CRYPTOGRAPHIC KEYS FROM NOISY DATA
THEORY AND APPLICATIONS

Ileana Buhan

Composition of the Graduation Committee:

Prof. Dr. P.H. Hartel	(University of Twente)
Dr. Ir. R.N.J. Veldhuis	(University of Twente)
Prof. Dr. Ir. B. Haverkort	(University of Twente)
Prof. Dr. Ir. C.H. Slump	(University of Twente)
Prof. Dr. Ir. R.L. Lagendijk	(Deft University of Technology)
Prof. Dr. Ir. A.J. Han Vinck	(University of Duisburg-Essen, Germany)
Dr. J.M. Doumen	(Irdeto)
Dr. Ir. T.A.M. Kevenaar	(Philips Research)

University of Twente
The Netherlands

Distributed and Embedded Security Group
P.O. Box 217, 7500 AE, Enschede, The Netherlands.

This research is supported by the Technology Foundation STW, applied science division of NWO and the technology programme of the Ministry of Economic Affairs under project number TIT.6323.

CTIT
Centre for Telematics and Information Technology

CTIT PhD Thesis Series Number 08-129
Centre for Telematics and Information Technology (CTIT)
P.O. Box 217-7500 AE Enschede-The Netherlands.

INSTITUUT VOOR PROGRAMMATUURKUNDE EN ALGORITMIEK
ipa

IPA: 2008-28
The work in this thesis has been carried out under the auspices of the research school IPA (Institute for Programming research and Algorithms).

ISBN: 97890-365-2738-5
ISSN:1381-3617

Cover: *Two-dimensional 7-hexagonal tiling* for encoding noisy data proposed in *Chapter* 4 of this thesis.
Design: Ştefan Dulman and Ileana Buhan
Printed by Wöhrmann Print Service.

Copyright © 2008 Ileana Buhan, Enschede, The Netherlands.

CRYPTOGRAPHIC KEYS FROM NOISY DATA
THEORY AND APPLICATIONS

DISSERTATION

to obtain
the doctor's degree at the University of Twente,
on the authority of the rector magnificus,
prof. dr. W.H.M. Zijm,
on account of the decision of the graduation committee,
to be publicly defended
on Thursday, October 23, 2008 at 15.00

by

Ileana Rozalia Buhan

born on 14 March 1979,
Târgu Lăpuş, Maramureş, România

The dissertation is approved by:

Prof. Dr. P.H. Hartel (promotor) and
Dr.Ir. R.N.J. Veldhuis (assistant promotor)

I dedicate this thesis to my family

Abstract

Biometric security systems that verify a person's identity by scanning fingers, hands, eye or face are becoming more and more common. As a result biometrics is one of the fastest growing industries. Applications for biometrics range from homeland security (for example the European biometric passport), physical access to various facilities (banks, amusement parks, office buildings, computer terminals, etc) and health and social services.

Utilizing biometrics for personal authentication is more convenient and than current methods such as passwords or PINs (nothing to carry or remember). Another important advantage of biometric authentication is that it links events to a user (passwords or token can be lost or stolen) and is becoming socially acceptable and inexpensive. Biometric authentication requires comparing a registered or enrolled biometric sample (biometric template or identifier) against a newly captured biometric sample (for example, a fingerprint captured during a login).

However, biometric authentication is not perfect and the output of a biometric authentication system can be subject to errors due to imperfections of the classification algorithm, poor quality of biometric samples, or an intruder who has tampered with the biometric authentication systems. Although biometric authentication is intended primarily to enhance security, storing biometric information in a database introduces new security and privacy risks, which increase if the database is connected to a network. This is the case in most practical situations.

The most severe threats are: *impersonation*, where an attacker steals templates from a database and constructs a synthetic biometric sample that passes authentication; *irrevocability*, where once compromised, biometrics cannot be updated or reissued; *privacy*, which is the exposure of sensitive personal information without the consent of the owner. A solution to these threats is to apply template-protection techniques, which make it hard for an attacker to recover the biometric data from the templates.

This thesis looks at security aspects of biometric authentication and proposes solutions to mitigate the risk of an attacker who tries to misuse biometric information or who bypasses modules of biometric systems to achieve his malicious goals.

Our contribution is threefold. Firstly we propose 3W-tree, an analysis tool used to *identify critical attack scenarios* for a biometric system. We apply the 3W-tree design tool to the SmartGun biometric recognition system with the purpose of identifying critical security issues. Secondly, we explore the challenges of *secure template protection*, which are both theoretical and practical and we put forward solutions to part of the issues. Thirdly, we present a practical solution to the *secure template transfer*, which should allow transfer of the biometric traits between two biometrically enabled devices when no security infrastructure is available and the users are no security experts.

Acknowledgements

The last four years were an unforgettable experience and I would not change a thing! I was lucky to be surrounded by wonderful people who made my life fun, memorable and from whom I have learned so much.

Pieter Hartel, my promotor and daily supervisor made it all happen. He believed in me from the beginning and gave me the opportunity to do a PhD in his group. He tought me how to write papers, organize and present my ideas. At the same time he gave me the freedom to explore my own ideas and whenever I was stuck he managed to put me back on the right track. *Pieter* and *Marijke*, his wife, have at all times a joy of life which makes it a pleasure to be around them.

Raymond Veldhuis, my assistant promoter introduced me to the world of biometrics and never ceased to amaze me with his creativity and intelligence. He was always helpful and made me feel welcome every time I visited his group. *Jeroen Doumen* is an important part of my PhD life. We shared many discussions and experiences in the far corners of this world. He always challenged me and some of our discussions closely resemble a duel. We both enjoy it and became friends. I thank *Tom Kevenaar, Inald Lagendijk, Boudewijn Haverkort, Kees Slump* for assessing my manuscript as members of my graduation committee.

The Romanian group in Twente has a special place in my hart. *Raluca* and *Mihai*, who are true friends and were there for me when times got rough and with whom I have many happy memories: the discovery of Australia (a unforgettable 5000 km trip), the wild dancing parties, the many weekends spent together or the 75 km bike trip we did in one day. *Andreea* and *Blas* came later to Twente. Nonetheless they became actively involved in all of our group activities. They gave us dancing lessons and exotic recipes for cocktails and coffee. *Andreea* was the official translator for all our foreign trips as she speaks fluently no less then 6 languages. Eugen, introduced me to the world of flying and promised to take me one day up in the sky. *Diana* and *Andrei* and now little *Adişor*, who were there, always helpful and warm. I am lucky to have met you all.

