Mobile Banking Authentication Based on Cryptographically Secured Iris Biometrics

Nemanja Maček[1], Saša Adamović[2], Milan Milosavljević[3], Miloš Jovanović[4], Milan Gnjatović[5], Branimir Trenkić[6]

[1,6] School of Electrical and Computer Engineering of Applied Studies, 283 Vojvode Stepe st., Beograd, Serbia, e-mails: {nmacek, btrenkic}@viser.edu.rs

[2,3,4] Singidunum University, 32 Danijelova st., Beograd, Serbia, e-mails: {sadamovic, mmilosavljevic}@singidunum.ac.rs, milos.jovanovic.10@singimail.rs

[5] Faculty of Technical Sciences, University of Novi Sad, 6 Trg Dositeja Obradovića st., Novi Sad, Serbia, e-mail: milangnjatovic@uns.ac.rs

Abstract: This paper[1] presents an approach to designing secure modular authentication framework based on iris biometrics and its' implementation into mobile banking scenario. The system consists of multiple clients and an authentication server. Client, a smartphone with accompanying application, is used to capture biometrics, manage auxiliary data and create and store encrypted cancelable templates. Bank's authentication server manages encryption keys and provides the template verification service. Proposed system keeps biometric templates encrypted or at least cancelable during all stages of storage, transmission and verification. As templates are stored on clients in encrypted form and decryption keys reside on bank's authentication server, original plaintext templates are unavailable to an adversary if the phone gets lost or stolen. The system employs public key cryptography and pseudorandom number generator on small-sized templates, thus not suffering from severe computational costs like systems that employ homomorphic encryption. System is also general, as it does do not depend on specific cryptographic algorithms. Having in mind that modern smartphones have iris scanners or at least high-quality front cameras, and that no severe computational drawbacks exist, one may conclude that the proposed authentication framework is highly applicable in mobile banking authentication.

Keywords: mobile banking; authentication; biometrics; iris; cryptography

[1] This paper is an altered version of [19] presented at the 9th International Conference on Business Information Security (BISEC), in Belgrade, Serbia, 2017. The modifications include the following: application of the proposed framework in mobile banking, detailed description of iris feature extraction, experimental evaluation with highly-realistic dataset and more detailed security evaluation of the system.

1 Introduction

Mobile banking is a service provided to customers by a financial institution that allows financial transactions to be conducted using a mobile device (a smartphone or a tablet) and accompanying software, usually provided by the same institution. Having said that , one may conclude that mobile banking is one of the most security-sensitive tasks performed by a typical smartphone user [1]. Although many financial institutions offer their mobile banking services "with peace of mind" [2], there is not a bulletproof solution providing users with 100% security guarantee. There are several security aspects regarding financial transactions conducted via mobile devices that should be addressed: physical security of the device, security of the application running, authentication of the user and the device to the service provider (bank's authentication server), encryption of data being transmitted and data that will be stored on device for later analysis by the customer. This paper addresses the authentication of the user and the device to the service provider. Variety of authentication methods are implemented in mobile banking today, all having their upsides and downsides. As an example, passwords are the easiest method to implement, but customers that employ passwords to mobile banking authentication are at risk of fraud or theft. Major companies have identified the need for strong security countermeasures and they are producing new hand-held devices with built-in biometric scanners. According to Gartner, over 30% of mobile devices are currently using biometrics, and banks should see that as an opportunity to secure their customers and transactions rather than a barrier to adoption [3].

Biometric authentication is the process of establishing user identity based on physiological or behavioral qualities of the person [4, 5]. Biometrics may be addressed as an ultimate authentication solution: users do not need to remember passwords or carry tokens and biometric traits are distinctive and non-revocable in nature [6], thus offering non-repudiation [7]. Like any personal information, biometric templates can be intercepted, stolen, replayed or altered if unsecured biometric device is connected to a network or if a skilled adversary gains physical access to a device which does not employ anti-forensic techniques that would prevent extraction of sensitive data (i.e. unprotected templates). Brief surveys of attacks on biometric authentication systems are given in [8, 9]. Due to non-revocability of biometric data aforementioned attacks may lead to identity theft. Having said that, it becomes clear that biometric systems operate with sensitive personal information and that template security and privacy are important issues one should address while designing authentication systems. To counterfeit identity theft, one should not rely on post-mortem misuse identification [10] – it should be prevented with technological countermeasures that provide strong template security and user's privacy protection. Additionally, the performance of the biometric system should be downgraded to the reasonable level after introducing these countermeasures to the system, i.e. they are expected not to degrade the verification accuracy to unacceptable level or introduce severe computational costs or storage requirements.

2 Approaches to Biometric Template Protection

One approach to biometric template security and privacy is cancelable biometrics. Two main categories of cancelable biometrics can be distinguished: intentional distortion of biometric features with non-invertible transforms [11], such as block permutation of iris texture, and biometric salting. Cancelable biometrics that employs non-invertible transforms is based on application of the same transformation to a given biometric sample during enrollment and verification. There are a large number of non-invertible transforms for variety of biometric modalities, and some of them operate with the key. Having said that, each compromised template can easily be revoked and another transformation can be applied during re-enrollment; if transformation operates with the key, only the key is changed. Examples of cancelable transforms applicable to fingerprint and iris are given in [12] and [13], respectively. However, non-invertible transforms may be partially reversible and they usually degrade overall verification accuracy, thus they are not a fail-safe solution to a biometric template protection problem. Biometric salting refers to transformations of biometric templates that are selected to be invertible, where any transformation is considered to be an approach to biometric salting even if templates have been extracted in a way that it is not feasible to reconstruct the original biometric signal [14]. Although biometric salting does not degrade the verification accuracy, non-invertible transfors provide higher level of security. Hence, biometric salting is not a fail-safe solution to the problem either.

