client for future transactions, this delay is only considered in the first occurrence of the process, and the same goes for the template creation and Information Fusion phases. The user registration phase, i.e. the stage in which the RSA key is generated, the biometric template is constructed, and the BNIF is computed, takes 292.7–1803.5 ms in total. Every attempt to sign a transaction provided that a new registration does not occur, takes 1860 ms, 1859.1 of which are occupied by the camera processing.

4.3. Discussion

The results prove that ECB2 is a reliable, efficient, and secure digital signature scheme. Privacy and quick authentication are guaranteed by acquiring biometric instances of the face. The identity of participating nodes to the blockchain network and the possibility of conducting effective monitoring against double-spending actions are ensured. The execution of the signature process provides very short times and the proposed framework can also be adopted for devices that have limited computational capabilities.

5. Conclusions and future work

In this article, we have presented an encryption scheme based on face biometrics to increase the level of transparency and security in Smart Contract transactions, without reducing user privacy. The solutions used in the field of face detection and encoding have been shown in detail, together with the methodology through which the template for user identity verification has been built. The fusion of this information, through the illustrated BNIF algorithm, with RSA private key, assumes suitable characteristics in terms of randomness, making the exploit of the signature very difficult. Furthermore, the experiments related to the identification of 1000 users based on various lighting and posture conditions revealed 94% of success within the first four authentication attempts. Finally, the implementation rules and stages of contract execution, together with the related solution adopted to mitigate double-spending attacks, have been explained. Apart from the contribution of the time variable associated with the mining in the blockchain, the execution of the ECB2 authentication scheme and contract signing requires at most 1.86 s. In the future, face biometrics could

Fig. 7. Example of images from CelebA.

be substituted or integrated with other biometrics traits. We also plan to improve the user experience in the steps involving the Biometrics. This can be made creating an interface to guide the user through the Template creation with suggestions to guarantee the heterogeneity of the features. In addition, also during the multiple attempts, we can give the user some suggestions to successfully be recognized, as change illumination, background and so on.

CRediT authorship contribution statement

Carmen Bisogni: Methodology, Validation, Investigation, Writing. **Gerardo Iovane:** Conceptualization, Formal analysis, Supervision. **Riccardo Emanuele Landi:** Software, Validation, Investigation, Writing. **Michele Nappi:** Conceptualization, Resources, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Seshadri SS, Rodriguez D, Subedi M, Choo KR, Ahmed S, Chen Q, et al. IoTCop: A blockchain-based monitoring framework for detection and isolation of malicious devices in internet-of-things systems. IEEE Internet Things J 2020;1.

[2] Soltanisehat L, Alizadeh R, Hao H, Choo KR. Technical, temporal, and spatial research challenges and opportunities in blockchain-based healthcare: A systematic literature review. IEEE Trans Eng Manage 2020;1–16.

[3] Sherman AT, Javani F, Zhang H, Golaszewski E. On the origins and variations of blockchain technologies. IEEE Secur Priv 2019;17(1):72–7.

[4] Lin C, He D, Huang X, Khan MK, Choo K-KR. DCAP: A secure and efficient decentralized conditional anonymous payment system based on blockchain. IEEE Trans Inf Forensics Secur 2020;15:2440–52.

[5] Ma S, Deng Y, He D, Zhang J, Xie X. An efficient NIZK scheme for privacy-preserving transactions over account-model blockchain. IEEE Trans Dependable Secure Comput 2020. http://dx.doi.org/10.1109/TDSC.2020.2969418.

[6] Gervais A, Karame GO, Wüst K, Glykantzis V, Ritzdorf H, Capkun S. On the security and performance of proof of work blockchains. In: proceedings of the 2016 ACM SIGSAC conference on computer and communications security. 2016. p. 3–16.

[7] Natoli C, Gramoli V. The blockchain anomaly. In: 2016 IEEE 15th international symposium on network computing and applications. 2016. p. 310–7.

[8] Georghiades Y, Flolid S, Vishwanath S. HashCore: Proof-of-work functions for general purpose processors. In: 2019 IEEE 39th international conference on distributed computing systems. IEEE; 2019, p. 1951–9.