Of course I cannot forget my dear Turkish friends: *Ozlem* and *Mustafa*, always full of joy, *Aysegul* and *Kamil*, with their warm smiles, *Seckin*, *Cem* and *Murat*

my neighbors. Without them the many volleyball games, movie nights, and all the parties would have not been the same. My dear Indian friends *Supryio*, *Anindita*, *Kavitha*, *Kiran*, and *Vasughi* introduced me to the fascinating Indian culture either by cooking delicious spicy food or by teaching us exotic Indian dances, which we had to perform in public! I especially want to thank *Supriyo* for his help in the last stages of the preparation of the manuscript. To *Michel* I am in debt for his patience and dedication in teaching me Dutch. *Lodewijk* provided guidance in the interaction with *belastingdiesnt* and still is my *apple* mentor.

I shared my office with many people and although in the beginning it was a bit crowded it was all worth it as I got to know them: *Ari*, *Anca*, *Marcin*, *Richard*, *Luan*, *Mohsen*, *Saeed* and *Trajce*, who has the gift of bringing people together. *Qiang*, who came during the last year of my PhD and who got immediately involved, his sharp observations had a definite influence in improving my work. Our group would have not been the same without *Sandro*, *Emmanuele* and *Damiano* or the italians as we used to call them and with whom I share the same passion for good coffee. *Mohammed*, was my last office colleague and I enjoyed sharing the same office space and our deep, sometimes philosophical conversations.

I am obliged to *Wolter* who accepted, without hesitation to translate the abstract of this thesis and who compiled a very detailed list of things to do before the defense of my thesis. I am grateful to the group's secretaries *Marlous*, *Nicole*, *Thelma* and *Nienke* for being always helpful especially, when managing the financial matters or booking conference trips.

The *Signals and Systems* group was my second home. *Xiaoxin*, *Qian*, *Wessel*, *Bas* and *Chun* were always helpful. They patiently explained how biometric classification works and provided me with biometric data, sometimes by working overtime. Their contribution was vital for validating my results.

Andre Hartgers from KLPD always told wonderful stories about police and I want to thank him for his support for the project. I must say that our user committee meetings were held in exotic places, we managed to always have fun and at the same time we learned so much from: *Asker Bazen*, *Ruud van Munster*, *Thierry Jacobs*, *Frank Karelse*, *Michael Schumacher*, *Pieter Dalhammer* and *Cas Damen*.

To my colleagues and professors from the North University of Baia Mare, my home university, who introduced me to the fascinating world of science I am indebt to choosing academia as a carrier and welcomed me back.

Vreau să le mulţumesc părinţilor mei care au fost întotdeauna alături de mine şi care, în ciuda distanţei sunt atât de aproape, ei au crezut în mine şi m-au susţinut în toate deciziile mele.

Nagy szerencsém volt hogy a legjobg nagymama es nagytata a világból legyen. Mindent nagyon szépen köszönöm!

Luminiţa, which in romanian means little light is more then that to me. She is the best sister I could have asked for and I know that no matter what comes ahead this will not change. I finally thank *Ştefan*, my future husband who is the newest member of my family. His love, support and sometimes simply his presence makes my day brighter and I cannot imagine the last years or the years to come without him.

25 September 2008
Enschede, The Netherlands

Contents

Chapter 1

Introduction

Biometrics are automated methods that allow the recognition of a person based on their physiological or behavioral characteristics. Biometric based technologies offer an elegant solution for human machine authentication. With biometrics, events can be linked directly to a person while passwords or tokens maybe used by others than the authorized user. Biometrics are convenient and user friendly as biometric identifiers do not have to be remembered and cannot be lost.

There are two main concerns regarding biometric authentication. The first concern is the accuracy of biometric recognition. It is known that due to natural variations (and noise) a biometric system may falsely accept or reject users. The second concern is related to the fact that once a biometric identifier is compromised it cannot be used again for biometric authentication because a user cannot renew his biometric.

This thesis looks at security aspects of biometric authentication and proposes solutions to mitigate the risk of an attacker who tries to misuse biometric information or bypass modules of biometric systems to achieve his malicious goals.

User authentication is the process of verifying the claimed identity of a user by a computer system, often as a prerequisite to allow access to resources in the system. For the purpose of user authentication one can use what a user *knows*, for example a password or a PIN, what a user *has*, typically a token such as a smartcard, or something the user *is*, in other words a biometric identifier.

Biometric authentication refers to any security system that uses measurable human physiological or behavioral characteristics to determine human identity. Ideally these characteristics should be measurable, unique, invariable over time,

and should not be easily duplicated. Biometric authentication systems are used in two ways: to identify people and to verify the claimed identity of registered users. Typical application domains include laptop login, access to airports, banks, military installations, etc.

For biometric recognition one needs several components: firstly, a reader or scanning device which measures the biometric identifier (a camera for face or a recorder for voice) secondly, software that converts the scanned information into digital form and compares the biometric identifiers and thirdly, a database that stores the biometric data for comparison.

During *enrollment* the biometric system learns the identity of its users and stores their identities in a database. Enrollment is usually performed once in the lifetime time of the biometric system. During *authentication* the biometric system matches the measured biometric identifiers to the ones stored in the database and decides whether they come from the same person. Authentication is performed every time the identity of a person is verified.

For most biometric systems that verify the identity of the user before allowing him access to protected resources the main threat is an *un*authorized user gaining access to the system, normally called a *false acceptance*. The main goal of an attacker is to "convince" the biometric authentication system that he is another person with access to the protected resources. An authorized user who is *falsely rejected* by the biometric system on the other hand, represents merely a convenience problem since the user can employ an alternative identity verification method to access the protected resources.

For some applications like controlling access to a military installation, a *low false accept rate* thus *high security* is more important whereas for other applications like laptop login, a *low false reject rate* thus a *user friendly system* is more appropriate. It is known that these requirements are conflicting and lowering the error rates of a biometric recognition system is the main focus of most research on biometrics [16].

1.1 Research Question

A biometric authentication system is intended to recognize whether the submitted biometric data corresponds to the features deposited in the database. Any malfunction in performing the designated task is an *error*.

Examples of events that may cause an error in the functioning of a biometric recognition system are listed below. The examples given are by no means exhaustive, our purpose is to illustrate the diversity of things that can go wrong.

Example 1. Variations in the biometric data exceed the expected threshold. A user changes her hair style drastically between enrollment and authentication, so the face recognition biometric recognition system is unable to correctly identify the user. A fingerprint biometric authentication system finds the hands of the user dirty or sweaty and as a result the images collected by the sensor are degraded to the extent where authentication is no longer possible.

Example 2. Bogus identities in the database. The service provided by a biometric recognition system is correct, however, the database has been altered by an intruder who has introduced new bogus identities. As a result of this attack, an attacker may assume different identities and roles.

Example 3. Biometric templates stored the in the database. An intruder reads the biometric identity in the database and reconstructs the original biometric identifier. Matsumoto [51] shows how to build a gummy fingerprint from the minutia information stored as a biometric template in the case of fingerprint recognition system. As a result, the attacker can construct a gummy fingerprint which is falsely accepted by the biometric recognition system.

