 doi:10.20944/preprints201810.0618.v1

Peer-reviewed version available at *Cryptography* **2018**, *2*, 39; doi:10.3390/cryptography2040039

Type of the Paper (Article)

Security and Cryptographic Challenges for Authentication based on Biometrics Data

Stefania Loredana Nita[1], Marius Iulian Mihailescu[2,*] and Valentin Corneliu Pau[3]

[1] University of Bucharest, Department of Computer Science; stefanialoredanani@gmail.com
[2] RCCL, Miami, Florida; marius.mihailescu@hotmail.com
[3] The Academy of Romanian Scientists, Bucharest, Romania; v_pau@utm.ro
* Correspondence: marius.mihailescu@hotmail.com; Tel.: +40740030310

Abstract: Authentication systems based on biometrics characteristics and data represents one of the most important trend in the evolution of our world. In the near future, biometrics systems will be everywhere in the society, such as government, education, smart cities, banks etc. Due to its uniqueness characteristic, biometrics systems will become also vulnerable, privacy being one of the most important challenge. The classic cryptographic primitives are not sufficient to assure a strong level of secureness for privacy. The following work paper represents an effort to present the main cryptographic techniques and algorithms that can give us the possibility to raise a certain level of secureness for privacy. We will show their own challenges (strengths and weaknesses). We will demonstrate how we can use the most common and well-known techniques and algorithms in order to get a maximum efficiency and a high level in assuring the integrity of the biometrics data.

Keywords: classification, machine learning, chaos-based cryptography, Hadoop, data clustering, biometrics.

---

1. Introduction

Biometric represents a science through which a system can identify in a unique way the individual based on his physiological (face, fingerprint, iris, hand geometry, retina etc.) and behavioral (voice, gait, signature, keystroke etc.) traits. Due to the rich of applications and systems that occur at this moment on the market, the authentication technology is very widespread from e-commerce applications, door access, and smart city technologies to Internet-of-Things technologies and applications. Once these solutions where declared as providing better security in authentication, automatically issues with privacy preserving and data integrity has been found didn't stop to occur. Indeed, biometrics systems become more convenient to be used as compared with different classic authentication systems such as token based (e.g. ID cars) or knowledge based (e.g. passwords) [12]. A biometric-based authentication system is using two different operating modes: identification and verification modes. These two different modes are very important as they represent security gaps, which are exploited by unwanted users (e.g. hackers/crackers).

What is happening in identification mode?

In identification mode, the system has a clear goal to carry out the one-to-many comparison meant to set up the individual identity. With other words, the user's identity and the templates that are stored in the database are compared and based on the result a decision will be made. The purpose of the identification process is to answer to the question "Who am I?" If we are talking about the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 26 October 2018 doi:10.20944/preprints201810.0618.v1

Peer-reviewed version available at Cryptography 2018, 2, 39; doi:10.3390/cryptography2040039

implementation process, we can say that they are time consuming for deploying and needs a huge amount of time for processing in order to find the proper match within the database [12].

What is happening in verification mode?

In verification mode, the system is carrying out the one-to-one comparison. This type o comparison is used to set up the individual identity. The user is claiming that identity and the system has the duty to verify if the claim is genuine or forged. The goal of the verification process is to answer to the question "Am I who I say I am?" [12].

An interesting question raised by different professionals is "What measurements for the biological characteristics is making them to be qualified as a biometric?" Most of the human physiological and/or behavioral characteristics used in the system for authentication as long as they are satisfying the following requirements:

- *Performance:* this requirement is quite important as the characteristic should be enough invariant. The respect has to be assure for the matching criterion over a period.
- *Distinctiveness:* by choosing two persons should be sufficient different in terms of the characteristic.
- *Universality:* the criteria consist in its unique characteristic that has to be for each person.
- *Collectability:* the requirement is a metric that is quantitatively measured.

