Biometrics and Cryptosystems: Biometric Cryptosystems

Alexander Voglsperger

Institute of Networks and Security, Johannes Kepler University, Altenberger Straße 69, Linz, 4020, Austria.

Contributing authors: alexander.voglsperger@pm.me;

Abstract

This seminar report is about the topic of *Biometric Cryptosystems*. We are going to discuss, what it is, how it works and why it is a good idea to use. Furthermore, major challenges and their respective solutions will be discussed. This includes cryptography, biometrics, key binding and generation, fuzzy vault and commitment, and a list of a few problems and issues.

Keywords: biometric cryptosystems, key binding, key generation, fuzzy vault, fuzzy commitment, secure sketch, fuzzy extractor

1 Introduction

A secure way to transmit data has become crucial as the world gets more and more into a digitally connected. Therefore, it is important to encrypt data and provide access only to users who can authenticate themselves. Cryptography aims to provide a secure way to communicate with one or multiple other person over an otherwise insecure channel. To achieve this it has to guarantee a set of goals — confidentiality, data integrity, authentication and non-repudiation [1].

Cryptosytems normally use a key to authenticate a user. This key is traditionally based on either knowledge- or proof-of-ownership. *Knowledge-based keys* are typically found as a password. But to maintain a proper key management, a truly random key has to be larger than the common passwords. As

with the Advanced Encryption Standard (AES) the key can be up to *256 Bits* large [2]. That means that keys are usually stored on a device as it becomes pretty hard to remember. To get the actual key, another form of authentication has to be provided. Often times, this is a password or pin code. As this method releases a key when the authenticity is proven, it is called *release-based* authentication. As with *possession-based keys*, the key is stored on a tamper-proofed device, e. g. a *security-key* or *smart card* which usually incorporate some form of protocol, like FIDO or U2F.

This may sound reasonably secure, but there are some downsides that must be considered when deploying them. Knowledge-based keys can be forgotten, or a third party can discover it using various methods. The part where a third party has knowledge of the key is especially important in a time when phishing and data leaks are not that uncommon and passwords like "*123456*" or "*password*" are still widely used. Possession-based keys can be lost or stolen, which also mitigates the security to some point [1] [3].

This is where *biometrics* come into play. Biometrics are defined as "Automated recognition of individuals based on their *behavioral* and *physiological* characteristics"[4]. This includes *non-static* characteristics like keystrokes, signatures, voice and *static* ones such as fingerprint, face, iris, etc. . This technology profits from user convenience, hard to copy, share and distribute and very difficult to lose or forget it. Furthermore, each user's biometrics different but equally secure, therefore everyone has the same level of security. Biometrics are not the ultimate go-to on their own, as they also suffer some problems, which are explained in chapter 2 [1] [3].

Biometric Cryptosystems are a way to take advantage of benefits in both fields. A wide spectrum is covered, as the objectives of biometrics and cryptography are very different. This is due to the nature of them. Cryptography can provide adjustable security, and biometrics allows for key-diversity and removes the need for typical knowledge- and possession-based key authentication. More detail is in chapter 5, where we talk about challenges and problems that come with biometric information and biometric cryptosystems [5].

2 Biometrics

As already defined in the introduction 1, individuals are recognized based on their behavioral and physiological characteristics. There is no optimal biometric, as each one has its strengths and weaknesses. This means that the used biometric detail depends on the requirements of the application it is used by. *Umut Uludag et. al.* created a table as seen in figure 1, where they compared different biometric identifiers on a few factors. It is possible to see that behavioral traits, e. g. *keystroke*, *signature* and *voice* are not that distinctive and are substitute to change. E. g. if a person has a cold and therefore the voice sounds different, or a person injures the hand and has to type differently than usual.

This is the reason, that behavioral features are often times used as secondary factors in combination with knowledge- or possession-based key. Furthermore, the performance is lower, as it usually requires more time to capture enough data to properly decide if the person can be positively verified [1] [3] [6].

Biometric identifier	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	H	L	M	H	L	H	H
Fingerprint	M	H	H	M	H	M	M
Hand geometry	M	M	M	H	M	M	M
Iris	H	H	H	M	H	L	L
Keystroke	L	L	L	M	L	M	M
Signature	L	L	L	H	L	H	H
Voice	M	L	L	M	L	H	H

Fig. 1 Comparison of Biometric Identifiers based on seven factors [3, Table 1]

2.1 Influence by the environment

As biometrics are properties tied to an individual person, they have to be scanned first. This means that the physical information has to be scanned and converted to a digital format. In this step, it is totally normal to have some invariance. E. g. a finger is never on exactly the same spot on a fingerprint scanner and being pressed onto the sensor slightly different, or the daylight will change the appearance when using face recognition. Furthermore, noise and other technical reasons impact the result of the scan [1] [6].

