Biometric Transaction Authentication using Smartphones

Martin Stokkenes[1], Raghavendra Ramachandra[1], Christoph Busch[1]

Abstract: Secure and robust authentication of users and customers is critical, as an increasing number of services from banks, health and government sectors are made available to people as online services. Recent development in the area of biometrics, e.g. biometric systems in smartphones, has contributed to higher adoption of the technology as a viable authentication factor in modern systems.

In this work, we propose an approach for authenticating transactions in an online bank by using a combination of Bloom filters and error correcting codes. Firstly, protected biometric templates, using Bloom filters, are generated from faces detected in images captured using smartphones. Secondly, a key, shared between a smartphone and a bank server, is encoded using error correcting codes. The encoded key is then secured in the smartphone using the protected biometric templates. Authentication of a banking transaction is realised by unlocking the secured key with a protected biometric template that is close to the template used to lock the key. Experiments are performed on a database consisting of images and videos captured using an iPhone 6S.

Keywords: Transaction Authentication, Smartphone, Bloom filters, Error-correcting codes, Banking

1 Introduction

Secure and reliable identification and verification of customers and users is of paramount interest in applications such as online banking and services provided by government agencies. There are several ways in which biometrics can be applied in online banking. It can be used in the on-boarding process, where a customer opens a bank account, as part of the know your customer (KYC) regulations banks and other businesses are beholden to. In this scenario biometrics requires a deeper integration into the bank systems. The second way in which biometrics can be used as an authentication factor on a similar level as the traditional methods such as passwords, personal identification numbers (PIN), one time passwords(OTP) and hardware tokens. While adopting biometrics one must consider recent regulations with the recent European Union Directive the Revised Payments Services Directive (PSD2) [EP16a] and the General Data Protection Regulation (GDPR) [EP16b] that strongly imposes the privacy preserving with biometrics for finance and payment sector.

In this work, we propose a novel scheme for privacy preserving biometrics for authenticating transactions in an online bank by using a combination of Bloom filters and error-correcting codes. The proposed method is evaluated using the smartphone database comprised on 50 subjects collected using iPhone 6S that has indicated the reliability of the proposed method.

[1] Department of Information Security and Communication Technology, Norwegian University of Science and Technology, Gjøvik, Norway, {martin.stokkenes,raghavendra.ramachandra,christoph.busch}@ntnu.no

The rest of the paper is organized in the following manner: Section 2 describes the motivation and background for this work. Section 3 gives a detailed description of the proposed approach. Section 4 presents the experiments and results. Section 5 discusses future work and 6 gives concluding remarks.

2 Background and Motivation

In [HB10], Hartung et al. proposed a protocol for authentication of transactions in an online banking scenario using biometrics as authentication factor which is called Biometric Transaction Authentication Protocol (BTAP).

The protocol contains four entities; the banking server, the banking software (where the customer accesses the online bank), a biometric transaction device (which can capture the user's biometric characteristics), and lastly the user. A high level overview of the protocol is presented in Figure 1. The order of the steps in the protocol is indicated by the numbers.

When a user registers in the system, the banking server generates a key which is shared securely between the banking server and the biometric transaction device (BTD). A hash of the key is stored on the server, and on the biometric transaction device the key is secured by concealing it with the users biometric data. The key can only be uncovered when the user presents their biometric data during an authentication attempt.

Once a transaction is initiated by a user in the banking software, the transaction details (e.g. transaction amount, sender account number, receiver account number) is transmitted to the banking server, and the BTD which is in the possession of the user. On the BTD the user first verifies the transaction details, then provides a biometric sample for authentication. The biometric sample is used to uncover the key, and a hash of the key is used as key in a message authentication code (MAC) where the input message is the transaction details received earlier. The MAC is then transmitted to the banking server for verification. The banking server performs the same steps using the stored hash as key in the MAC. If the key was uncovered correctly on the BTD the MAC from the BTD will match with the MAC generated on the banking server and the transaction is successfully authenticated.