Another approach to providing biometric template security and privacy is the application of homomorphic encryption schemes [15, 16]. Homomorphic encryption refers to cryptographic algorithms that allow some computations to be performed in the encrypted domain. Research on homomorphic encryption algorithms that support both addition and multiplication based on lattice encryption was expected to provide novelties in biometric template security [17], but no results were reported in relevant literature. Although applicable in theory (e.g. homomorphic encryption appears to be suitable for application in systems that employ bitwise XOR to calculate Hamming distance during verification between two binary iris templates), there are two reasons why it is not practical: the encrypted template is large and the system is computationally expensive. Reader may consult [16] for more details.

The main contribution of this paper is a secure modular authentication system based on iris biometrics applicable to mobile banking. An approach presented in this paper employs public key cryptography, pseudorandom number generators and cancelable biometrics. Non-invertible transformation operates with the key stored on a device's trusted storage. The system does not suffer from the drawbacks of homomorphic encryption as cryptographic operations are not computationally expensive and no large templates are created. Biometric templates are encrypted or at least cancelable during all stages of operation (excluding feature extraction) resulting in a system prone to a variety of attacks. Having in mind that the system satisfies requirements set to a cryptographically secured biometric system that provides strong privacy protection listed in [10], and that devices with iris scanners are emerging technology, we can conclude that this modular system is suitable for implementation in mobile banking.

3 Counterfeiting Attacks with Modular Architecture

Biometric authentication systems consist of four modules: sensor, feature extractor, matcher and template database. If these modules reside on one device, authentication system is vulnerable to variety of attacks [18]. These include fake biometrics, replay attack, attack on the feature extraction module, attack on the channel between feature extractor and matcher, compromising the database, attack on the communication channel between template database and the matcher and overriding the result declared by the matcher module. Some of these attacks are easy to execute if the system is not properly designed. For example, if the system does not employ liveness detector, it is easy to perform sensor attack with fake biometrics. To prevent execution of aforementioned attacks, entire system is split into two high-level modules, residing on two devices. Additionally, both cancelable biometrics and strong cryptographic protection are introduced to the system. Modular system now consists of multiple clients (devices used to capture biometrics, manage auxiliary data and create encrypted cancelable templates) and an authentication server (device that manages encryption keys and verifies cancelable templates). As proposed system deals with the iris biometrics, which employs XOR operation to verify a person, a cancelable transform that partially reassembles one-time-pad cypher (the key is employed more than once, but is of the same length as plaintext) is used.

Aside from cryptographic security, system is expected to provide strong privacy protection, resulting in following set of requirements: (1) biometric templates remain encrypted or at least cancelable during all stages of storage, transmission and verification (e.g. authentication server should never obtain plaintext templates,) and (2) no client is allowed to access private keys stored on authentication server as it may compromise the security of the stored templates. Further, system should be resilient to a template substitution and all low level attacks, it should not suffer from severe computational drawbacks and cryptographic countermeasures should not degrade overall accuracy (i.e. they should not introduce additional false acceptance or false rejection rates to the system).

4 Proposed Modular Authentication Framework

A framework for modular authentication systems based on conventional XOR biometrics, such as iris, is presented in this section [19]. Conventional XOR biometrics is based on Hamming distance calculation between templates obtained during enrollment and verification phases. Hamming distance is chosen as verification metrics as it is suitable for application of one-time-pad partially based non-invertible transforms of the template, i.e. simple XOR operation with the non-invertible transform key of the same length as the original template. This method of biometric template protection guards the end user from identity theft and allows the user to easily re-enroll with another key, if any suspiction about the key being compromised occurs.

During the enrollment phase, the user provides numeric user ID and non-invertible transform key K_t to the client. Let H(x) denote the hash function (one may select solution-specific), ID the identity of the user and K_{priv}, K_{pub} the private and the public key, respectively. Hash of the user ID is calculated on the client and sent to the authentication server. Authentication server generates a keypair (K_{priv}, K_{pub}), stores the private key with hash of user ID (H(id), K_{priv}) and sends public key to the client. Client obtains biometrics, creates a binary template b_0, and generates cancelable binary template $b = K_t \oplus b_0$. Client generates random seed s_0 and encrypts it with the public key: $s_E = \mathrm{E}(s_0, K_{pub})$, where E denotes the encryption operation. Any public-key encryption algorithm that suffice the principles behind the information theory and strong cryptography can be used. Client generates a keystream s = PRNG (s_0) using pseudorandom number generator and given seed, where PRNG denotes applicable pseudorandom number generator. Client calculates $s \oplus b$, stores (H(id), s_E, $s \oplus b$) and discards the rest of the data.