In the above examples there are two classes of errors. The first class consists of *nonmalicious errors* when the user is honest and the biometric system has not been tampered with (e.g. dirty or sweaty hands) as in the scenario presented in the first example. The error rates of a biometric system are determined by the accuracy with which the matching engine can determine the similarity between a measured sample and the expected value which is stored in the database. It is commonly agreed that there is no error free biometric system. Minimizing the error rates of a biometric recognition system is the main area of *biometric research* and involves aspects of signal processing and pattern recognition.

The second class consists of *malicious errors* where an intruder is the cause of these errors, an example in this sense is presenting a fake gummy fingerprint to the sensor of a fingerprint recognition system, as in the attack scenarios described in the second and third examples. Although the system correctly recognizes the presented fingerprint, the user who presents the fake fingerprint is falsely accepted by the biometric system.

Malicious errors can have two essentially different causes. The first cause is *abuse of physical modules*, where an attacker targets a physical module in the biometric recognition system architecture for example the sensor. The target of the attack is to force the biometric system to produce a false reject or a false accept at the will of the attacker. The second cause is *abuse of biometric information* where biometric data is used to extract or correlate information about the user to whom

it belongs, for example personal preferences for on-line applications, medical information, etc. Thus we have the following research question:

Main Research Question:	How can we mitigate the risk of *malicious* errors in a biometric recognition system?

There are a few difficulties in answering the main research question. Firstly, perfect security is too expensive and unattainable. When defending against all possible attacks the cost is prohibitive. Thus, usually the most likely threats are identified and defenses are implemented following the best case scenario philosophy: *"the best you can buy"*.

Secondly, defenses are implemented to counter the capabilities of an intruder. In the security world it is commonly accepted that there is no security measure possible when the intruder is highly motivated and has unlimited resources (criminal organizations).

Thirdly, defense methods are typically implemented with the application and system architecture in mind. For a biometric recognition system the particular architecture can vary greatly, according to the intended use scenario. Thus each instance of a biometric recognition system has to be evaluated independently and different defense strategies should be designed in each case.

To answer the main research question we need to understand the application. The implementation, usage scenario of the biometric identifier and the corresponding points of vulnerability influence the defense method for each application. In the following section we introduce the application which motivates the research presented in the rest of the thesis.

1.2 Secure Grip Application

The research of this thesis is done in the context of the Secure Grip project, which focuses on the design, implementation and evaluation of a prototype grip-pattern recognition system for the development of a smart gun, intended for use by the police. Grip-pattern recognition ensures that the weapon can only be fired by an authorized user. The gun should be useless in the hands of anyone else who might intend to misuse the weapon.

We propose to use biometric recognition to make a gun smart. The grip of our SmartGun is covered with a grid of pressure sensors that are protected against wear and tear. These sensors are capable of measuring both the static pressure pattern as a function of the place where the gun is being held (representing the po-

sitions and shapes of the fingers on the grip and the pressure exerted by them) and the dynamic pressure pattern (i.e. the pressure variation) as a function of place and time when the grip tightens prior to pulling the trigger. The main research question is refined now for the architecture of the SmartGun:

Refined Research Question:	How can we mitigate the risk of *malicious* errors in the in the architecture of the biometric SmartGun?

The SmartGun is a new type of biometric application, for which a false rejection is the most serious threat as this would result in a police officer not being able to use the weapon when necessary. For a police officer to trust his gun the false reject rate must be below 10^{-4}, which is the accepted failure rate for police weapons in use. For the SmartGun the overall error rate (mechanical and grip) should remain below 10^{4}.

To answer the refined research question we propose a three stage process: the first step is the *identification* of the relevant causes of errors for a *false rejection oriented* biometric recognition system, the second step is the *classification* of the identified causes according to their effect on the security of the system, and the third step is the *analysis* of threats.

1.3 Thesis Overview

Mitigating the risks of malicious errors is the main topic of this thesis. In *Chapter* 2 we give an overview of the threat analysis for a generic biometric recognition system architecture and we propose a systematic method, the 3W-tree (Who, hoW and What) to identify and classify relevant threats for a false rejection oriented biometric recognition system architecture. The result of the 3W-tree analysis indicates two possible research directions.

The first is SECURE TEMPLATE STORAGE, which states that it should not be possible to reconstruct the biometric identifier from the data stored in the gun. In *Chapter* 3 and *Chapter* 4 we focus on the theoretical aspects of protection techniques for noisy data, which has applications in the area of secure storage of biometric information and cryptographic key extraction from noisy data.

The second research direction is SECURE TEMPLATE TRANSFER, which states that it should be possible for two police officers to exchange their biometric identifiers between two guns when no security infrastructure is available and when the users are no security experts. In *Chapter* 5 we propose a new protocol which

allows two users to construct a communication key from noisy data, in an ad-hoc scenario. Finally, we summarize the contributions in the thesis and suggest future work in *Chapter* 6. We now elaborate further on the contributions of each chapter in some detail.

CHAPTER 2. We propose 3W-tree (Who, What, hoW) for identifying false rejection threats to biometric security systems. Analysis based on a 3W-tree leads to concrete questions regarding the security of the system. Questions raised by other methods (e.g. attack trees) do not lead to the same level of specific questions. Our method is more concrete than other methods because we make explicit assumptions about the generic architecture of the system, thus exposing all main components in the architecture that are vulnerable to attack. Our method is not less general than other methods because other architectural assumptions can be plugged in easily. Our method is intended to be used as a design aid.

To demonstrate the potential of 3W-tree in the security analysis of the biometric system we apply the 3W-tree to the biometric SmartGun. As a result of *our analysis* we identify two research directions.

The first research direction is security with noisy data. Weapons may be stolen or lost. Therefore, it is important to store the biometric template in a protected form. A solution could be to store the biometric template in tamper resistant hardware. However, due to the weight and space restrictions this is not desirable. Another solution would be to use cryptographic techniques to store the biometric templates in encrypted format, such that an attacker cannot construct a valid biometric identifier from the information stored in the gun. However this solution is not applicable due to the natural variation in biometric measurements which renders comparison in the encrypted domain difficult.

The second identified research direction is spontaneous secure interaction. Police officers often work in teams so that appropriate templates can be loaded into the weapons at the police station. However, in an emergency situation this is not possible; in this case police officers have to team up unprepared and exchange templates in the field, such that all weapons are available for all police officers in the team. Biometric data is sensitive information thus during the exchange the templates must be protected. Officers may work with colleagues from other departments, even from neighboring countries, so a shared key, or a public key infrastructure where the certificate associated with these keys must be verifiable on-line is not realistic. Also, one cannot expect a police officer to perform some complicated interfacing operation with his gun in the field.

The theoretical concept of 3W-tree appears in [2].