For more details on the protocol recounted here, please refer to the works by Hartung et al. [HB10]. The motivation for this work comes from the desire to evaluate if such a protocol is feasible by using a smartphone as the biometric transaction device. The main challenge

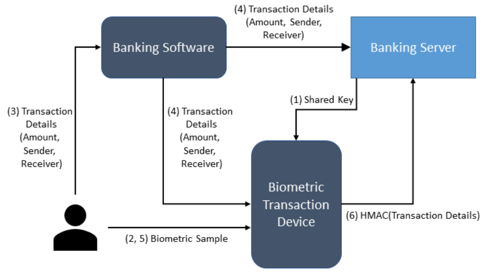

(4) Transaction Details (Amount, Sender, Receiver)
Banking Software
Banking Server
(1) Shared Key
(3) Transaction Details (Amount, Sender, Receiver)
(4) Transaction Details (Amount, Sender, Receiver)
Biometric Transaction Device
(6) HMAC(Transaction Details)
(2, 5) Biometric Sample

Fig. 1: Overview of the BTAP protocol

in realizing the protocol lies in how to secure the biometric data on the BTD. The following section will describe the proposed approach for how this challenge is addressed.

3 Biometric System

This section focuses on the integration of a biometric system to secure the key on the BTD as described in the previous section. An overview of the proposed approach is shown in Figure 2. Enrollment is the event that occurs when a user registers in the system, end verification occurs when the user tries to authenticate a transaction. The biometric system is a face recognition system and uses images captured using the smartphone. The following sections describes the different elements involved in the process.

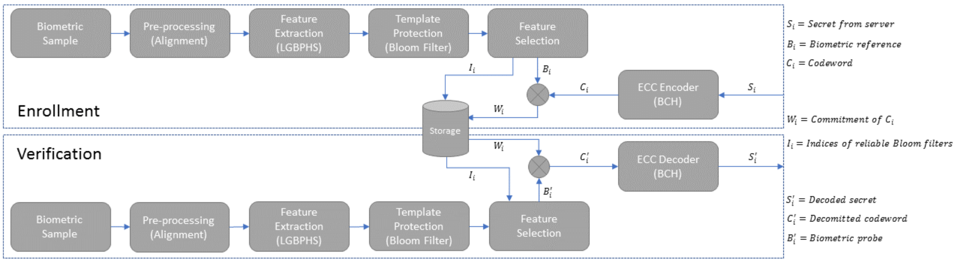

Biometric Sample
Pre-processing (Alignment)
Feature Extraction (LGBPHS)
Template Protection (Bloom Filter)
Feature Selection
I_i
B_i
C_i
ECC Encoder (BCH)
S_i
Enrollment
W_i
Storage
Verification
W_i
C_i'
ECC Decoder (BCH)
S_i'
I_i
B_i'
Biometric Sample
Pre-processing (Alignment)
Feature Extraction (LGBPHS)
Template Protection (Bloom Filter)
Feature Selection
S_i = Secret from server
B_i = Biometric reference
C_i = Codeword
W_i = Commitment of C_i
I_i = Indices of reliable Bloom filters
S_i' = Decoded secret
C_i' = Decomitted codeword
B_i' = Biometric probe

Fig. 2: Block diagram of the proposed approach

3.1 Pre-processing, Feature Extraction and Bloom filter Template Protection

Pre-processing: After an image is captured using the built-in camera in the smartphone it is processed to detect faces and normalized. Faces are detected using the face detection algorithm present in dlib [Ki09]. This gives the bounding box for the location of the face in the image. Facial landmark detection, using dlib pre-trained model, is then used to obtain points describing the locations of the eyes. The eye locations are used to identify any rotation in the face by finding the angle of a line between the eye center positions. The face is then rotated according to this angle.

Feature Extraction and Bloom filters: Facial features and protected templates are extracted from the images by adopting the methods used in [Go16, Go18] for face verification. This method is implemented in python [2], and uses the open source signal-processing and machine learning toolbox Bob [3] [An12]. For this work, the methods have been ported to work on the iOS platform to enable evaluation on an iPhone 6S device.

Extraction of facial features is based on a method called Local Gabor Binary Pattern Histogram Sequence (LGBPHS). A set of Gabor filters is applied to the normalized facial images, and produces several Gabor Magnitude Pictures (GMPs). The GMPs are transformed to Local Gabor Binary Pattern (LGBP) maps and divided into blocks. Histograms are computed from each of the blocks and concatenated to produce the feature vector for the facial image.

[2] `https://github.com/dasec/face-bf-btp`

[3] `https://www.idiap.ch/software/bob`

Facial features are protected by generating Bloom filters from the feature vector obtained in the previous step. Bloom filters is a probabilistic data structure, originally used to test whether an element exists in a set. As a template protection method Bloom filters are used as a one-way function to transform biometric features. They are obtained using the same parameters as described in [Go18] for the facial features. The final protected template consists of 2400 binary Bloom filters, each of length 16.