During the verification phase, the user provides numeric user ID and non-invertible transform key K_t to the client. Client obtains biometrics, creates a template b_0' and generates cancelable binary template $b' = K_t \oplus b_0'$. Client calculates user ID hash and retrieves values s_E and ($s \oplus b$) from stored record (H(id), s_E, $s \oplus b$) with the corresponding user ID hash. Client calculates $s \oplus b \oplus b'$ and sends it with the encrypted seed s_E to the authentication server. Hash of the user ID calculated on the client is sent to the authentication server. Authentication server retrieves private key from stored record (H(id), K_{priv}) with the corresponding user ID hash. Let D denote the decryption operation. Authentication server decrypts the seed by doing $s_0 = \mathrm{D}(s_E , K_{priv})$ and generates the keystream: s' = PRNG (s_0). As pseudorandom number generator is deterministic and same seed is used to generate keystreams both in enrollment and verification phases, generated keystreams s and s' will be identical, i.e. $s = s'$. Aside from this, the same non-invertible transform key K_t is used is both phases. Thus, server calculates $s \oplus b \oplus s' \oplus b' = b \oplus b' = K_t \oplus b_0 \oplus K_t \oplus b_0' = b_0 \oplus b_0'$ and compares the Hamming distance between templates b_0 and b_0' with the threshold. According to that result, the decision is made (user is genuine or imposter) and sent back to the client. One should note that, although the result of comparison is the Hamming distance between original, unaltered templates obtained via feature extractor, server makes the calculation using cancelable templates generated with the non-invertable transform key.

4.1 Security Evaluation of the Proposed Framework

Security of the system may be summarized as follows. Templates are encrypted or at least cancelable during all stages of storage, transmission and verification, and the client is not allowed to access private keys stored on authentication server, which satisfies the conditions set for an ideal biometric system. System employs two factor authentication thus making an imposter with helper data virtually impossible to claim as genuine user. If templates stored on a client are somehow compromised, re-enrollment with another transform key and encryption key-pair will remediate the situation. Substitution attacks cannot be performed, as the public key is discarded at the end of enrollment. As an adversary cannot recreate the keystream s from the encrypted seed s_E and the public key, system is resilient to most of the attacks on the biometric encryption systems.

5 Implementation in the Mobile Banking Authentication Scenario

Authentication server resides in the bank. As authentication server stores encryption keys, it is logical that encrypted templates reside on the client. This prevents an attacker who obtains illegal access to authentication server to decrypt templates. The client is a mobile device (smartphone or a tablet) with an iris scanner. Additional software that provides feature extraction and cryptographic operations is installed on the client (as an additional application provided by the bank). Non-invertible transform key is stored on the device. User obtains this key from the bank as an output of true random number generator; the length of the key must be equal to the length of iris template as the XOR of original template and the key is performed straight after the feature extraction. User is allowed to wipe both the key and the data stored during enrollment phase both locally, if he suspects the data is somehow compromised, and remotely, if the device gets stolen. The bank is allowed to do remote data wiping also, if the authentication server is somehow compromised.

During the enrollment phase the system operates as depicted in Figure 1.

- Client-side application calculates hash of the devices' IMEI and sends it to the authentication server. Devices' IMEI is hashed to protect user's privacy – hashing prevents plaintext transmission between the client and the server as well as the storage of user-sensitive plaintext data on the server-side.
- Server generates a private-public keypair (K_{priv}, K_{pub}), stores the private key with hash of IMEI (H(IMEI), K_{priv}) and sends public key to the mobile device.
- User provides iris biometrics to the mobile device. Client-side application creates a binary iris template b_0 (as explained in section 5.1 of this paper) and generates cancelable binary template $b = K_t \oplus b_0$ using non-invertible transformation key stored on the device. Client-side application further generates random seed s_0 and encrypts it with the public key: $s_E = \mathrm{E}(s_0, K_{pub})$. Application generates a keystream $s = \mathrm{PRNG}\ (s_0)$ using pseudorandom number generator and given seed, calculates $s \oplus b$, stores values (s_E, $s \oplus b$) on the device and discards the rest of the data.

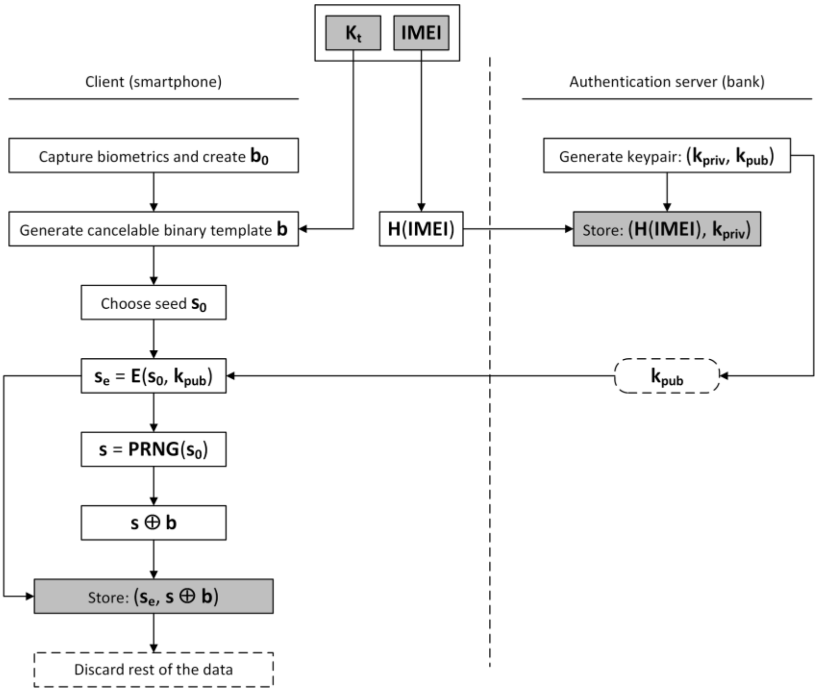

Figure 1
Enrollment phase

During the verification phase the system operates as depicted in Figure 2.