CHAPTER 3. The use of biometric features as key material in security protocols has often been suggested to avoid long passwords or keys. However, the use of biometrics in cryptography does not come without problems. It is known that biometric information lacks uniformity and it is not exactly reproducible, which is the opposite of what is considered suitable for a cryptographic key. Fuzzy extractors allow cryptographic keys to be generated from noisy, non-uniform biometric data. They can be used to authenticate a user to a server without storing her biometric data directly. This is important because the server may not be trusted.

The contribution of this chapter is related to the SECURE TEMPLATE STORAGE research direction. We show that there exists a relation between the strength of the keys extracted from biometric data and the quality of the biometric data in terms of FAR (false acceptance rate) and FRR (false rejection rate). We estimate the min-entropy values for the cryptographic keys derived from continuous distributions, thus linking real-life continuous biometric distributions to methods like fuzzy extractors in a new construction we call the *cs*-fuzzy extractor. We relate the min-entropy of the cryptographic keys to the FAR, thus formalizing the intuition that the min-entropy of an extracted key (in bits) cannot be more than $-\log_2(\mathrm{FAR})$. This last point motivates research into improving the FAR (i.e., the classification results) of biometric systems. Also, from a practical perspective it is useful to evaluate the potential of the biometric data in the context of a specific cryptographic application. The concept of *cs*-fuzzy extractors appears in [5] while the extended version, which includes examples appears in [8].

CHAPTER 4. When using a fuzzy extractor for a specific application, extra features are needed, such as the renewability of the extracted strings, and the ability to use the fuzzy extractor directly on continuous input data instead of discrete data. The contribution of this chapter is related to the problem of SECURE TEMPLATE STORAGE. We propose the fuzzy embedder as a generalization of the fuzzy extractor construction. A fuzzy embedder naturally supports renewability, as it allows a key to be embedded instead of extracted. Moreover, a fuzzy embedder supports direct analysis of quantization effects, as it makes no limiting assumptions about the nature of the input source. We give a general construction for fuzzy embedders based on the technique of quantization index modulation (QIM). We present and analyze, as an exercise, two constructions in the two dimensional space. Our 6-hexagonal tiling construction offers $((\log_2 6)/2 - 1)$ approximately 0.3 extra bits per dimension of the space compared to the known square quantization based fuzzy extractor. The other construction, the 7-hexagonal tiling, turns out to be optimal from resilience to noise perspective. The contribution of this chapter appears in [6].

CHAPTER 5. Mobile devices are designed to interact anytime, anywhere. In many scenarios however it is desirable to associate devices in a secure way. For example when sharing contact information via a wireless link in an unsecured environment. This problem is known in the literature as secure device association. Solutions have to be specifically designed such that secure association can be realized between previously unassociated devices. Security means that the solution must offer guarantees of the association partner identity and the solution must be resistant to eavesdropping and to a man-in-the-middle attack. The ideal solution should provide a balance between security and user friendliness.

The contribution of this chapter is related to the problem SECURE TEMPLATE TRANSFER for which we propose a practical solution where biometrics are used to establish a common key between the pairing devices. Our approach has at least two major advantages over related work. Firstly, it offers the possibility to transfer trust from humans to machines without any available security infrastructure. Biometric recognition offers physical validation, thus guaranteeing the identity of a device owner. Secondly, the process is short and user friendly. In the pairing protocol the keys extracted from biometric data are combined to form a session key.

The idea is both simple and effective. Suppose that two users wish to set up a secure communication channel. Both own a biometrically enabled handheld device (for example with face recognition or grip pattern biometrics). Both devices are equipped with a biometric sensor and a short range radio. Each device is capable of recognizing its owner for example by face recognition. Then the users take each others picture. Each device now contains a genuine template of its owner and a measurement that approximates the template of the other user. The idea is that each device calculates a common key from the owner template and the guest measurement. In our solution, all Alice has to do to set up a secure communication with Bob is to take a picture of him and let Bob take a picture of her. The protocol is even more general: it can be applied on any type of biometric channel, including grip pattern biometrics.

We evaluate the performance of the protocol from three different perspectives. Firstly, we analyze the security of the protocol against two types of adversaries Eve which has computational capabilities and Charlie a Dolev-Yao attacker. Secondly, we evaluate the performance of the protocol with two types of real life biometric data: face recognition and hand grip pressure pattern. Thirdly, we look at our protocol from the perspective of the users. Our usability analysis shows that our subjects find the SAfE protocol fun to use, and that they would like to have the SAfE pairing available on their mobile devices.

The secure template transfer protocol which uses hand grip biometric appears in [4]. A new application for secure ad-hoc pairing using mobile devices together

with a usability appears in [7]. The extended version which contains experimental validation of real life biometrics is published in [1].

1.4 Conclusions and Outlook

To give a pertinent answer to the main research question on page 8 we refine its scope to the biometric SmartGun application. This is a particular type of application for which a false rejection threat is more harmful than a false acceptance threat. At the time of exploring this problem there were no threat models that could accommodate this new class of biometric application. Therefore, we propose a new design tool, the 3W-tree, which fills this gap. As a result of the 3W-tree analysis of the biometric SmartGun application we identify the two research directions explored in the thesis: the first direction is the secure template storage and the second is the secure template transfer, both of which are vital for the security architecture of the SmartGun application. The scope and application of the results is not restricted, however, to the SmartGun.

Secure template storage is an active research area, which promises user friendly solutions for the problem of interfacing between humans and computers. We explore both theoretical and practical aspects of this problem which is a particular case of the problem of cryptographic key extraction from noisy data. Our contribution is the extension of the theoretical model, to something that can be used in practice. It is an important step forward. However, the research leads to the identification of new, important questions. For example, the security measures used for evaluating different solutions do not make a clear distinction between security and privacy. While security measures are useful from the risk management perspective there are no effective measures for the privacy of the individual user.

Secure template transfer is a particular case of the more generic problem of securely pairing two or more devices when two persons *happen* to meet. Personal devices, which are carried around at all times and the dynamic interactions between the owners of such devices demand solutions which are fast, easy and which do not rely on *any* pre-existing security infrastructure. Classical security solutions require either an on-line connection (Certification Authorities which can assign credentials) or previously shared knowledge (a cryptographic key). These assumptions are not always valid in the dynamic world of today. The problem is to find alternative methods to create security credentials, which are both user-friendly and offer good security. Our solution, the SAfE protocol uses biometrics to create communication keys. The advantage of our solution is its inherent user friendliness and strong security guarantees. However, biometric measurements

are inherently noisy and there is room for improving the error rates of the biometric recognition algorithm.

In this thesis we advance the field of cryptography with noisy data in two ways. The first is from a theoretical perspective by putting forward new definitions, constructions and theorems, which give new insights in the field. The second is from a practical perspective by proposing a new, practical application of cryptography with noisy data in the area of secure, spontaneous interaction.