When we are dealing with a real life biometric system, there is a number of issues that has to be taken into consideration in order to take a complete advantage of the full system and to combat those security issues that could occur on different section:

- *Performance*, the accuracy and speed, two main characteristics that refers to the achievable recognition, are required to achieve the desired recognition accuracy and speed. Also, operational and environmental factors are affecting the accuracy and speed;
- *Acceptability*, a factor that will indicates which people are willing to accept the use of a particular biometric identity in terms of characteristic using in a daily lives;
- *Circumvention* reflects how easy is to fool a system using different methods meant to steal data and to corrupt the integrity of the data.

As we will be able to see, the following content of the present work paper will cover the most important aspects of security flaws that are raised by each of the components of the biometric system. The work paper will show and demonstrate theoretical and practical two main benefits: (1) we will demonstrate how we can protect the integrity of the biometric data using Machine Learning Classification and what benefits we can obtain. Another benefit, and (2) applying chaos-based cryptography over encrypted biometric encrypted data. The two main method for secure and guarantee the integrity of the biometric encrypted data will be demonstrated in a professional system architecture based on Hadoop and Data clustering.

The new challenges from the last two years represents a very important alarm signal for both, academy and business environments. The security threats and gaps in assuring the privacy and integrity found, represents one of the most important occasion from which we have to take the maximum advantage in creating new theoretical and practical security frameworks.

Below we have underline the main new challenges that in our opinion will create new research directions for security and cryptography field, such as:

- Using Machine Learning Classification over the encrypted biometric data;
- Encryption of biometric data in a Data Clustering environment;

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 26 October 2018 doi:10.20944/preprints201810.0618.v1

Peer-reviewed version available at *Cryptography* **2018**, *2*, 39; doi:10.3390/cryptography2040039

- Encryption of biometric data using Chaos-based cryptography;

In order to be able to follow the ideas and to understand how they will be applied in a real environment, we need to understand main four modules of the biometric system and their vulnerabilities.

The mentioned challenges raised above will be treated and the issues solved using Chaos-based cryptography and machine-learning classification applied in cluster environment using authentication based biometrics.

2. Machine Learning Classification Over Biometric Encrypted Data

In this section, we present two machine-learning techniques applied on biometric encrypted data: hyperplane decision and Naïve Bayes.

2.1. Preliminaries

2.1.1. Machine learning

"Machine learning is a field of computer science that gives the ability to learn without being explicitly programmed" – Samuel Arthur [1].

The below techniques have been applied and implemented in a software solution in order to simulate the methods on biometric data. The learning techniques implemented are described below. The solution software was implemented in .NET Framework 4.5 using C# and Microsoft SQL Server 2016. Due to the status of the software application, we cannot provide the source code in this work paper. For those who are interested about the application they can visit the web page https://www.researchgate.net/project/Biometrics-Analysis-Tool.

There are four types of learning techniques implemented are:

- *Supervised learning* is a type of inductive learning based on training sets, in which, the agent receives a set of inputs and their corresponding outputs. The task of the agent is to learn the links between every input and its corresponding output and to generate a template function that will be able to solve problems for new inputs.
- *Semi-supervised learning*. In this type of learning, the agent receives an incomplete training set.
- *Unsupervised learning* is not using training sets, but the agent needs to discover on its own different patterns in dataset.
- *Reinforcement learning* is a type of learning in which the training data is given as feedback for the agent, such that if its output is "good" it receives a reward, otherwise it receives a punishment. The target of the agent is to maximize its reward, providing better and better outputs. The meaning of "good" output is different depending on the environment in which the agent is used [4].

Classification represents a machine learning technique (included in supervised learning) in which the inputs are divided into two or more classes. The input is a *feature vector* $v = (v_1, \ldots, v_n) \in \mathbb{R}^n$ that will be classified by applying a classification function $f_m: \mathbb{R}^n \to \{x_1, \ldots, x_c\}$ on v, and the output is $x_{c^*} = f_m(v)$, where $c^* \in \{1, \ldots, c\}$; x_{c^*} is the class in which v falls, based on model m.