The method to combat this issue is Error Correcting Code (ECC), which introduces extra information for correcting errors that may occur. ECC is typically used when transferring information over noisy connections. When some information gets lost or changes, it can be reconstructed using this extra information. To generate such extra information, different algorithms such as *Reed Solomon Codes*, *Low-Density-Parity-Check Codes* etc. An important aspect when choosing such an algorithm is the ability to filter out noise and still be able to use the output for security measures [1].

Practically ECC are designed for communication and data storage. Therefore, *G. S. Karimovich et. al.* [7] mention that optimal correction codes only exist in a theoretical environment using certain assumptions.

2.2 Biometric Matcher

The *Biometric Matcher* is the system behind matching a stored template with a different template that was given as an input. As already described in chapter 2.1, a biometric feature can be influenced on the environment and are captured from an analog input. Therefore, an exact matching like with e. g. passwords is not helpful as there will always be some invariance. Furthermore, the placement of the biometric on the sensor also matters, as not all scanners provide placement constraints to remove too much variation. A practical biometric matcher typically performs an alignment on two pattern's features, respectively. The similarity is measured as a type of *matching score*, which includes a threshold for acceptable variation. The bigger the score, the more similarity between the two patterns [3].

As an example, *U. Uludag et. al* [3] describe the application of a biometric matcher based on a fingerprint's *minutiae*. The assumption in that example is that the minutiae-based representation is done using value triplets of (x, y, Θ) for easier processing. The first two values are coordinates, and the Θ is the orientation of the ridge at the given minutia. A live-scan of a good-quality image typically has $20 - 70$ minutiae, according to the authors. This provides a representation that is valid, compact, and robust. The input and stored template are placed on top of each other and then compared. If a minutia is within a certain threshold area of a corresponding minutia. The correspondence is calculated by the coordinates and the orientation. An example of the matching process is shown in figure 2.

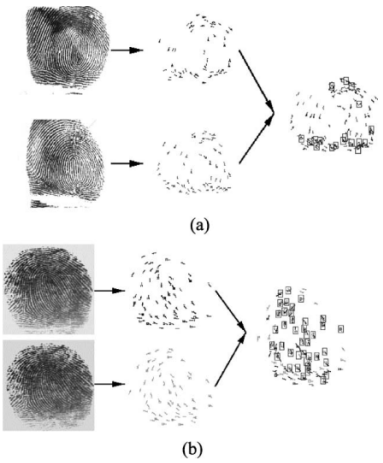

Fig. 2 Biometric Matching using a fingerprint's minutiae. *a)* Matching two different fingers. *b)* Matching the same finger. [3, Figure 5]

3 Biometric Cryptosystems

A *Biometric Cryptosystem* is the combination of *Biometrics* 2 and *Cryptography*. It takes the benefits of these topics and allows the usage of biometrics for cryptography. A main part is the dynamic generation of keys from/with biometric features. These keys can then be used for, e. g. authentication, encryption, etc. . In this chapter, the focus is on the things that have to be considered when using biometrics with cryptography [5]. Generation and usage of Keys is explained in chapter 4.

3.1 Securing Biometric Features

As biometric data can uniquely identify a person, it is important to store the template in a way, that it is computationally hard to recreate the original biometric template from stored data. On the other hand, it has to be relatively easy to generate the stored template. One way of securing a biometric feature is using a *Hash*-Function. Which can then also be used for authentication as shown in figure 3. Here it is important to use a hash function that is not invertible, as otherwise it won't secure the template. + It is also possible to hide the cryptographic key within the user's biometric template, like it is done with steganography, where the key is extracted upon valid authentication. However, the security is dependent on the secrecy of the hide and reveal process. Which can be compromised if the algorithm is deterministic and the attacker has enough data to find the common information. This is the reason procedures like *Key Binding* and *Key Generation* as explained in chapter 4.1 exist.