- Hash of the device IMEI is calculated on the client-side application and sent to the authentication server.
- User provides biometrics to the mobile device. Client-side application creates binary iris template b_0' and generates cancelable binary template $b' = K_t \oplus b_0'$. Application retrieves values s_E and $(s \oplus b)$, calculates $s \oplus b \oplus b'$ and sends it with the encrypted seed s_E and hash of the devices' IMEI to the authentication server.
- Server retrieves private key from stored record (H(IMEI), K_{priv}) with the corresponding device IMEI hash, decrypts the seed with the private key by doing $s_0 = \mathrm{D}(s_E , K_{priv})$ and generates the keystream: $s' = \mathrm{PRNG}\ (s_0)$. As stated in section 4, due to deterministic nature of PRNGs, same seeds will produce identical keystreams s and s', and the same key K_t is used during enrollment and verification. Authentication server further calculates $s \oplus b \oplus s' \oplus b' = b \oplus b' = K_t \oplus b_0 \oplus K_t \oplus b_0' = b_0 \oplus b_0'$. As with the framework, the result of comparison is the Hamming distance between original, unaltered templates, but the server makes the calculation using cancelable templates (thus having no access to original ones, nor to the non-invertible transform key).

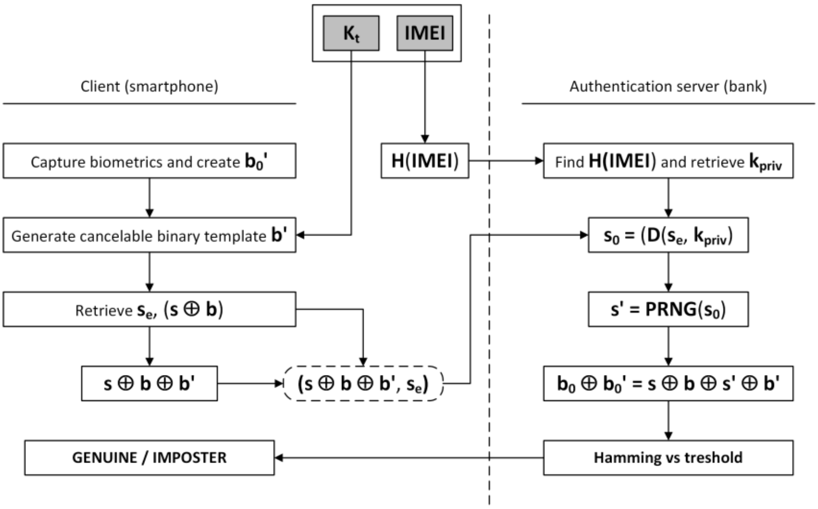
K_t
IMEI
Client (smartphone)
Authentication server (bank)
Capture biometrics and create b0'
H(IMEI)
Find H(IMEI) and retrieve kpriv
Generate cancelable binary template b'
s0 = (D(se, kpriv)
Retrieve se, (s ⊕ b)
s' = PRNG(s0)
s ⊕ b ⊕ b'
(s ⊕ b ⊕ b', se)
b0 ⊕ b0' = s ⊕ b ⊕ s' ⊕ b'
GENUINE / IMPOSTER
Hamming vs treshold

Figure 2
Verification phase

Pseudocodes for enrollment and verification phases are listed below.

User enrollment algorithm

```
INPUT: $b$ – plaintext biometric template, $K_t$ – transform key
OUTPUT: $s \oplus b$ – encrypted biometric template, $s_E$ – encrypted seed
Client:
1.    send (H(IMEI))
2.    $b = K_t \oplus b_0$
3.    random ($s_0$); $s$ = PRNG ($s_0$)
4.    get ($K_{pub}$); $s_E$ = E($s_0$, $K_{pub}$)
5.    store ($s_E$, $s \oplus b$)
Server:
1.    get (H(IMEI))
2.    generate ($K_{priv}$, $K_{pub}$)
3.    send ($K_{pub}$); store (H(IMEI), $K_{priv}$)
```

User verification algorithm

```
INPUT: $b'$ – plaintext biometric template, $K_t$ – transform key, $t$ - threshold
OUTPUT: decision
Client:
1.    send (H(IMEI))
2.    $b' = K_t \oplus b_0'$
3.    send ($s \oplus b \oplus b'$, $s_E$)
4.    get (decision)
Server:
1.    get (H(IMEI), $s \oplus b \oplus b'$, $s_E$)
2.    $s_0$ = D($s_E$ , $K_{priv}$);  $s'$ = PRNG ($s_0$)
3.    $b_0 \oplus b_0' = s \oplus b \oplus s' \oplus b'$
4.    if ($b_0 \oplus b_0'$) < $t$ then decision = “genuine” else decision = “imposter”
5.    send (decision)
```