In this case, the feature vector will represent all the biometric data over which the classification data is applied.

$$v = (v_1, \ldots, v_n) \in \mathbb{R}^n \quad (1)$$

$$biometric\ vector = (b_v) = (b_{v_1}, \ldots., b_{v_n})$$

$$f_m: \mathbb{R}^n \longrightarrow \{x_1, \ldots, x_c\} \text{ over } b_v \text{ and the output is } x_{c^*} = f_m(v) \text{ where } c^* \in \{1, \ldots, c\}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 26 October 2018 doi:10.20944/preprints201810.0618.v1

Peer-reviewed version available at *Cryptography* **2018**, *2*, 39; doi:10.3390/cryptography2040039

The x_{c^*} represents the class in which b_v falls, being based on model m.

Two important classification algorithms are Naïve Bayes and hyperplane decision-based classifier.

Naïve Bayes. The model m of this classifier is based on probabilities: the probability that class x_i occurs is $\{p(X = x_i)\}_{i=1}^{c}$ and the probability that the element v_j of v occurs in the particular class x_i. The classification function is:

$$x_{c^*} = \max_{i\in\{1,\dots,c\}} p(X = x_i | V = v) \tag{2}$$

$$= \max_{i\in\{1,\dots,c\}} p(X = x_i, V_1 = v_1, \dots, V_n = v_n)$$

In order to obtain the second equality Bayes rules was applied.

In biometrics, the Naïve Bayes will help us to classify the biometric data and to obtain a faster time for processing. The classification is done based on the biometric characteristics.

Hyperplane decision. The model m contains c vectors in $\mathbb{R}^n$ $(m = \{m_i\}_{i=1}^{c})$ and the classification function is [2]:

$$x_{c^*} = \max_{i\in\{1,\dots,c\}} \langle m_i, v \rangle, \tag{3}$$

where $\langle m_i, v \rangle$ represents the inner product between m_i and v, in a hypothesis space H having defined an inner product$\langle \cdot,\cdot \rangle$.

In biometrics, the hyperplane decision will help us to identify and to classify in a much better way the biometrics 3D representations of face recognition.

2.1.2. Cryptography

Cryptosystem. It has more components: plaintext space P, ciphertext space C, key space K, and encryption functions $E = \{E_k | \, k \in K\}$, and decryption functions $D = \{D_k \, | k \in K\}$:

$$\forall e \in K \; \exists d \in K \text{ such that } D_d\big(E_e(m)\big) = m, \forall m \in P \tag{4}$$

Encryption scheme. It represents a particularization of a cryptosystem. There are two types of encryption schemes: *symmetric key* schemes (in which the same key is used for encryption and also decryption) and *public key* encryption schemes (in which a public key is used to encrypt messages and a private key is used to decrypt messages).

Homomorphic encryption. It is a special type of encryption which allows to apply functions over encrypted message, resulting also an encrypted result, which, when decrypted it is the same as applying the same function over unencrypted message. This is a powerful cryptographic technique, because it increases the security of data, as the operations are applied on encrypted data, resulting an encrypted output, which will be decrypted only the users that own the decryption key. Unfortunately, at this moment a *fully homomorphic encryption* scheme (that allows to apply *any* function on encrypted data) does not exist, because the existing computational capabilities are overwhelm. An example of partial homomorphic encryption is RSA cryptosystem [3], where the homomorphic operation is multiplication:

$$E(m_1) \cdot E(m_2) = m_1^r m_2^r \bmod n = (m_1 m_2)^r \bmod n = E(m_1 \cdot m_2), \tag{5}$$

where the public key is modulus n and the exponent is r, and encryption function is

$$E(m) = m^r \bmod n \tag{6}$$

Applying in biometrics, the function has to be changed in order to allow the operations on bits to be applied for each bit separately.

2.2. Techniques

In this section, we present the constructions of the above classification techniques proposed and improved by the authors of [4], such that they could be applied on encrypted data. The authors of [4] have shown that their methods are successfully applied on large real datasets, including in face detection, which became widely used in biometric authentication.