3.2 Securing the Key - Fuzzy Commitment

Error-correcting codes (ECC) is a technique from information and coding theory commonly used in digital communications to enable transmission of data over noisy channels. By adding redundancy to the message, an error correcting scheme is able to detect a number of errors and correct them. An ECC can be considered as a set of codewords and redundancy is added to the message by mapping it to one of the codewords contained in the set. ECCs have also been adopted for applications in biometric systems. In [JW99] Juels et al. proposed a method called the fuzzy commitment scheme (FCS) . In FCS a codeword is randomly selected from the set of available codewords in an ECC. The codeword is committed using a binary biometric template of equal length and in addition the codeword is concealed using a hash function. In an authentication scenario, the goal is to decommit the codeword using a biometric probe and generate a hash from the decomitted codeword which should be equal to the one stored during enrollment.

In this work the FCS is adapted to work with the transaction authentication protocol described in Section 2. During enrollment the key S_i is transmitted to the BTD. The user is asked present their biometric characteristic and from the feature extraction process we obtain the biometric template B_i. The ECC encoder is used to generate the codeword C_i from S_i. Now, C_i is committed by performing exclusive or (XOR) operation between C_i and B_i and we obtain W_i which is stored in the device: $W_i = C_i \oplus B_i$.

Once the user is requested to verify a transaction, a new sample is acquired of their biometric characteristic and the biometric template $B_i^{'}$ is obtained. This time, the XOR operation is performed between W_i and $B_i^{'}$ which gives the codeword $C_i^{'}$: $C_i^{'} = W_i \oplus B_i^{'}$.

If the hamming distance between the enrolled biometric template and the probe biometric template is less than or equal to the correction threshold t it will be possible to obtain the original codeword and reveal the key: $||B\_B_i^{'}|| <= t$.

If $C^{'}$ is decoded correctly, we retrieve the key S_i and it can be used to generate a MAC on the biometric transaction device that will match the with the MAC generated on the banking server and thus verify the transaction.

The Bose–Chaudhuri–Hocquenghem (BCH) codes are adopted for the error correction method in the fuzzy commitment scheme used to secure the key. A BCH code can be described by $BCH(n,m,t)$ where n is the length of the codeword, m is the length of the message to be encoded and t is the number of errors it is able to correct. In this case it is similar to a Hamming distance classifier as the result of performing XOR operation on two binary strings gives the number of disagreeing bits between the two strings, or the number of errors between them. BCH in this work is integrated on iOS based on implementation by Robert Morelos-Zaragoza[4].

3.3 Reliable Bloom Filter Selection

The feature vectors obtained from the feature extraction method contains more elements compared to a BCH code which is feasible to use. It is therefore necessary to reduce the feature space to obtain a vector with a suitable length. During enrollment a number of samples are captured from the user and protected templates based on Bloom filter is generated. The Bloom filters from each sample is compared against each other using Hamming Distance (HD). The average HD for each Bloom filter is computed and the positions or indices of the N_{bf} Bloom filters with lowest average HD is stored for the feature, and the BFs in those positions are used to create the final feature vector. N_{bf} is determined by dividing the size of the codeword with the size of each Bloom filter, see Eq 1. During authentication the indices stored during enrollment are used to select the N_{bf} number of Bloom filters required for the final template and one bit is discarded at the end of the vector to match the length of the codeword.

$$N_{bf} = \frac{|codeword| + 1}{|BloomFilter|} \tag{1}$$

4 Experiments and Results

4.1 Dataset and Experimental Setup

The proposed method was developed on a dataset consisting of face images from 43 subjects captured using a Samsung Galaxy S5 device. For each subject there are five reference images and five probe images. The evaluation of the proposed approach was performed on a dataset consisting of 50 subjects. In this dataset, the face images are extracted from video recordings which were captured using an iPhones 6S. For each subject, there are two sessions which each contain two videos of five seconds length. The videos are recorded at 30fps with a resolution of 1280×720. The sessions were recorded at different times. In the evaluation protocol face images are extracted from frame number 30, 60, 90, 120 and 150. This results in five images from each video and in total: 10 images used as reference and 10 images used as probes. Table 1 shows a detailed overview of the dataset used in the experiments.