5.2 Generating Binary Iris Templates

For more details on iris feature extraction methods reader may consult [20]. Before the template is generated from extracted features, acquired iris image must be pre-processed. Outer radius of iris patterns and pupils are first localized with Hough transform that involves a canny edge detector to generate an edge map. Poorly localized iris will result in unsuccessful segmentation and poor reproducibility of the template, which further results in high false rejection rates. Hough transform identifies positions of circles and ellipses [21] – it locates contours in an *n*-dimensional space by examining whether they lie on curves of a specified shape. Hough transform for outer iris and pupil boundaries and a set of *n* recovered edge points (x_i, y$_i$) is defined by:

$$H\left(x_c, y_c, r\right) = \sum_{i=1}^{n} h\left(x_i, y_i, x_c, y_c, \mathrm{r}\right), \tag{1}$$

$$h\left(x_i, y_i, x_c, y_c, r\right) = \begin{cases} 1, \left(x_i - x_c\right)^2 + \left(y_i - y_c\right)^2 - r^2 = 0 \\ 0, \left(x_i - x_c\right)^2 + \left(y_i - y_c\right)^2 - r^2 \neq 0 \end{cases}. \tag{2}$$

The circle (x_c, y_c, r) through each edge point (x_i, y_i) is defined as:

$$\left(x_i - x_c\right)^2 + \left(y_i - y_c\right)^2 = r^2 . \tag{3}$$

The triplet that maximizes H (x_c, y_c, r) is common to the greatest number of edge points and is a reasonable choice to represent the contour of interest [22]. Once an iris image is localized, regions of interests are defined and it is transformed into fixed-size rectangular image. The normalization process employs Daugman's rubber sheet model that remaps the iris image $I(x, y)$ from Cartesian to polar coordinates [20]:

$$I\left(x\left(r,\theta\right), y\left(r,\theta\right)\right) \rightarrow I(r,\theta) . \tag{4}$$

Parameter r is on the interval [0, 1] and θ is the angle [0, 2π]. If iris and pupil boundary points along θ are denoted as (x_i, y_i) and (x_p, y_p), respectively, the transformation is performed according to:

$$x\left(r,\theta\right) = \left(1-r\right) x_p\left(\theta\right) + r x_i\left(\theta\right), \tag{5}$$

$$y\left(r,\theta\right) = \left(1-r\right) y_p\left(\theta\right) + r y_i\left(\theta\right). \tag{6}$$

The rubber sheet model does not compensate rotational inconsistencies, but it takes into account pupil dilation size inconsistencies in order to produce a normalized representation with constant dimensions [23] set by angular and radial resolution. Angular resolution is set by number of radial lines generated around the iris region, while radial resolution refers to the number of data points in the radial direction.

Although various extraction methods are reported in the literature, discriminant features are extracted from a normalized iris using conventional method based on Gabor filtering. This method is validated as suitable feature extraction method in various researches presented by other authors. Normalized image is broken into a number of 1-D signals that are convolved with 1-D Gabor wavelets. The frequency response of 1-D log-Gabor filter [24] is given by:

$$G(f) = \exp\left(-\left(\log \frac{f}{f_0} \right)^2 \Big/ 2\left(\log \frac{\sigma}{f_0} \right)^2 \right), \tag{7}$$

where f_0 denotes center frequency, and σ the bandwidth of the filter. Phase quantization is applied to four levels on filtering outputs (each filter produces two bits of data for each phasor) and the quantized phase data is used to encode an iris pattern into a bit-wise biometric template. The number of bits in the biometric template depends on angular and radial resolution and the number of used filters, while the template entropy depends on the number of used filters, their center frequencies and the parameters of the modulating Gaussian.

5.3 Performance Evaluation of the Proposed System

Performance of the proposed system depends on various factors, such as the quality of the camera and illumination, as well as the parameters of employed feature extraction algorithms. It is very important to state that in our mobile banking scenario the accuracy of the system does not depend on the cryptographic protection and cancelable biometrics – they introduce no additional false acceptance or false rejection rates. Majority of the experiments on iris verification reported in the literature employ CASIA-Iris database, collected by the Chinese Academy of Sciences' Institute of Automation [25]. However, in order to get the realistic picture on how iris verification works with smartphones, a custom dataset is created using Huawei P10 Lite front camera. Images were subsequently processed in MATLAB (version R2016a). The iris image dataset used in our experiments consists of 210 gray-scale samples from 10 subjects obtained outdoors and indoors with different illumination. Each iris image is normalized into an 8-bit 240x20 pixel image, and a 1-D log-Gabor filter with σ=0.5 and 12 pixel center wavelength is subsequently applied, resulting in a 9600 bit template. These parameters were found to provide high local entropy and optimum encoding [26, 27]. One randomly selected outdoor image for each subject is used to enroll the user and all images are used to verify them. Results of experimental evaluation are given in Table 1.

Table 1
Experimental evaluation on realistic dataset (iris images captured by smartphone's front camera)

Scene	FAR	FRR
Low threshold (reducing FAR)		
Outdoors (daylight)	0 %	2 %
Indoors (normal illumination)	0 %	2 %
Indoors (medium illumination)	0 %	4 %
Indoors (poor illumination)	0 %	18 %
High threshold (reducing FRR)		
Outdoors (daylight)	0 %	0 %
Indoors (normal illumination)	0 %	0 %
Indoors (medium illumination)	2 %	2 %
Indoors (poor illumination)	6 %	4 %

Although verification with low threshold values fails indoors with poor illumination, this is not something we consider to be the drawback, as user is allowed to retry. The real problem occurs if the threshold is high, as user may still be verified as genuine, even if larger number of bits differ between two templates. This results in occurrence of false acceptance with medium illumination (less than 450 lumens, approximately one 9-11 watts compact fluorescent lamp illuminating 25 square meters sized room), or poor illumination (less than 200 lumens, i.e. one 3-5 watts compact fluorescent lamp illuminating the room of the same size). In other words, if the treshold is to high and the illumination is inappropriate, system enters the danger zone and is no more applicable to the mobile banking due to occurrence of false acceptance rates. Outside of that zone, it operates stable and may only require additional authentication attempt(s). Having said that, it is necessary to keep the verification threshold as low as possible to avoid false acceptance.