2.2.1. Auxiliary algorithms

In [4] the cryptosystems that have been used are Quadratic Residuosity [5] (where $P = \mathbb{Z}_2$) and Paillier cryptosystem (where $P = \mathbb{Z}_N$ and N is modulus of Paillier) [6]. Further, notation $(b)_{QR}$ means the bit b is encrypted with Quadratic Residuosity, $(m)_{PA}$ means the integer m is encrypted with Paillier, SK_{QR} and PK_{QR} are secret and public key for Quadratic Residuosity and SK_{PA} and PK_{PA} are secret and public key for Paillier.
The entities implied in this sections are two parties A and B for building blocks and C (client) and S (server) for classifiers.
Authors of [4] have defined some auxiliary operations: comparison with unencrypted inputs, comparison with encrypted inputs, reversed comparison over encrypted data, negative integers comparison and sign determination, which will be used in protocols defined below. The below algorithm will demonstrate how the encryption over biometric data can work bit by bit.

Algorithm 1 – max over encrypted data [4]

Input A: k integers encrypted using Paillier ($(a_1)_{PA}, \ldots, (a_k)_{PA}$), the length l of a_i (in bits), PK_{QR} and SK_{QR}
Input B: SK_{PA}, PK_{PA}, the length l in bits
Output A: $\max a_i$

A: generate random permutation π over $\{1, \ldots, k\}$
A: $(max)_{PA} := a_{\pi(i)}$
B: m := 1
for i = 2 **to** k **do**
    $b_i = \max \leq a_{\pi(i)}$
    A: randomly generate integers $r_i, s_i := (0, 2^{\lambda+l}) \cap \mathbb{Z}$
    A: $(m_i')_{PA} := (max)_{PA} \cdot (r_i)_{PA}$ $\triangleright\ m_i' = max + r_i$
    A: $(a_i')_{PA} := (a'_{\pi(i)})_{PA} \cdot (s_i)_{PA}$ $\triangleright\ a_i' = a_{\pi(i)} + s_i$
    A: send $(m_i')_{PA}$ and $(a_i')_{PA}$ to B
    if b_i is true **then**
        B: $m := i$
        B: $(v_i)_{PA} := Refresh(a_i')_{PA}$ $\triangleright\ v_i = a_i'$
    else

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 26 October 2018 doi:10.20944/preprints201810.0618.v1

Peer-reviewed version available at Cryptography 2018, 2, 39; doi:10.3390/cryptography2040039

B: $(v_i)_{PA} := Refresh(m'_i)_{PA}$ $\rhd v_i = m'_i$

end if

B: send $(v_i)_{PA}$ to A

B: send $(b_i)_{PA}$

A: $(max)_{PA} := (v)_{PA} \cdot (g^{-1} \cdot (b_i)_{PA})^{r_i} \cdot ((b_i)_{PA})^{s_i}$ $\rhd max = v_i + (b_i - 1) \cdot r_i - b_i \cdot t_i$

end for

B: send m to A

A: output $\pi^{-1}(m)$

184 Table 1 - max over encrypted data [4]

185 The next algorithm will show how to change an encryption scheme E_1into and encryption scheme
186 E_2, both having the same plaintext size M. The authors of [4] supposed that E_1 and E_2 are
187 additively homomorphic, and semantically secure. In the above algorithm $(c)_i$ means c is
188 encrypted using encryption scheme $E_i, i \in \{1, 2\}$.

Algorithm 2 – Change encryption scheme [4]

Input A: $(c)_1$, PK_1 and PK_2

Input B: SK_1, SK_2

Output A: $(c)_2$

A: pick $r \in M$

A: send $(c')_1 := (c)_1 \cdot (r)_1$ to B

B: decrypt $(c')_1$ and re-encrypt with E_2

B: send $(c')_2$ to A

A: $(c)_2 = (c')_2 \cdot (r)_2{}^{-1}$

A: output $(c)_2$

189 Table 2 – changing encryption scheme [4]

Algorithm 3 – Private inner product [4]