Chapter 2

Threat Model and 3W-tree

This chapter provides an overview of the architecture of a biometric system and its building blocks. The reader is offered a global view on the main challenges of securing a biometric system. The "standard" architecture of a biometric system is extended by adding components like crypto, audit logging, power, and a representation of the environment to increase the analytic power of the threat model. Our contribution is the 3W-tree, an analysis tool used to identify critical attack scenarios for a biometric system. We use the 3W-tree to analyze the SmartGun biometric recognition system with the purpose of identifying critical security issues. Two important research directions are identified as a result of our 3W-tree analysis. The first research direction is secure storage or biometric templates. Secure storage of biometric templates prevents the compromise of biometric templates when a SmartGun is lost or stolen. The second research direction is secure pairing of mobile devices when no security infrastructure is available. A user friendly secure pairing protocol allows police officers to exchange biometric templates securely when teams are formed spontaneously in the field and as a result of the pairing protocol any member of the team can use all the weapons of the team. We explore these research directions in the following chapters of this thesis.

Currently, new applications of biometrics that have a completely different threat model from classical biometrics are emerging. For example, *Terrorist Watch List* applications and *Smart Gun* applications are characterized by the fact that a *false rejection*, i.e. an authorized user *not* gaining access to the system, could lead to life threatening situations. Terrorist watch lists use facial or fingerprint recognition [17] to identify terrorists. Watch lists are mainly used at airports.

For this application, the main threat is a *false rejection* which means that a potential terrorist on the list is not recognized and allowed to board an aircraft. In this case, a *false acceptance* results in a convenience problem, since legitimate subjects are denied access and their identity needs to be examined more carefully to get access. The biometric Smartgun [85] is a weapon that should fire only when operated by the rightful owner. Such weapons are intended to reduce casualties among police officers whose guns are taken during a struggle. The most promising technology for this application is biometric grip pattern recognition [84]. Again, a *false rejection* is the most serious threat as this would result in a police officer not being able to use the weapon when necessary. Both terrorist watch list and biometric SmartGun are *false rejection* oriented biometric recognition systems. Current threat models for biometric recognition systems are not suitable for false rejection oriented biometric system.

CONTRIBUTION. Categorizing all possible threats on a system results in a threat model which can be used to identify critical attack scenarios. However, as the complexity of the architecture for a biometric recognition system increases so does the complexity of the threat model and its utility for security engineers decrease. We propose a 3W-tree (Who, What, hoW tree) as an analysis tool to identify critical attack scenarios for a biometric system. The 3W-tree is versatile as it can be used to identify both false rejection and false acceptance threats to biometric security systems.

Analysis based on a 3W-tree leads to concrete questions regarding the security of the system. Questions raised by other methods (e.g. attack trees) do not lead to the same level of specific questions. Our method is more concrete than other methods, because we make explicit assumptions about the generic architecture of the system, thus exposing all main components in the architecture that are vulnerable to attack. Our method is not less general than other methods because other architectural assumptions can be plugged in easily. Our method is intended to be used as a design aid.

We apply the 3W-tree design tool to the SmartGun biometric recognition system with the purpose of identifying critical security issues. As a result of our analysis two important research directions are identified. The first research direction is secure storage of template protection. Secure storage of biometric templates prevents a compromise of biometric templates when a SmartGun is lost or stolen. The second research direction is secure pairing of mobile devices when no security infrastructure is available. This demand arises because police officers work in teams that are sometimes formed on an ad-hoc basis. Each officer in the team must be able to fire the weapon of the other officer. A user friendly secure pairing protocol allows police officers to securely exchange biometric templates

when teams are formed spontaneously in the field and as a result any member of the team can use all weapons. We explore these research directions in the following chapters of this thesis.

GENERAL TERMINOLOGY. A *system* is an entity that delivers a *service*. In the case of a biometric recognition system the service is to recognize live measurements compared to identities stored in the database. A *failure* is an event that occurs when the delivered service deviates from the correct service. The deviation from the intended service is called *error*. The hypothesized cause of an error is called a *fault*. Faults can be internal or external to a system. A *vulnerability* is an internal fault that enables an external fault to cause an error. A *threat* can be a fault, an error, or a failure which has both the potentiality aspect (e.g., faults being not yet active, service failures), and a realization aspect (e.g., active fault, error that is present, service failure that occurs). A malicious internal or external fault is an *attack*. The players in the system are users and intruders. A *user* is an entity that receives services from the system while an *intruder* is a malicious entity (machine or human) that attempts to exceed any authority she might have and alter services or alter the system functionality or performance, or access confidential information. *Security* is a composition of the attributes of confidentiality, integrity and availability. The terminology used follows Avizienis, *et. al* [12].

ROAD MAP. *Section* 2.1 gives an overview of the threat models presented in the area of biometric recognition systems. The extended architecture of a biometric recognition system is presented in *Section* 2.2 and the state of the art regarding attacks on components of a biometric recognition system. *Section* 2.3 describes the 3W-tree, the method proposed as a design aid to help identifying relevant attacks for biometric systems. In *Section* 2.4 we apply the 3W-tree analysis to a biometric *SmartGun*, which uses hand grip pattern recognition to identify the owner of a gun before allowing him to fire the weapon. Conclusions are presented in the last section.

2.1 Related Work

In this chapter we review threat modeling and analysis techniques in the area of biometric system security and general security threat analysis techniques focusing on false rejection.

SECURITY OF BIOMETRIC SYSTEMS. Like all security systems, biometric systems are vulnerable to attacks [42, 65]. One specific attack consists of presenting fake inputs such as false fingerprints [83] to a biometric system. To analyze such threats systematically various threat models have been developed. We discuss the most important models: the Biometric Device Protection Profile (BDPP) [22], the Department of Defense & Federal Biometric System Protection Profile for Medium Robustness Environments (DoDPP) [47], the U.S. Government Biometric Verification Mode Protection Profile for Medium Robustness Environments (USGovPP) [23] and Information Technology-Security techniques -A Framework for Evaluation and Testing of Biometric Technology (ITStand) [31]. In the sequel we refer to these three protection profiles and the ITStand simply as "the standards".

Threat Number	**Description**
8.4	Attacker modifies matching threshold.
10.2	Attacker modifies user identifier.
11.2	Attacker cuts power to the system.
13.1	Attacker tampers, modifies, bypasses, or deactivates one or more hardware components.
13.3	Attacker floods one or more hardware components with noise, (e.g. electromagnetic or acoustic energy)
14.1	Attacker tampers, modifies, bypasses, or deactivate one or more software or firmware executables
14.3	A virus (or other malicious software) is introduced into the system.
15.1	Attacker tampers, modifies, bypasses or deactivates one or more connections between components.