The concrete threshold depends on the camera used to capture iris image, and it should be set on the client-side application automaticaly (if the pre-calculated optimal threshold for the concrete device exists in records on devices previously used for that purpose) or by bank's authorized officer, during the first enrollment (if pre-calculated data does not exist for the concrete model). The later one should employ several captures of user's iris, a set of irises belonging to different persons and decidability as the metric, which takes into account the mean and standard deviation of the intra-class and inter-class distributions. The overall decidability of iris recognition is revealed by comparing Hamming distance distributions for same versus for different irises [20]. Users should not be allowed to set this value by themselves.

Another issue of iris verification system is the presence of contact lenses. Contact lenses, particularly textured ones, obfuscate the natural iris patterns, thus presenting a challenge to the iris verification. Effects of contact lenses on iris verification systems were analyzed in [28, 29]. Yadav et al. [29] presented lens detection algorithm that can be used to reduce the effect of contact lenses, stating that their approach outperforms other lens detection algorithms and provides improved iris recognition performance.

5.4 Security Evaluation of the Proposed System

Regarding security of the proposed mobile banking authentication solution, same conclusions can be made as with the framework it is built upon. Templates are encrypted or at least cancelable during all stages of operation, and the mobile device is not allowed to access private keys stored on authentication server. Authentication server has no access to the transform keys and cancelable templates created on the mobile device during enrollment. If the phone is stolen, an adversary cannot claim as legitimate user as the system is prone to all attacks listed in [18] as well as to hill-climbing [30], non-randomness [31], re-usability [32], blended substitution [33] and linkage attack [34].

Although some key-exchange protocols may be introduced to the system, the most secure way to distribute non-invertible transform key is to make the user obtain it directly from the bank, as it eliminates chances of identity spoof (which may cause further social engineering attacks). Both the user and the bank are allowed to remotely wipe all stored data (including the key) if the phone gets lost or stolen. If the device is somehow returned to the owner, he or she may retrieve new non-invertible key from the bank and undergo re-enrollment procedure. During the re-enrollment, user will provide new biometric sample to the device, client-side application will generate new seed for pseudorandom number generator, and bank’s authentication server will generate new private-public keypair that will further be used to encrypt and decrypt the seed.

One should note that physical access to the device does not allow an adversary to retrieve the non-invertible transformation key. Latest smartphone models shipped with biometric sensors that operate with several modalities (e.g. fingerprint, face and iris) running an Android operating system (version 6 or higher) include countermeasures that prevent physical acquisition of sensitive data, even with the state of the art forensic tools and devices. This fact originating from digital forensics provides us with sufficient level of security when certain amount of sensitive information is stored on the device, such as this non-invertible transform key.

The data being transmitted over the network (reassembling the Alice-Bob scenario used to explain cryptographic protocols) and stored on smartphones and the bank server is depicted in Figure 3. According to Figure 3, the following values are transmitted: user-specific public key (just one time, during enrollment), hash of the users’ IMEI (during enrollment and during each verification), and the XOR of enrolled and verification cancelable templates with the keystream (during each verification): $s \oplus b \oplus b'$. We could not identify any possible weakness that would allow an adversary to extract information from the transmitted data. One thing, however, is very important to state – the choice of poor pseudorandom number generator may lead to small leakage of information when $s \oplus b \oplus b'$ is transmitted from client to the server. Hence, one should use the cryptographically strong generator that is highly entropic in the information theory sense.

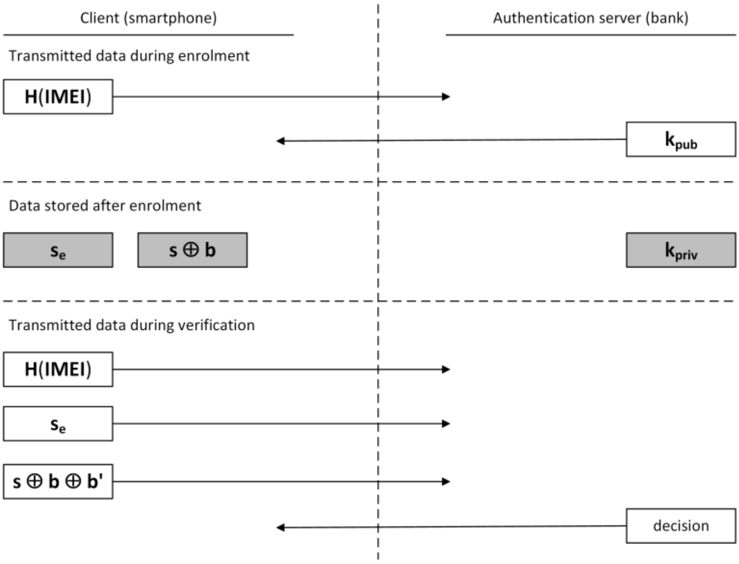
Client (smartphone)
Authentication server (bank)
Transmitted data during enrolment
H(IMEI)
k_{pub}
Data stored after enrolment
s_e
$s \oplus b$
k_{priv}
Transmitted data during verification
H(IMEI)
s_e
$s \oplus b \oplus b'$
decision