Input A: $a = (a_1, \dots, a_d) \in \mathbb{Z}^d, PK_{PA}$

Input B: $b = (b_1, \dots, b_d) \in \mathbb{Z}^d, SK_{PA}$

Output A: $(\langle a, b \rangle)_{PA}$

B: encrypt b

B: send $(b_i)_{PA}$ to A

A: compute $(v)_{PA} = \prod_i (b_i)_{PA}^{x_i} \bmod N^2$ $\rhd v = \sum b_i a_i$

A: re-randomize

A: output $(v)_{PA}$

190 Table 3 – Private inner product [4]

191 The authors of [4] proved that these algorithms are secured in *honest but curious* model.

192 **2.2.2. Naïve Bayes**

193 In order to apply Naïve Bayes on encrypted data, there are needed more transformations. In [4], it
194 was working with the logarithm of the probability distribution:

195 $$x_c^* = \max_{i \in \{1,\dots,c\}} \log p(X = x_i | V = v)$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 26 October 2018 doi:10.20944/preprints201810.0618.v1

Peer-reviewed version available at Cryptography 2018, 2, 39; doi:10.3390/cryptography2040039

$$= \max_{i \in \{1,\dots,c\}} \left\{ \log p(X = x_i) + \sum_{j=1}^{n} \log p(V_j = v_j | C = c_i) \right\} \quad (7)$$

The two types of auxiliary table were used:

- One table in which are stored values $P(i) = \lceil K \log p(X = x_i) \rceil$
- One table for every feature j and class i: $T_{i,j}(q) \approx \lceil K \log p(Y_j = q | V = v_i) \rceil, \forall q \in D_j$

where D_j represent the domain of possible values of v_j, and $K \in \mathbb{N}$ is a constant.

Now to apply Naïve Bayes on encrypted data, the client needs to compute $(p_i)_{PA} = (P_i)_{PA} \prod_{j=1}^{n} \left(T_{i,j}(v_j)\right)_{PA}$, and then uses Algorithm 1 to obtain $\max p_i$.

2.2.3. Hyperplane decision

For hyperplane decision the things are quite simple, because the client will calculate $(\langle m_i, v \rangle)_{PA}, i \in \{i, \dots, c\}$, using Algorithm 3 for inner product, and then Algorithm 1 is used to compute *max* on encrypted inner product.

3. Data Clustering in Cloud Computing

3.1. Preliminaries

3.1.1. Fixed-width clustering algorithm

Clustering is a machine learning technique (included in unsupervised learning) in which a set of objects is partitioned into groups called clusters. Different from classification, in clustering there are no predefined clusters, thus the algorithm needs to find relationships or similarities between objects [8].

Fixed-width clustering (FWC) algorithm is based on a distance measure. The steps of FWC are the following:

1. From a given dataset D with an established cluster width w, generate a random set of m clusters: $C_i, i \in \{1, \dots, m\}$.
2. Compute Euclidean distance between every point $p_j, j = \{1, \dots, n\}$ and every cluster C_i, using the formula:

$$d_{ij}(c_i, p_j) = \sqrt{\left(c_{ix} - p_{jx}\right)^2 + \left(c_{iy} - p_{jy}\right)^2} \quad (8)$$

3. If $d_{ij}(c_i, p_j) \leq w$, then p_j belongs to C_i cluster; adjust the centroid of C_i by computing the mean of the points that C_i contains at this moment, using the formula (n is the number of points in C_i):

$$centroid(C_i) = \left(\frac{p_{1x} + \dots + p_{nx}}{n}, \frac{p_{1y} + \dots + p_{ny}}{n} \right) \quad (9)$$

4. If $d_{ij}(c_i, p_j) > w$, then p_j is the new centroid of C_i.
5. Reiterate steps 2, 3, 4 until the end of D.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 26 October 2018 doi:10.20944/preprints201810.0618.v1

Peer-reviewed version available at Cryptography 2018, 2, 39; doi:10.3390/cryptography2040039

3.1.2. MapReduce

MapReduce [7] is a programming model for large datasets processing, which works exclusively on $(key, value)$ pairs. It consists in three steps: $map, shuffle, reduce$, and the user needs to define map and $reduce$ functions. The basic idea is that map function takes as input a set of $(key, value)$ pairs and outputs an intermediary set of $(key, value)$. These outputs are processed by $shuffle$ function, which groups all intermediary values corresponding to an intermediary key, and sends them to $reduce$ function. The $reduce$ function will try to join these values in order to decrease the number of values. Usually just one value will result for the input key per $reduce$ invocation.