Dataset	Device	No. of Subjects	Number of Images	Enrolment	Probe
Development	Samsung Galaxy S5	43	10	5	5
Evaluation	iPhone 6S	50	20	10	10

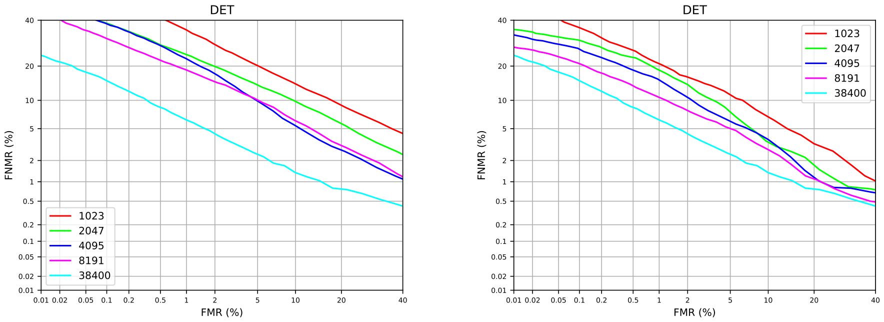
Tab. 1: Details of the datasets used during development and evaluation of the proposed approach

[4] `http://www.eccpage.com/bch3.c`

The performance of the biometric system is presented using the error rates false non-match rate (FNMR), false match rate (FMR) and equal error rate (EER). All experiments are evaluated using code running on the iPhone 6S.

4.2 Experiment 1: Impact of template and codeword size

In this experiment the evaluation concerns the biometric recognition performance as a function of the number of bits contained in the biometric templates. The feature extractor and template protection method generates a feature vector consisting of 2400 Bloom filters or $2400 \times 16 = 38400$ values. In order to reduce the number of bits from the original template the method described in section 3.3 is applied. Here it is compared against reduced feature vectors generated by randomly selecting the indices for which Bloom filters to use as the final feature vector. Figure 3 shows the DET curves from the two approaches. Figure 3a contains the graphs when random indices are selected and Figure 3b shows the DET curves when reliable Bloom filter indices are selected.



(a) DET curves showing results when random indices are selected during feature reduction

(b) DET curves showing results when reliable indices are selected during feature reduction

Fig. 3: Plots showing recognition performance

It is clearly illustrated from Figure 3 and Table 2 that selecting Bloom filters based on the Hamming distance produces less degradation in recognition performance, compared to random selection, when reducing the feature space from the original feature vector.

	Random		Reliable	
Vector Length	1-FNMR (%) @ FMR = (0.01)	EER (%)	1-FNMR (%) @ FMR (0.01)	EER (%)
1023	33.7	12.3	48.1	8.1
2047	47.9	9.8	64.4	6.1
4095	51.7	7.3	66.6	5.6
8191	55.6	7.6	72.3	5.0
	1-FNMR (%) @ FMR = (0.01)		EER (%)	
38400	75.9		3.2	

Tab. 2: Recognition performance for FMR fixed at 0.01% and EER for the different configurations of vector length

4.3 Experiment 2: Performance with BCH error correcting codes

Based on evaluation in the previous section, reducing the feature space to 8191 bits gives the least amount of degradation compared to the original template. However, the decoding time for the 8191 BCH code is an order of magnitude greater than that of 4095. Therefore we choose a trade-off between recognition performance and computational performance and analyse the impact of different triplets of $BCH(4095, m, t)$ By finding the lowest hamming distance present in the imposter distribution it is possible to choose a configuration of BCH which gives 0% FMR. In this case, the lowest HD in the imposter distribution is 418. The first BCH code with lower error correcting capability is BCH(4095, 676, 439). Table 4 gives the biometric performance in terms of $1 - FNMR$ and FMR for BCH configurations from $0\%FMR$ to $0.0086\%FMR$.