Figure 3
Data being transmitted and data being stored

Additionally, one should note that the security of the entire system depends also on the security of iris recognition subsystem. Although iris scanners should be hard to trick into false acceptance, a group of hackers managed to do so with the Galaxy S8 iris-based authentication; hardware required to complete the attack cost less than the smartphone itself [35]. If the phone does not employ fake iris countermeasures, similar scenarios may occur – for example, data extracted from selfies found on the stolen phone with performant cameras may be used to obtain fake iris images. Several approaches have been proposed to detect fake irises. An approach to iris contact lens detection based on deep image representations [36] uses a convolutional network to build a deep image representation and an additional fully connected single layer with softmax regression for classification. Sinha et al.'s iris liveness detection approach [37] employs Flash and motion detection of natural eye in order to detect the liveliness of real iris images before matching from stored templates, thus significantly increasing security and reliability of the system. A solution suitable for implementation in mobile devices was proposed by Gragnaniello, et al. [38]; a fast and accurate technique to detect printed-iris attacks is based on the local binary pattern descriptor (LBP). Their algorithm encompasses three steps: computation of the high-pass image residual, feature extraction based on a suitable LBP descriptor and classification with support vector machines with a linear kernel. According to authors, the detection performance is extremely promising, despite the very low complexity.

Conclusions

An implementation of modular authentication system based on iris biometrics into mobile banking scenario was presented in this paper. Strong cryptography that is not bound to a specific public key algorithm or pseudorandom number generator and bitwise XOR cancelable biometrics were introduced to the modular system in order to prevent execution of number of attacks on classical biometric and biometric encryption systems. Employed cryptographic countermeasures do not degrade the verification accuracy and do not introduce severe computational costs. According to security evaluation of the system, results of the experiments conducted with realistic dataset, and the fact that devices with iris scanners are emerging technology, we conclude that this modular architecture is highly applicable in mobile banking scenario. The only drawback of the proposed modular authentication framework is it’s limitation to biometric modalities that are verified by calculating Hamming distance. Although it is applicable to iris and, conditionally, fingerprint [33], we will focus our further research into developing authentication systems that can employ other biometric modalities, e.g. systems based on speaker recognition and face recognition. Additionally we will evaluate the application of fake iris detection approaches presented in [37] and [38] in our system in order to raise overall level of security, as well as the application of lens detection algorithm [29] to reduce the effect of contact lenses and increase verification accuracy if subjects wear lenses.

References

[1] M. Mannan, P. C. Van Oorschot: Security and Usability – The Gap in Real-World Online Banking. NSPW’07, North Conway, NH, USA, Sep. 18-21, 2007

[2] Y. S. Lee, N. H. Kim, H. Lim, H. Jo, H. J. Lee: Online banking authentication system using mobile-OTP with QR-code. In Proc. 5th International Conference on Computer Sciences and Convergence Information Technology (ICCIT), November 2010, pp. 644-648, IEEE

[3] C. Stamford: Gartner Says 30 Percent of Organizations Will Use Biometric Authentication for Mobile Devices by 2016. February 4, 2014, available online, last time visited April 2018

[4] A. K. Jain, A. Ross, S. Prabhakar: An Introduction to Biometric Recognition. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 14, pp. 4-20, 2004

[5] A. K. Jain, A. Ross: Introduction to Biometrics. In Handbook of Biometrics, A. Jain et al. (Eds), Springer, 2008

[6] Y. C. Feng, P. C. Yuen, A. K. Jain: A Hybrid Approach for Face Template Protection. In Proceedings of SPIE Conference of Biometric Technology for Human Identification, Orlando, USA, Vol. 6944, pp. 325, 2008

[7] P. Balakumar, R. Venkatesan: A Survey on Biometrics-based Cryptographic Key Generation Schemes. International Journal of Computer Science and Information Technology & Security, Vol. 2, No. 1, pp. 80-85, 2012

[8] A. K. Jain, K. Nandakumar, A. Nagar: Biometric Template Security. EURASIP J. Adv. Signal Process, 2008:1-17, 2008

[9] J. Galbally, C. McCool, J. Fierrez, S. Marcel, J. Ortega-Garcia: On the Vulnerability of Face Verification Systems to Hill-Climbing Attacks. Pattern Recognition, 43(3) pp. 1027-1038, 2010

[10] A. Stoianov: Cryptographically secure biometrics. In SPIE Defense, Security, and Sensing, International Society for Optics and Photonics, 2010

[11] N. Maček, B. Đorđević, J. Gavrilović, K. Lalović: An Approach to Robust Biometric Key Generation System Design. Acta Polytechnica Hungarica, Vol. 12, No. 8, pp. 43-60, 2015

[12] N. K. Ratha, S. Chikkerur, J. H. Connell, R. M. Bolle: Generating Cancelable Fingerprint Templates. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 29(4), pp. 561-572, 2007

[13] J. Zuo, N. K. Ratha, J. H. Connell: Cancelable iris biometric. In Pattern Recognition, ICPR 2008, 19th International Conference on (pp. 1-4), IEEE, 2008