A well-known software framework is Hadoop MapReduce [8] that allows processing in parallel large datasets (multi-terabytes of data). We will not give more details, for a comprehensive view please read [8].

3.2. FWC algorithm with MapReduce

In [9] the authors propose a distributed version of FWC algorithm, using MapReduce on a large number of virtual machines (VMs), as follows:

- Inputs: dataset D and the set of clusters $C_1, \dots, C_m$
- Partitioning: the N points of D are allocated to the M available VMs (if $\frac{M}{N}$ is not integer, then the remaining points are allocated to the last VM).
- Map function. The input is dataset D encrypted and kept into Hadoop Distributed File System (HDFS) as $(key, value)$ pairs, where key represents the position of $value$ in a data file and $value$ represents the encryption of numerical of the data point. The data files are global and sent to all mappers. The map function in proposed model computes the squared Euclidean distance (in order to shun the square root):

$$E\left(d_{ij}(c_i, p_j)\right) = \left(E(c_{ix}) - E(p_{jx})\right)^2 + \left(E(c_{iy}) - E(p_{jy})\right)^2 \tag{10}$$

  The $output$ of map is a set of $(key, value)$ pairs, where key is the position of $value$ into a data file and $value$ is the distance $E(d_{ij}) \stackrel{\text{def}}{=} E\left(d_{ij}(c_i, p_j)\right)$. Note that $E(v)$ means value v is encrypted using function E.
- $Reduce$ function. The output of a map function becomes the input for a $reduce$ function. The $reduce$ function needs to find a minimum distance between every point $E(p_j)$ and every centroid $E(c_i)$ and then to put data point $E(p_j)$ into corresponding cluster (the one that corresponds to the minimum distance) [8].

Next, we present the pseudo-code algorithms for map and $reduce$ functions as the authors provide them in [9].

Algorithm 4 – Map function [9]

Input: encrypted dataset $E(D)$

Output: $< key, ctxt(value) > \rightarrow < index, encrypted\ distance\ E(d_{ij}) >$

Initialization: Choose a random set of clusters $c_1, \dots c_m$ from a given dataset $E(D)$

index = 0

for (i=0 to $D.length$) do

  for (j=0 to $c.length$) do

    $E(d_{ij}) = computeDist(E(p_j), E(c_i))$

    $index\ = \ i + j$

  end

end

End

Take index as key

Construct value as an encrypted numerical value $E(d_{ij})$

Output $< key, ctxt(value) >$ pair

Table 4 – *map* function for distributed version for FWC

Algorithm 5 – Reduce function [9]

Input: $< index, encrypted\ distance\ E(d_{ij}) >$

Output: $< ctxt(key), ctxt(value) > \rightarrow < E(c_i), E(p_j) >$

Initialization: $E(minDis)$

for (i=0 to $D.length$) do

  $E(minDis) = \min(d_{i1}, \dots, d_{ij})$

  if $E(minDis) \leq w$ then

    $assign(E(p_j), E(c_i))$

    $update(E(c_i))$

  else

    $createNewCluster\left(E(p_j)\right)$

  end

end

Take $E(p_j)$ as key

Construct value as a numerical value $E(c_i)$

Output $< ctxt(key), ctxt(value) >$ pair

Table 4 – *reduce* function for distributed version for FWC

We can easily see the potential of this approach of FWC algorithm in biometry. For example, it could be used on large datasets of human face datasets in order to find different groups among the subjects by analyzing the face images. An advantage of the proposed model in [9] is that computations are made on encrypted data, assuring data privacy.