Table 2.1: *False Rejection related threats from* ITStand *[31].*

In many ways, the standards are similar. In particular, they do not always make a clear distinction between a threat leading to a *false rejection* and a threat leading to a *false acceptance*. We call these ambiguous threats "catch all" threats. We identify in the standards a total of 48 distinct threats of which only 3 are *false rejection* threats. These are: (1) cutting the power to the system, (2) flooding hardware components with noise and (3) exposing the device to environmental parameters that are outside its operating range. In addition, there are 12 "catch all" threats that cover both *false rejection* and *false acceptance* threats.

It is difficult to compare threats amongst the four standards. For example, BDPP contains one T.TAMPER threat while ITStand contains three tamper related threats: one for hardware tampering another for software or firmware tampering and one for channels. In ITStand tampering and bypassing is mentioned when describing the same threat while BDPP explicitly mentions the T.BYPASS threat. ITStand is the most complete in identifying *false rejection* threats, it identifies the largest number (8) of such rejections (See *Table* 2.1). However, only threat 13.3 is a clear false rejection. All the others are "catch all" threats. There are three tamper related threats: one related to hardware tampering (13.1), one related to software tampering (14.1) and one for channel tampering (15.1).

The threats to biometric recognition systems in the standards are general, not specifying the exact point in the system that is vulnerable, or the circumstances that make the system vulnerable to attack. The method of attack is also not clear, all that is said is that hardware can be tampered with, bypassed or deactivated. These threats lack the exact how and where and thus their practical value is not clear.

SECURITY TAXONOMIES. There are many general security taxonomies in the literature. They classify attacks based on one or more grounds of distinction. Some taxonomies group attacks using similar grounds of distinction, but use different classes. For example, both *Neumann and Parker's SRI Computer Abuse Methods and Models* [60] and *Jayaram and Morse's Network Security Architectures* refers to misuse techniques, [50]. However, the Neumann classification identifies classes like: *external, hardware misuse, masquerading, pest programs, bypasses, active misuse, passive misuse, inactive misuse, indirect misuse* while Jayaram and Morse's taxonomy identifies only 5 different classes i.e. *physical, system weak spots, malignant programs, access rights and communication based*. Other taxonomies view attacks from totally different angles, for example *Anderson's Penetration Matrix* [50] has three types of penetrators: *external, internal and misfeasance* while *Knight's Vulnerability Taxonomy* [50] defines a vulnerability as having five parts (*Fault, Severity, Authentication, Tactic, Consequence*). Each part is defined according to a different taxonomy. None of these classifications pay special attention to biometrics.

We will develop a specific, informative attack classification method that can capture both false rejection threats as well as false acceptance threats.

2.2 Generic Architecture of a Biometric Recognition System

We begin by describing the architecture and the life cycle of a typical biometric recognition system.

For the purpose of biometric recognition one needs (1) a biometric reader which measures the biometric identifier (e.g. a normal camera for face recognition or a recorder for voice recognition), (2) software that converts the recorded information into digital form and compares match points and (3) a database that stores the biometric data for comparison.

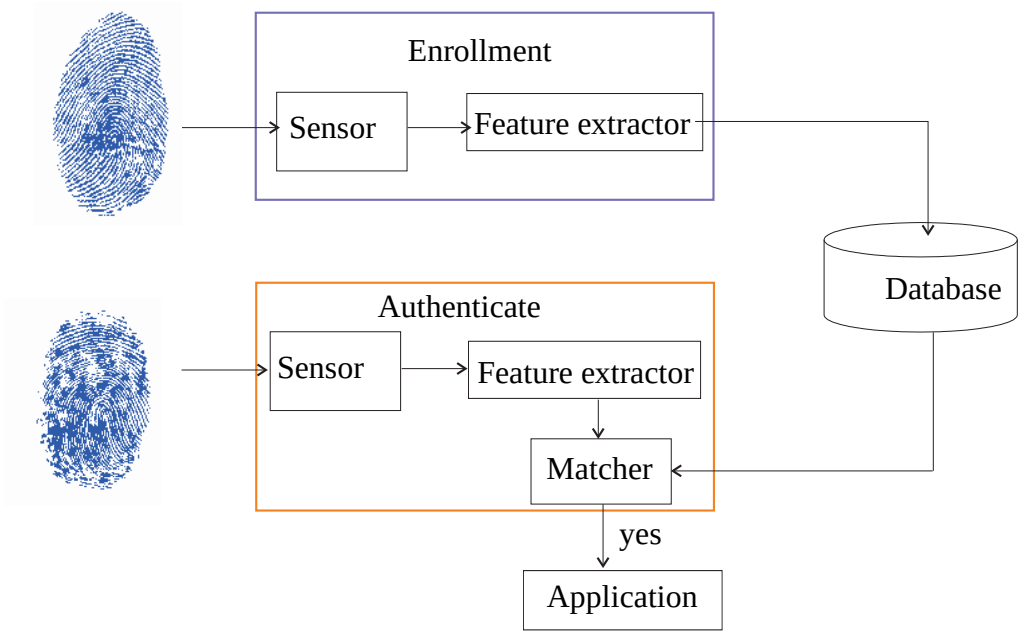


Figure 2.1: *Generic Architecture of a Biometric Recognition System. The arrows indicate communication channels that are used to transfer information between components.*

There are two phases in the lifetime of a biometric system. The first, is the *enrollment* phase when the biometric system learns the identity of a user and stores the relevant data in a databases. The second, is the *authentication* phase when a person is accepted or rejected as being enrolled in the database. *Figure* 2.1 shows a block diagram of a generic biometric recognition system.

In *Figure* 2.1 the *biometric reader* is the interface between the real world and the biometric system; it has to acquire all the necessary data. In many applications this is an image acquisition system, but it can be changed according to the desired biometric modality.

Another component in *Figure* 2.1 is the *feature extractor* which performs all the necessary pre-processing: remove artifacts from the sensor, enhance the input (e.g. by removing background noise), image translation and rotation, normalization, etc. Its most important task is to extract distinguishing features in the biometric identifier recorded by the sensor.

During enrollment several biometric impressions are collected from an individual. A synthesis of all characteristics of these impressions called the *template* is computed and stored in the *database* for each individual.

The *matcher* is used during authentication to compare a feature vector calculated from a live measurement with the template stored during enrollment. Due to noise it is expected that the two will differ. The matcher computes a distance (e.g. Hamming distance or Malahanobis distance, etc.) between the feature vector and the template. If the computed distance is below a pre-established threshold the feature vector and the template are said to come from the same individual, and according to the application the user is allowed to access the resources protected by the biometric system.