[9] Vranken H. Sustainability of bitcoin and blockchains. Curr Opin Environ Sustain 2017;28:1–9.

[10] Küfeoğlu S, Özkuran M. Bitcoin mining: A global review of energy and power demand. Energy Res Soc Sci 2019;58:101273.

[11] Khattak HA, Tehreem K, Almogren A, Ameer Z, Din IU, Adnan M. Dynamic pricing in industrial internet of things: Blockchain application for energy management in smart cities. J Inf Secur Appl 2020;55:102615. http://dx.doi.org/10.1016/j.jisa.2020.102615.

[12] Cheng J-C, Lee N-Y, Chi C, Chen Y-H. Blockchain and smart contract for digital certificate. In: 2018 IEEE international conference on applied system invention. IEEE; 2018, p. 1046–51.

[13] Buterin V, et al. A next-generation smart contract and decentralized application platform. white paper 3 (37). 2014.

[14] Böhme R, Christin N, Edelman B, Moore T. Bitcoin: Economics, technology, and governance. J Econ Perspect 2015;29(2):213–38.

[15] Hegedűs P. Towards analyzing the complexity landscape of solidity based ethereum smart contracts. Technologies 2019;7(1):6.

[16] Pilkington M. Blockchain technology: principles and applications. In: Research handbook on digital transformations. Edward Elgar Publishing; 2016.

[17] Zaghloul E, Li T, Mutka MW, Ren J. Bitcoin and blockchain: Security and privacy. IEEE Internet Things J 2020.

[18] Rosenfeld M. Analysis of hashrate-based double spending. 2014, arXiv preprint arXiv:1402.2009.

[19] Feng Q, He D, Zeadally S, Khan MK, Kumar N. A survey on privacy protection in blockchain system. J Netw Comput Appl 2019;126:45–58.

[20] Chen J, Gorbunov S, Micali S, Vlachos G. Algorand agreement: super fast and partition resilient byzantine agreement. IACR cryptology ePrint archive: report 2018, 2018, p. 377.

[21] Mahfouz A, Mahmoud TM, Eldin AS. A survey on behavioral biometric authentication on smartphones. J Inform Secur Appl 2017;37:28–37. http://dx.doi.org/10.1016/j.jisa.2017.10.002.

[22] Kiran MA, Yogeshwari P, Bhavani KV, Ramya T. Biometric authentication: A holistic review. In: 2018 2nd international conference on I-SMAC (IoT in social, mobile, analytics and cloud) (I-SMAC)I-SMAC (IoT in social, mobile, analytics and cloud) (I-SMAC), 2018 2nd international conference on. 2018. p. 428–33.

[23] Buriro A, Crispo B, Conti M. AnswerAuth: A bimodal behavioral biometric-based user authentication scheme for smartphones. J Inform Secur Appl 2019;44:89–103. http://dx.doi.org/10.1016/j.jisa.2018.11.008.

[24] Barra S, Fenu G, De Marsico M, Castiglione A, Nappi M. Have you permission to answer this phone? In: 2018 international workshop on biometrics and forensics. 2018. p. 1–7.

[25] Galdi C, Nappi M, Dugelay J-L. Secure user authentication on smartphones via sensor and face recognition on short video clips. In: Au MHA, Castiglione A, Choo K-KR, Palmieri F, Li K-C, editors. Green, pervasive, and cloud computing. Cham: Springer International Publishing; 2017, p. 15–22.

[26] Zhang Q, Li H, Sun Z, Tan T. Deep feature fusion for iris and periocular biometrics on mobile devices. IEEE Trans Inf Forensics Secur 2018;13(11):2897–912.

[27] Fenu G, Marras M, Boratto L. A multi-biometric system for continuous student authentication in e-learning platforms. Pattern Recognit Lett 2018;113:83–92, Integrating Biometrics and Forensics.

[28] Choudhary SK, Naik AK. Multimodal biometric authentication with secured templates — A review. In: 2019 3rd international conference on trends in electronics and informatics. 2019. p. 1062–9.

[29] Abate AF, Nappi M, Ricciardi S. Smartphone enabled person authentication based on ear biometrics and arm gesture. In: 2016 IEEE international conference on systems, man, and cybernetics. 2016. p. 003719–24.