4. Our proposed solution

This section will describe the proposed solution, which has been implemented already practical, and the results obtained where positive in encrypting data. The idea was developed and presented in details in [10] and [11].

The below scheme can be adapted with success for all the mentioned algorithms mentioned above. All the algorithms were been implemented with success in C, C++, Java, and C#. The source code will be available at the link mentioned below.

The scheme has been created in a very flexible manner giving the possibility to be adapted accordingly to any type of algorithms especially for those ones from machine learning field.

The algorithm has two components:

1 – The session key algorithm, and

2 – The scheme used for enrollment with data integrity checking and validating for the biometric data.

In Figure 4 – 1, we can see that the permutation functions are applied on all the components that play a role in the enrollment scheme. The permutation function will allow the original value to be re-arranged in such a way that will be very difficult to understand something from permuted value. The original values influence the encrypted system. The content of data could be protected using the complex active action, which comes from the characteristic of the system mentioned above. The behavior will result into a random series, which could be utilized to data encryption from secret communication. Thus, an appropriate key controls encryption or decryption of data content. Machine learning is also an appropriate choice for using a hash function due to one-way property. The model described in Figure 4 - 2 stores the encrypted Biometric Template using Session Key. A session key is the process that generates a random encoding and decoding key which ensures the privacy of a session of communications. A similar example can be found in [5] and [6].

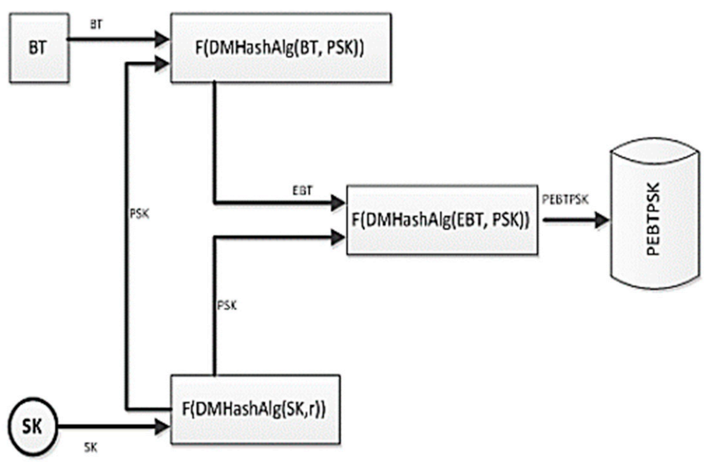

Fig. 4 – 1. Data integrity checking and validation for biometric data

As some notations for the scheme presented in Figure 4 – 1, the followings have to be considered:

- BT – biometric data

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 26 October 2018 doi:10.20944/preprints201810.0618.v1

Peer-reviewed version available at Cryptography 2018, 2, 39; doi:10.3390/cryptography2040039

- SK – represents the session key, which is generated using a one of the algorithms, which were presented in Chapter 2 and 3 and combined, with elements of machine learning described in Section 2.1.1. The BT contains the biometric vector (b_v) as we have discussed previous.
- PSK – represents the permuted session key, which is used to generate the extended version of permuted transformation of the session, key (SK). F(SK) represents a function used for permutation which can be used with any hash function generated based on the main ideas presented above.
- EBT – represents the biometric data, which are encrypted. In order to generate the biometric template encrypted, the hash function construction is applied F(DMHashAlg(BT, PSK)). The hash function is based on a simple XOR function and both functions F(DMHashAlg(SK,r)) and F(DMHashAlg(EBt, PSL)) functions are used together beside the hash functions with the permutation of the bits SK, EBT and ESK.
- PEBTPSK – the permuted biometric data and also the permuted session key (SK) will be used to generate the final step in order to concatenate the biometric template F(DMHashAlg(EBT, PSL)). In order to assure the decryption process, the biometric pattern is using the functions and session key, which has been used to encrypt the template.