BCH code	1023	2047	4095	8191
Average Decoding Time	0,002s	0.011s	0.063s	0.353s

Tab. 3: Decoding time for different BCH codes on iPhone 6S

Codeword length (n)	Message Length (m)	Correction Capability (t)	1-FNMR	FMR
4095	802	415	59.1%	0%
4095	790	422	61.2%	0.0004%
4095	784	423	61.6%	0.0004%
4095	772	426	62.5%	0.0004%
4095	760	427	62.8%	0.0004%
4095	748	429	63.3%	0.0008%
4095	736	430	63.7%	0.0012%
4095	724	431	64.0%	0.0016%
4095	712	435	65.3%	0.0041%
4095	700	437	65.9%	0.0057%
4095	688	438	66.3%	0.0073%
4095	676	439	66.7%	0.0086%

Tab. 4: Biometric Recognition Performance for different values of m and t in BCH(4095, m, t) codes

By varying the error correction capability of the security of the biometric system can be changed according to given requirements. In the literature it is common to report performance for FMR of 0.01%. The result which is closest to that operating point, from Table 4, is the BCH(4095, 700, 437) where $1 - FNMR = 65.9\%$

5 Discussion and Future Work

By adopting a template protection scheme before applying the fuzzy commitment the security of the biometric data is increased. In case someone is able to obtain the codeword used in the XOR operation, the attacker will only obtain the protected template. For the fuzzy commitment scheme, the security is measured by the difficulty in obtaining the secret key from the commitment. A great amount of research has focused on how to measure the strength of biometric data in terms of entropy [Su13]. However, it is not always clear which approaches are best suited. In [Zh11] the authors propose to use the average min-entropy to estimate the security of the biometric data when applying it in a fuzzy commitment scheme. As future work, it is the goal to analyze the entropy of the biometric feature vectors generated in this work in order to estimate the security of the approach.

6 Conclusion

This work has studied the use of Bloom filter protected biometric templates and the fuzzy commitment scheme to secure a key which can be used in an authentication protocol to verify transactions in online banking. By adopting a template protection scheme on the biometric template before it is used in the fuzzy commitment, it achieves an extra level of security as an attacker would need to break both the fuzzy commitment and the Bloom filter template protection scheme. By selecting reliable Bloom filters the feature space can be reduced without a great loss in recognition performance. However, there is still work required to achieve a suitable performance for practical use.

Acknowledgments This work was carried out under the funding from the Research Council of Norway (Grant No. IKTPLUSS 248030/O70).

References

[An12] Anjos, A.; Shafey, L. El; Wallace, R.; Günther, M.; McCool, C.; Marcel, S.: Bob: a free signal processing and machine learning toolbox for researchers. In: 20th ACM Conference on Multimedia Systems (ACMMM), Nara, Japan. October 2012.

[EP16a] European Parliament, Council of the European Union: , Directive 2015/2366/EUof the European Parliament and of the Council of 25 November 2015 on payment services in the internal market, amending Directives 2002/65/EC, 2009/110/EC and 2013/36/EU and Regulation (EU) No 1093/2010, and repealing Directive 2007/64/EC, 2016.

[EP16b] European Parliament, Council of the European Union: , Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC, 2016.

[Go16] Gomez-Barrero, Marta; Rathgeb, Christian; Galbally, Javier; Busch, Christoph; Fierrez, Julian: Unlinkable and irreversible biometric template protection based on bloom filters. Information Sciences, 370-371:18–32, nov 2016.

[Go18] Gomez-Barrero, Marta; Rathgeb, Christian; Li, Guoqiang; Ramachandra, Raghavendra; Galbally, Javier; Busch, Christoph: Multi-biometric template protection based on bloom filters. Information Fusion, 42:37–50, jul 2018.

[HB10] Hartung, D.; Busch, C.: Biometric Transaction Authentication Protocol. In: 2010 Fourth International Conference on Emerging Security Information, Systems and Technologies. pp. 207–215, July 2010.

[JW99] Juels, Ari; Wattenberg, Martin: A fuzzy commitment scheme. In: Proceedings of the 6th ACM conference on Computer and communications security - CCS ’99. ACM Press, New York, New York, USA, pp. 28–36, 1999.

[Ki09] King, Davis E.: Dlib-ml: A Machine Learning Toolkit. Journal of Machine Learning Research, 10:1755–1758, 2009.

[Su13] Sutcu, Y.; Tabassi, E.; Sencar, H. T.; Memon, N.: What is biometric information and how to measure it? In: 2013 IEEE International Conference on Technologies for Homeland Security (HST). pp. 67–72, Nov 2013.

[Zh11] Zhou, X.; Kuijper, A.; Veldhuis, R.; Busch, C.: Quantifying privacy and security of biometric fuzzy commitment. In: 2011 International Joint Conference on Biometrics (IJCB). pp. 1–8, Oct 2011.