[30] Bentahar A, Meraoumia A, Bendjenna H, Chitroub S, Zeroual A. Biometric cryptosystem scheme for internet of things using fuzzy commitment principle. In: 2018 international conference on signal, image, vision and their applications. 2018. p. 1–6.

[31] Nair VS, Reshmypriya GN, Rubeena MM, Fasila KA. Multibiometric cryptosystem based on decision level fusion for file uploading in cloud. In: 2017 international conference on recent advances in electronics and communication technology. 2017. p. 29–32.

[32] Ding X, Fang H, Zhang Z, Choo KR, Jin H. Privacy-preserving feature extraction via adversarial training. IEEE Trans Knowl Data Eng 2020;1.

[33] Albakri A, Mokbel C. Convolutional neural network biometric cryptosystem for the protection of the blockchain's private key. Procedia Comput Sci 2019;160:235–40, The 10th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN-2019)/The 9th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2019)/Affiliated Workshops.

[34] Páez R, Pérez M, Ramírez G, Montes J, Bouvarel L. An architecture for biometric electronic identification document system based on blockchain. Future Internet 2020;12(1):10.

[35] Liu Y, Sun G, Schuckers S. Enabling secure and privacy preserving identity management via smart contract. In: 2019 IEEE conference on communications and network security. IEEE; 2019, p. 1–8.
[36] Zhang H, Chen X, Lan X, Jin H, Cao Q. BTCAS: A blockchain-based thoroughly cross-domain authentication scheme. J Inform Secur Appl 2020;55:102538. http://dx.doi.org/10.1016/j.jisa.2020.102538.
[37] Rivera R, Robledo JG, Larios VM, Avalos JM. How digital identity on blockchain can contribute in a smart city environment. In: 2017 international smart cities conference. 2017. p. 1–4.
[38] Osipkov I, Vasserman EY, Hopper N, Kim Y. Combating double-spending using cooperative P2P systems. In: 27th international conference on distributed computing systems. IEEE; 2007, p. 41.
[39] Yu X, Shiwen MT, Li Y, Huijie RD. Fair deposits against double-spending for bitcoin transactions. In: 2017 IEEE conference on dependable and secure computing. IEEE; 2017, p. 44–51.
[40] Lee HJ, Choi MS, Rhee CS. Traceability of double spending in secure electronic cash system. In: 2003 international conference on computer networks and mobile computing, 2003. IEEE; 2003, p. 330–3.
[41] Lee H, Shin M, Kim KS, Kang Y, Kim J. Recipient-oriented transaction for preventing double spending attacks in private blockchain. In: 2018 15th annual IEEE international conference on sensing, communication, and networking. IEEE; 2018, p. 1–2.
[42] Nicolas K, Wang Y. A novel double spending attack countermeasure in blockchain. In: 2019 IEEE 10th annual ubiquitous computing, electronics mobile communication conference. 2019. p. 0383–8.
[43] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition, vol. 1. 2005. p. 886–93.
[44] Déniz O, Bueno G, Salido J, De la Torre F. Face recognition using histograms of oriented gradients. Pattern Recognit Lett 2011;32(12):1598–603.
[45] Schroff F, Kalenichenko D, Philbin J. FaceNet: A unified embedding for face recognition and clustering. In: 2015 IEEE conference on computer vision and pattern recognition. 2015. p. 815–23.
[46] Iovane G, Puccio L, Lamponi G, Amorosia A. Electronic access key based on innovative Information Fusion technique involving prime numbers and biometric data. J. Discrete Math. Sci. Cryptogr. 2011;14(3):207–25.
[47] Iovane G, Bisogni C, De Maio L, Nappi M. An encryption approach using information fusion techniques involving prime numbers and face biometrics. IEEE Trans Sustain Comput 2020;5(2):260–7.
[48] Rukhin A, Soto J, Nechvatal J, Smid M, Barker E. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Tech. rep., Booz-allen and hamilton inc mclean va; 2001.
[49] Liu Z, Luo P, Wang X, Tang X. Deep learning face attributes in the wild. In: Proceedings of international conference on computer vision. 2015.