[14] C. Rathgeb, A. Uhl: A survey on biometric cryptosystems and cancelable biometrics, EURASIP Journal on Information Security 2011, 2011:3, open access, no pagination

[15] J. Bringer, H. Chabanne: An authentication protocol with encrypted biometric data. In International Conference on Cryptology in Africa, pp. 109-124, Springer Berlin Heidelberg, 2008

[16] B. Schoenmakers, P. Tuyls: Computationally secure authentication with noisy data. Chapter 9 in P. Tuyls, B. Škorić, T. Kevenaar, eds., Security with Noisy Data: Private Biometrics, Secure Key Storage and Anti-Counterfeiting, Springer-Verlag, London, pp. 141-149, 2007

[17] C. Gentry: Fully Homomorphic Encryption Using Ideal Lattices. 41st ACM Symposium on Theory of Computing (STOC), pp. 169-178, 2009

[18] R. Jain, C. Kant: Attacks on Biometric Systems – An Overview. International Journal of Advances in Scientific Research, 1(07), pp. 283-288, 2015

[19] N. Maček, M. Milosavljević, I. Franc, M. Bogdanoski, M. Gnjatović, B.Trenkić. Secure Modular Authentication Systems Based on Conventional XOR Biometrics. In Proc. of the 9th Int. Conf. on Business Information Security (BISEC2017), Belgrade, October 18th, 2017, pp. 27-32

[20] J. Daugman: How iris recognition works. Circuits and Systems for Video Technology, IEEE Transactions on, 14(1) pp. 21-30, 2004

[21] D. J. Kerbyson, T. J. Atherton: Circle Detection using Hough Transform Filters. Fifth International Conference on Image Processing and its Applications, Edinburgh, UK, 04 – 06 July 1995, pp. 370-374

[22] R. P. Wildes: Iris Recognition – an Emerging Biometric Technology. Proceedings of the IEEE, 85(9) pp. 1348-1363, 1997

[23] G. Amoli, N. Thapliyal, N. Sethi: Iris Preprocessing. International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 2, No. 6, pp. 301-304, 2012

[24] D. J. Field: Relations between the Statistics of Natural Images and the Response Properties of Cortical Cells. Journal of the Optical Society of America, Vol. 4, No. 12, 1987

[25] Biometrics Ideal Test, http://biometrics.idealtest.org

[26] S. Adamović, M. Milosavljević: Information Analysis of Iris Biometrics for the Needs of Cryptology Key Extraction. Serbian Journal of Electrical Engineering, Vol. 10, No. 1, pp. 1-12, 2013

[27] S. Adamović, M. Milosavljević, M. Veinović, M. Šarac, A. Jevremović: Fuzzy commitment scheme for generation of cryptographic keys based on iris biometrics. IET Biometrics, Vol. 6, No. 2, pp. 89-96, 2017

[28] J. S. Doyle, K. W. Bowyer, P. J. Flynn: Variation in accuracy of textured contact lens detection based on sensor and lens pattern. In Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on, pp. 1-7, 2013, IEEE

[29] D. Yada, N. Kohli, J. S. Doyle, R. Singh, M. Vatsa, K. W. Bowyer: Unraveling the effect of textured contact lenses on iris recognition. IEEE Transactions on Information Forensics and Security, Vol. 9, No. 5, pp. 851-862, 2014

[30] A. Adler: Vulnerabilities in Biometric Encryption Systems. LNCS, Springer 3546, pp. 1100-1109, 2005

[31] E.-C. Chang, R. Shen and F. W. Teo: Finding the Original Point Set Hidden among Chaff. In Proc. ACM Symp. ASIACCS'06, Taipei, Taiwan, pp. 182-188, 2006

[32] X. Boyen: Reusable cryptographic fuzzy extractors. In Proc. 11th ACM Conf. CCS, Washington, DC, pp. 82-91, 2004

[33] W. J. Scheirer, T. E. Boult: Cracking Fuzzy Vaults And Biometric Encryption. Biometric Consortium Conference, Baltimore, September 2007

[34] A. Cavoukian, A. Stoianov: Biometric Encryption – The New Breed of Untraceable Biometrics. In N.V Boulgouris et al., eds., Biometrics: fundamentals, theory, and systems, Wiley-IEEE Press, pp. 655-718, 2009

[35] D. Goodin: Breaking the iris scanner lock in Samsung's Galaxy S8 is laughably easy. Ars Technica, May 23, 2017, available online, last time visited December 2018

[36] P. Silva, E. Luz, R. Baeta, H. Pedrini, A. X. Falcao, D. Menotti, D: An approach to iris contact lens detection based on deep image representations. In Graphics, Patterns and Images (SIBGRAPI), 28th SIBGRAPI Conference on, 2015, pp. 157-164, IEEE

[37] V. K. Sinha, A. K. Gupta, Manish Mahajan: Detecting fake iris in iris biometric system. Digital Investigation, Vol. 25, pp. 97-104, 2018

[38] D. Gragnaniello, C. Sansone, L. Verdoliva: Iris liveness detection for mobile devices based on local descriptors. Pattern Recognition Letters, Vol. 57, pp. 81-87, 2015

[39] S. Barzut, M. Milosavljević: Jedan metod formiranja XOR biometrije otisaka prstiju Gaborovom filtracijom. In Sinteza 2014 – Impact of the Internet on Business Activities in Serbia and Worldwide, Belgrade, Singidunum University, Serbia, pp. 610-615, 2014