In the end, the presented idea represents a new scheme for assuring the data integrity of the biometric data. The algorithm has been implemented with success and it was tested with positive results on a set of 700 unique biometric data of 523 subjects.
To access the source code of the application, please, visit the following web address: https://www.researchgate.net/project/Biometrics-Analysis-Tool.

5. Conclusions

Applying cryptographic mechanisms and machine learning over biometric data is not an easy task to accomplish. This fact can be due to the high complexity of how the biometric data are scanned and read from the user and transfer into the system. Every time that the evolution of technology is making important advances the complexity of assuring the security and integrity of the data is becoming a real pain for developers and designers of authentication systems on biometrics.
We have proven that applying machine learning techniques and cryptography mechanisms can be a task that can be accomplished. The most important aspect on which we have focused in this work paper is how the parameters can be represented and adapted within the algorithms and techniques used in cryptography and machine-learning. The complexity of the algorithms and the time processing can represented a problem but not so critical at this time for any of the system configuration based on the highest requirements possible.
The algorithms presented in Chapter 3 and Chapter 4 where implemented with success and we have obtained positive results.
All the results can be viewed at https://www.researchgate.net/project/Biometrics-Analysis-Tool. *The software used in analysis and simulation is presented at the mentioned web site. Due to copyrights, we didn't make the source code available at this moment.*

References

1. Samuel AL: Some studies in machine learning using the game of checkers. IBM Journal of research and development. 2000;44.1.2:206-226.
2. Nasrabadi NM: Pattern recognition and machine learning. Journal of electronic imaging. 2007; 16.4: 049901.

Peer-reviewed version available at *Cryptography* **2018**, *2*, 39; doi:10.3390/cryptography2040039

3. Rivest RL, Shamir A, Adleman L: A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM. 1976;21(2):120-126.
4. Bost R, Popa RA, Tu S, Goldwasser S: Machine learning classification over encrypted data. In NDSS. 2015;4324:4325.
5. Goldwasser S, Micali S: Probabilistic encryption & how to play mental poker keeping secret all partial information. In: Proceedings of the fourteenth annual ACM symposium on Theory of computing; May 1982: ACM; 1982. p. 365-377
6. Paillier P: Public-key cryptosystems based on composite degree residuosity classes. In: International Conference on the Theory and Applications of Cryptographic Techniques, May 1999; Berlin, Heidelberg: Springer; 1999. p. 223-238
7. Dean J, Ghemawat S: MapReduce: simplified data processing on large clusters. Communications of the ACM. 2007; 51(1):107-113.
8. Taylor RC: An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics. In BMC bioinformatics. 2010:11(12):S1.
9. Alabdulatif A, Khalil I, Reynolds M, Kumarage H, Yi X: Privacy-preserving data clustering in cloud computing based on fully homomorphic encryption. *In PACIS 2017: Societal Transformation Through IS/IT*. Association for Information Systems (AIS). 2017:1-13.
10. Mihailescu Marius Iulian, Nita Stefania Loredana, *Security of Biometrics Authentication Protocols. Theory and Practice Applications*, LAP LAMBERT Academic Publishing, ISBN-10: 3659777498, ISBN-13: 978-3659777493. Publishing Date: 11 September 2015, Language: English, 232 pages.
11. Mihailescu Marius Iulian, New Enrollment Scheme for Biometric Template using Hash Chaos-based cryptography, Elsevier - Procedia Engineering, Volume 69, 2014, pages 1459-1468, ISSN: 1877-7058.
12. Mihailescu Marius Iulian, Racuciu Ciprian, Grecu Dan Laurentiu, Nita Stefania Loredana, A Multi-Factor Authentication Scheme Including Biometric Characteristics as One Factor, 1st International Conference Sea-Conf, pp.:348-353, *Mircea cel Batran* Naval Academy Scientific Bulletin, Volume XVIII, Issue 1, ISSN: 2392-8956, ISSN-L: 1454-864X, CNCS Code: 884, 14-16 May 2015, Constanta, Romania.