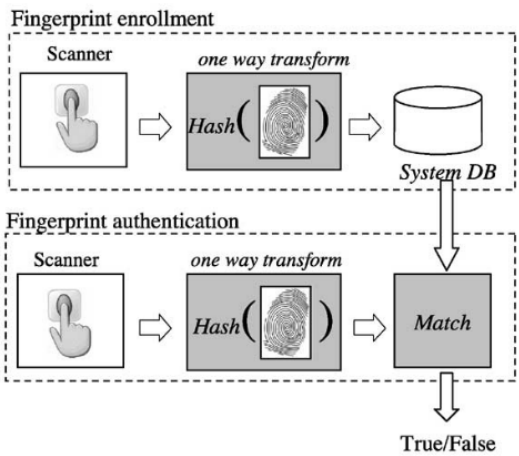

Fig. 3 Biometric authentication using Hash-Function [3, Figure 7]

3.2 Variation on different people

Another important part to keep in mind is the variation. To allow for a secure usage of the biometric feature, an algorithm has to be discriminative. In this context it means that there has to be a small intra-class variation but a high inter-class variation. In other terms, this means that multiple outputs of an algorithm on the same person should be the same or at least very similar. On the other hand, the output of an algorithm on different people should also produce vastly different results. This should also be taken into consideration when choosing a [6] [8].

3.3 Cancelable Biometrics

Biometrics have the disadvantage, that it cannot be replaced as easily as a knowledge- or possession-based key. This is due to the fact, that biometrics are natural information. However, it is possible to create biometrics which have protection added to allow such a feature. The two main categories are distinguished by one-way-transformations and salting. With one-way-transformations, e. g. hashing is introduced, which changes the actual input biometric. Salting introduces some extra information which alters the input biometric, which is stored alongside the output of the combination. This results in the original biometric never being stored. But the error rate increases significantly as the changes introduces variability when matching [3] [7].

4 Biometric Keys

Biometric keys use biometric features to provide or generate keys, which can be used for authentication or encryption. This is based on different methods to acquire and store keys and will be described next.

4.1 Key Release, Key Binding and Key Generation

- Key Release:
  A cryptographic key is stored in a database and after a successful biometric based authentication, the key gets released and is ready to use. This is similar to how password-based key-release mechanisms work. But as the key and biometric template still have to be stored somewhere, it is important to protect them against theft. Some methods are elaborated in more detail in chapter 4.2.
- Key Binding:
  A key with high randomness and entropy is generated and then hidden inside biometrics. This is called *helper data*, which then securely is stored in a database. The helper data is often generated from biometric features. In order to retrieve the key from this helper data, the used biometrics has to be provided again. To protect against incorrect bits or errors in the data, ECC is added in the process, to retrieve accurate keys that are free from errors.

- Key Generation:
  This works similar to the *Key Binding* process, but instead of some stored helper data, everything is retrieved from the biometric template and the key is generated directly out of it. An example is shown in chapter 4.4. In this approach, it is also important to implement some way of ECC [7].

4.2 Fuzzy Vault and Fuzzy Commitment

- Fuzzy Vault:
  A *fuzzy vault* is a cryptographic construct that secures a private key using a biometric feature, e. g. a fingerprint's minutiae points. The security is based on the polynomial reconstruction problem. The security of this process can be improved further by securing the polynomial evaluations using a *Fuzzy Commitment* scheme, as it makes it more difficult to decode even if the correct set of minutiae is selected [3] [9].
- Fuzzy Commitment:
  A *fuzzy commitment* is designed to tolerate more variation in the biometric template, by splitting it when storing the template and incorporating ECC into the retrieval process. It decomposes a given template into a *secret key* and a *secure sketch*. On a valid authentication, the secure sketch and biometric query are combined, and ECC are applied to retrieve the secret key [3] [9].

4.3 Secure Sketch and Fuzzy Extraction

- Secure Sketch:
  A *Secure Sketch* is a technique to create public helper information out of some input. The public information does not reveal its input, but still allows recovery of the original input with extra information that is similar enough. This method allows reliable reproduction of error-prone input without the security risk associated with storage [1] [9].
- Fuzzy Extractor:
  A *Fuzzy Extractor* is a tool that consistently produces values with a high amount of uniform randomness from its input. The extraction allows for the input to change a certain amount and still produces the same output. This allows the use in the generation of cryptographic keys [1] [9].

4.4 Generating a Key from Minutiae-Points

Let's take key-generation from *Minutiae-Points*, which is depicted by *R. Ranjan et. al.* [5]. Minutiae-points are local features that represent the ending or bifurcation of the ridges on a fingerprint.

These points are extracted using different image processing algorithms. Enhancement of the image is done using *histogram equalization* and filtering. To get the ridges, the image is converted into gray-scale and this then gets transformed into binary values. To properly extract the minutiae points, the ridges are thinned to be only one pixel wide, which is done using an algorithm

that removes redundant pixels over several iterations. With this result, the extraction of minutiae is fairly easy. But before generating the key, a mechanism to remove false minutiae is essential, as misreadings from the sensor and even occasional artifacts from the steps before would distort the key.