2.2.1 Vulnerabilities of a Biometric Recognition System

A biometric system has the potential to solve many of the problems associated with classical authentication systems. However, according to Bolle, *et al.*, [16] biometric systems are not considered much in the security literature and as a consequence there are many open questions on how to make biometric authentication work without creating additional security loopholes. In this section we present an extended architecture of a biometric system, see *Figure* 2.2 which helps to identify potential loopholes. We discuss vulnerabilities related to each component in the architecture as well as vulnerabilities related to the connecting channels.

Components like power or users are not part of a classical architecture of the biometric recognition system, of *Figure* 2.1. We argue that their role is crucial when designing the security architecture. Therefore, we extend the generic biometric architecture to include the following components:

- AUDIT LOG, important actions need to be recorded for later analysis.
- CRYPTOGRAPHY, to ensure the authenticity and integrity of data stored and transmitted on selected channels.
- POWER, is a major concern especially when the biometric device is portable. For example, replacing the power source might restart the application causing the biometric system to enter an unknown or unstable state.

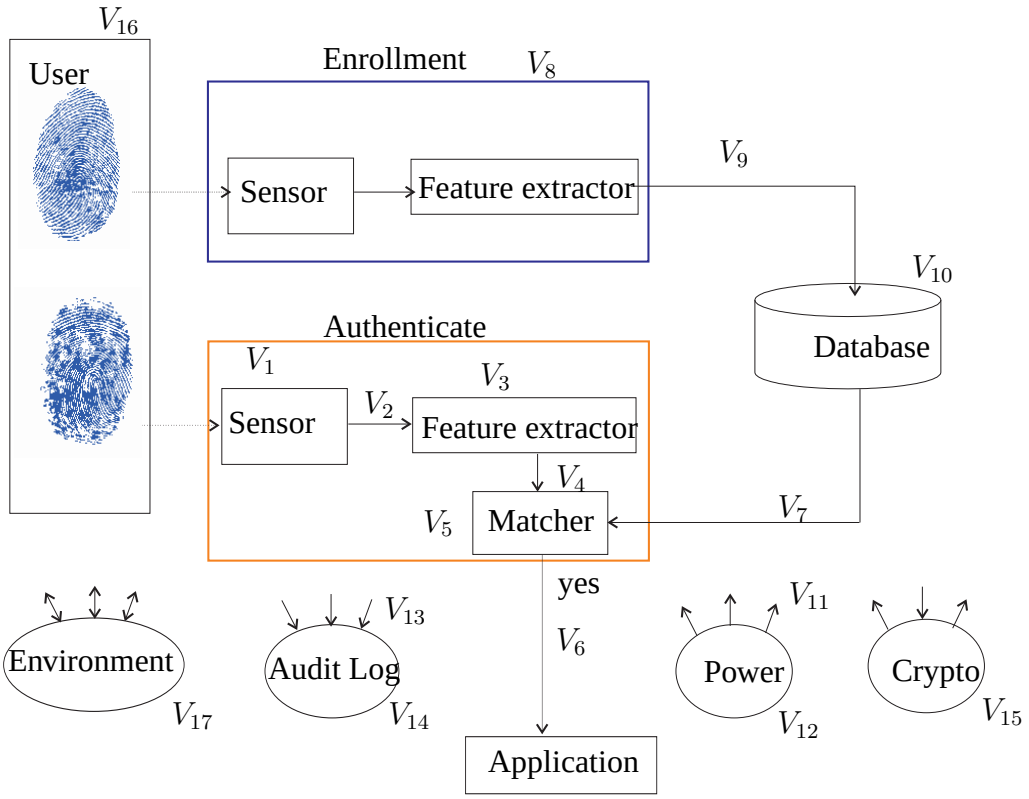


Figure 2.2: *General view of a Biometric Recognition System showing* 17 *vulnerabilities. We identify three classes of channels: the first class represents the internal channels (continuous lines), which are used to transport the information between the modules of the biometric recognition system. Secondly, we have external channels (dotted lines), which are used to input information to the biometric recognition system and thirdly we have implementation dependent channels (multiple arrows), which are present in between components according to the implementation of the biometric recognition system.*

- ENVIRONMENT, biometrics, like other protection mechanisms, are influenced by environmental conditions which can cause surprises. We also include in this category: operating parameters such as temperature, humidity, etc.

- USERS, some biometric systems like speech recognition are vulnerable to alcohol intake and stress [11]. Injuries may affect the accuracy of fingerprint recognition systems while changes in the appearance (different hair cut, mascara, etc) influence the performance of face recognition biometric systems.

The specific vulnerabilities corresponding to the extended architecture, shown

in *Figure* 2.2 are summarized in *Table* 2.2.

Vulnerability	**Description**
V_1	Is a vulnerability of the input device. The most serious threat to a biometric system is presenting a fake biometric [65]. The fabrication of something analogous to a real user is called a *Synthetic Biometric Feature Attack*. This attack can be implemented with or without tampering with the sensor.
V_2	Is the resubmission of a previously stored biometric signal in the channel between the sensor and the template extractor (replay attack/false data inject), or reuse of residuals.
V_3	Is a vulnerability related to the feature extractor. For example at a given time or under some specific conditions a Trojan Horse may *override the feature extraction*.
V_4	Is a vulnerability of the communication channel between the feature extractor and the matcher. For example an attack that exploits this vulnerability will insert a synthesized feature vector into the communication channel.
V_5	Is a vulnerability that allows a *Trojan Horse* attack. This time the target is the matcher, which is forced to produce a match or non-match result.
V_6	Is a vulnerability related to overriding the output of the matcher and thus bypassing the entire authentication process. The output of the matcher module could be forced to be either a match or a non-match.
V_7	Is a vulnerability which allows an attack on the communication between the (central or distributed) database and the authentication system. A template stored in the database is sent to the matcher through a channel. The representation of the template is changed before it reaches the matcher.
V_8	Is a vulnerability of the enrollment center. The enrollment and the authentication process have similarities in the sense that they are both implementations of an authentication protocol, and therefore enrollment is vulnerable to attack points $V_1, \ldots V_6$.
V_9	Is a vulnerability of the channel that links the enrollment center to the database. Control of this channel allows an attacker to override the (biometric) representation that is sent from enrolment to the biometric database.
V_{10}	Is a vulnerability of the database itself. An attack exploiting this vulnerability could result in corrupted templates, denial of service to the person associated with the corrupted template or authorization of a fraudulent individual.

V_{11}	Is a vulnerability of the channel that links the power source to the system. An attack could destroy for example this channel.
V_{12}	Is a vulnerability of the power source of the system, which can be tampered with.
V_{13}	Exploiting this vulnerability allows an attacker to prevent audit records from being recorded.
V_{14}	Is a vulnerability, which successfully exploited, allows audit records to be deleted or modified, thus masking the actions on an attacker.
V_{15}	Is a vulnerability related to the strength of the cryptographic algorithm employed. Security functions may be defeated through cryptanalysis on encrypted data, i.e. compromise of the cryptographic mechanisms.
V_{16}	Is a vulnerability related to the users of the system, which regardless of the role that they play, can compromise the security functions.
V_{17}	Is a vulnerability related to the environmental conditions (temperature, humidity, lighting, etc.). For example the extensive usage can degrade the security function of the system.