Using the pre-processed minutiae points, the actual key-generation can begin:

1. Look for a point H with the highest $(x + y)$
2. Draw a line from coordinates $(0, 0)$ to H, which is the *Key-Vector* K_V
3. Sort and store the minutiae points in an array A
4. ```
   val = Key-Length / Size of Minutiae-points set
   vector = Key-Length % Size of Minutiae-points set
   ```

5. ```
   K = []
   for i=1 to val:
       for j=1 to (Size of Minutiae-points set):
           x = A[i,j]
           if(x above L) -> K.append(0)
           else -> K.append(1)
   ```

6. Final key then is the concatenation of (length of vector) and K

5 Issues and Challenges

Despite the benefits that come with biometric cryptosystems, they also face a number of important issues and challenges that need to be considered when evaluating their effectiveness suitability for a given use case. This includes choosing certain methods, when creating a biometric system, security, and privacy concerns that come with them. Also, potential attacks are important to think of, as understanding them allows creating them as secure and effective as possible.

5.1 Variables of interest

Some characteristics of biometrics and the combination into biometric cryptosystems has some issues which are important to mention. The following list contains some examples given by *G. S. Karimovich et. al.* [7].

- Issues of Alignment:
  Alignment issues are some fundamental challenges which has already described in chapter 2.2. The alignment almost certainly differs most of the time, which causes matching to be more difficult. It is mentioned that the current algorithm-based alignment-methods are not quite enough for biometric cryptosystems. Methods that use deep-learning also exist with provide relatively good results but are time-consuming.

- Improper ECC:
  As already described in chapter 2.1, biometric input causes noise and invariance. That is the reason for the usage of ECC. Even though it has a valid reason to be used, it is also important to use an algorithm that still provides enough security after the application of ECC. Another issue is, that the correcting algorithm cannot differentiate between a legitimate user and one that is not. It tries to apply the correcting either way. A potential attack is described in chapter 5.3.

5.2 Security and Privacy Concerns

As described by *R. Asthana et. al.* [6], biometrics in combination with cryptosystem have some security and privacy concerns, which are important to consider.

- Authenticity but no Secrecy:
  As a biometric detail is part of the human body, it is a personal attribute and therefore provides authenticity. On the other hand, biometric information can be gathered without the consent of the person. A password has to be at least shared in some way, as it is secret but does not prove if a person is authentic.
- Cancel / Revoke biometrics:
  If a biometric gets compromised, it cannot be reset, as it is with passwords or other keys where a new one is issued. However, there are some techniques that can be used to create cancelable biometrics, which are discussed in chapter 3.3.
- Compromised forever:
  As a biometric is a person's characteristic of permanent nature, if it gets compromised once, it will be compromised forever. This means that one application gets compromised and a non-secure template is leaked, it is possible to use that template with other applications and get access too. Therefore, it is important to store the biometric template securely, as described in 3.1.
- Tracking across platforms:
  Biometrics can be used to uniquely identify a person, as mentioned in chapter 3.1 it is possible to track a person. This is especially the case when a person uses the same biometric feature in multiple application in a short period of time. Another concern is the sharing of biometric templates with and between intelligence agencies.

5.3 Potential Attacks

It requires some effort to create a reasonable secure system, but usually everything as some kind of attack vector. This is also the case with biometric cryptosystems and cancelable biometrics, which have some more general possibilities to get attacked and some that are introduces due to the design decisions.

The list below presents some examples of attacks that *G. S. Karimovich et. al.* [7] have identified.

- Brute-Force Attacks:
  As with almost any security system, when using biometric cryptosystems it is possible to brute force the keys generated from biometrics. Depending on the type of biometric feature used, it can create reasonable fast results. The authors' reason are keys with low entropy, which are generated by biometrics with not enough reliable data. This includes almost every biometric feature except iris and fingerprint.
- False-Acceptance Attack:
  As biometrics have to be compared on how similar two samples are, they introduce a False Acceptance Rate (FAR) which authenticates a non-legitimate person. Depending on the implementation, this FAR varies. As an example given by the authors, it statistically only requires 10^4 samples when the FAR is 0.01%. Meaning that with a large enough dataset of the biometric feature, this is doable in a somewhat reasonable time.
- ECC Abuse Attack:
  In chapter 5.1 it was mentioned that ECC can also be abused. Specifically, this attack targets *Fuzzy Vault* and *Fuzzy Commitment* from chapter 4.2, which incorporate ECC. The attack itself abuses the correction system, where the algorithm cannot distinguish between legitimate and other input. When the input is near the original information, it recovers mistaken bits into original ones and then authorizes an invalid user.
- Blended Substitution Attack:
  This attack works by combining the intruder's template with the targeted person's template. This united template then in return can allow potential authentication. The ease of retrieving a template and success depends on the type of biometrics used in the target system.

6 Conclusion

In this seminar report, we briefly covered *Cryptosystems* for authentication and encryption. Further, the two main categories, *knowledge-based* and *possession-based*. We elaborated why secure keys usually have to be securely stored somewhere. Be it protection using a password or storage on a tamper-proof device. This was followed with some reasoning on why it may seem secure but actually has some flaws that should not be neglected, e. g. insecure passwords or stolen/lost possession-based keys.

Next, a definition of *Biometrics* 2 and categorization into behavioral and physiological was given. Example biometric features for the two categorizations were also given. Also, some benefits of biometrics such as user convenience and not being able to forget/loose it. As also noted, biometrics are not the ultimate-go-to, as they all have different strengths and weaknesses. Therefore, we highlighted a table for comparing biometric identifiers based on

a few important factors. Also, important is the influence by the environment and how to mitigate some issues using ECC. The difficulty of matching biometrics was also explained by the means of an example using fingerprints.

With the information gained by talking about cryptosystems and biometrics, we described how the combination of the two topic's benefits achieves a useful system resulting in *Biometric Cryptosystems* 3. We mentioned that it is important to secure biometric features as they can uniquely identify a person and that the biometric should be processed by a one-way-function. The intra- and inter-variation was also briefly covered, as this must be considered when creating an algorithm. Another brief and more general topic were cancelable biometrics, which uses hashing or salting.

Still being related to biometric cryptosystems we described a few concepts such as *Key-Binding*, *Fuzzy Vault* and *Secure Sketch* 4. This was followed by an example on key-generation from a fingerprint's minutiae points.

At last, we aggregated some issues and challenges from other papers. Variables of interest includes things that should be considered, such as choosing a ECC that corrects issues but still provides appropriate security. Security and Privacy Concerns includes issues where a biometric provides authenticity but no security. And Potential Attacks describes attacks such as the abuse of ECC or attack on the FAR. Given these issues and challenges, it is clear that biometric cryptosystems are far from perfect and still a significant amount of research has to be done in order to ensure continuous effectiveness and usability in a wide range of applications.

Overall, the topic of *biometric cryptosystems* of includes many areas, which will have a significant impact in the future. Advancement in artificial intelligence making recognition more reliable and accurate. Research in Quantum-computing will result in quantum-safe encryption methods. Also, the field of combining multiple features possibly leads to more secure biometric cryptosystems.

References

[1] Teoh, A.B.J., Kim, J.: Error correction codes for biometric cryptosystem: An overview. Information and Communications Magazine **32**(6), 39–49 (2015)

[2] of Standards, N.I., Technology: Advanced Encryption Standard (AES). NIST FIPS PUB **197** (2001)

[3] Uludag, U., Pankanti, S., Prabhakar, S., Jain, A.K.: Biometric cryptosystems: issues and challenges. Proceedings of the IEEE **92**(6), 948–960 (2004). https://doi.org/10.1109/JPROC.2004.827372

[4] International Standardization Organization: ISO SC 37—Standing Document 2 (2006)

[5] Ranjan, R., Singh, S.K.: Improved and innovative key generation algorithms for biometric cryptosystems. In: 2013 3rd IEEE International Advance Computing Conference (IACC), pp. 943–946 (2013). https://doi.org/10.1109/IAdCC.2013.6514353

[6] Asthana, R., Walia, G.S., Gupta, A.: A novel biometric crypto system based on cryptographic key binding with user biometrics. Multimedia Systems **27**(5), 877–891 (2021). https://doi.org/10.1007/s00530-021-00768-8

[7] Karimovich, G.S., Turakulovich, K.Z.: Biometric cryptosystems: Open issues and challenges. In: 2016 International Conference on Information Science and Communications Technologies (ICISCT), pp. 1–3 (2016). https://doi.org/10.1109/ICISCT.2016.7777408

[8] Mai, G., Lim, M.-H., Yuen, P.C.: Binary feature fusion for discriminative and secure multi-biometric cryptosystems. Image and Vision Computing **58**, 254–265 (2017). https://doi.org/10.1016/j.imavis.2016.11.011

[9] Chafia, F., Salim, C., Farid, B.: A biometric crypto-system for authentication. In: 2010 International Conference on Machine and Web Intelligence, pp. 434–438 (2010). https://doi.org/10.1109/ICMWI.2010.5648101