Table 2.2: *Vulnerabilities associated to a biometric recognition system by Bolle* et al. *[16]*.

ATTACKS ON THE BIOMETRIC READER. The most serious threat on an input device is presenting a fake biometric [65]. The fabrication of a fake physical biometric is called a *Synthetic Biometric Feature Attack*. It is known that some biometrics are harder to forge: iris [30], retinal scan [43], face thermogram [40] while others are easier to forge: voice print [35], face [78], hand written signature [40] or fingerprint [83].

Since biometric information is not secret the original biometric can be obtained with or without the permission or cooperation of the "owner" of that biometric. In some cases, like fingerprint for example, extensive literature on how to produce fake identifiers is readily available for anyone who cares to try [83, 51].

When cost is not an issue to the attacker, all biometrics can be, and probably will be, the subject of a synthetic feature attack. The difficulty of such an attack depends on the implementation of a specific system [16].

The system must somehow be able to verify that the biometrics came from the person at the time of verification. *Liveness* determination verifies that a biometric sample is coming from a living person [64]. The synthetic feature biometric attack can be implemented as a *coercive*, *impersonation*, or *replay attack* with more or less tampering with the sensor [16].

A *coercive attack* is an attack where the authorized user's biometric data is presented in an illegitimate scenario. For example the attacker physically forces a genuine user to identify herself to an authentication system or after the physical removal from the rightful owner [16]. Designers have to think how to counter such attacks, for example by installing security cameras at ATMs.

An *impersonation attack* involves changing one's appearance so that the measured biometrics match an authorized individual. Examples of biometrics that can be the subject of this kind of attack are face, voice or signature. Multi-modal biometrics reduce the exposure to an impersonation attack (particularly if the system is checking for consistency between the modalities). Broad categories of impersonation threats are identified by Bacon *et al.* [47] as follows: (1) an imposter attempts to defeat the biometric authentication or identification either by a zero-effort forgery attempt, (2) the impostor directs his attacks on some known or suspected weakness and (3) the impostor attempts to subvert the identification or verification process by undermining the integrity of biometric templates.

A *replay attack* involves the re-presentation of previously recorded biometric data. This is the simplest attack possible against a biometric system. For example, take a picture of a person and present it to a face recognition biometric system. Current research tries to eliminate this kind of attack. For example face recognition systems try to detect the three - dimensionality of the face presented to the camera [21].

In case of fingerprint and palm recognition the replay attack can take the form of latent print reactivation. The oily residue from touching the surface of the scanner may leave a latent print that can be copied and reactivated into a valid print.

ATTACKS ON THE FEATURE EXTRACTOR. A *Trojan Horse* attack on the feature vector, produces at a given time or under some specific conditions a pre-selected feature. Care must be taken during the employment of the system to avoid this. Stored templates can be protected by encryption. Data transmitted between the sensor and the rest of the system could also be protected by cryptography. But here, unique session keys would be necessary (e.g. through time-stamping) to prevent data being replayed successfully. If the stolen template is used, then liveness testing could be used to ensure that the biometric is actually being submitted by a person.

Transformations e.g. cryptography can be applied on the feature vector only if the time element is not critical or the equipment can process data fast enough. Template transformation techniques have been developed to circumvent the compromise of a template by the legitimate substitution of the transformed version of the template for matching against a similarly transformed feature vector. This is called in the literature *cancelable biometrics* [64]. This is an intentional, repeat-

able distortion of a biometric signal based on a chosen transform. The biometric signal is distorted in the same fashion at each presentation, that is, during enrollment and for every subsequent authentication. This technique has been developed to protect the privacy of the individual and to permit the reutilization of a biometric sample even after the biometric feature has been stolen.

ATTACKS ON THE MATCHER. Again a *Trojan Horse* attack is possible, this time the target is the matcher, which can be forced to produce a high or low match score and thereby to manipulate the match decision [16].

ATTACKS ON THE STORAGE. Storing unprotected biometric templates in a database introduces a number of security and privacy risks. For example an attacker could steal a template from the database and construct artificial biometrics that pass the biometric authentication. Once compromised, the biometric can not be re-issued, updated or destroyed. Another possible attack is the unauthorized modification of one or more template representations in the database such that a false acceptance is forced. Tracking whether a user is enrolled or not in a certain database could result in exposure of sensitive personal information. Another attack that should be taken into account is *the double enroll attack* which means, as the name suggests, re-enrolling a user under a different name in the database with different privileges. The protection of the database is important because the final authentication system is only as secure as its enrollment database.

CHANNEL ATTACKS. Channels provide the ability to transfer information between input device, feature extractor, matcher and database. The system components that are communicating may be local or remote. Communication can be realized using different channels. *Figure* 2.2 shows that from 17 possible vulnerabilities related to a generic biometric system 5 threats are related to channels. This emphasizes the importance of addressing channel attacks. The *Connectivity assumption*[47] states that biometric templates must be protected during transmission between the biometric subsystems for example by cryptographic means, or by tamper resistant hardware.

POWER ATTACKS. By cutting the power to the system an attacker can make the system fail. Depending on the power source connected to the system batteries or electricity attacks may be different. Restarting the system after a power loss can result in an unstable system.

CRYPTO ATTACKS. Cryptanalysis on encrypted data or brute force attacks may help an attacker gain unauthorized access. If code or data associated with crypto-

graphic functions can be accessed inappropriately by a process or user the cryptographic mechanisms and the data protected by those mechanisms may be viewed, modified, or deleted.

AUDIT LOG COMPROMISE. An audit log compromise is not a direct attack on the system. However, an inadequate collection of audit data with the intention to hide the traces of an attack on the system is dangerous since it prevents attacks on the biometric system from being discovered.

ENVIRONMENTAL AND USER RELATED ATTACKS. A user may cause harm to a system intentionally or unintentionally. For example an administrator may incorrectly install or configure the biometric system, the result being an ineffective mechanism. Non-hostile administrators (unintentionally or under coercion) could incorrectly modify user privileges or the matching threshold or enroll an unauthorized user. Another threat is that an impostor may acquire administrator privileges. An attacker may cause failure of the biometric system by exposing the authentication device to conditions outside its normal operating range. The conditions refer to temperature, humidity, light, etc.

OTHER ATTACKS. In the following we describe other known attacks on biometric systems. These are more complex and involve successful exploitation of one or more vulnerability points in *Figure* 2.2

Hill-Climbing Attack. This attack is described by Bolle et. al [16]. The biometric sample is slightly modified and then submitted to the matching algorithm repeatedly. The output score of the current biometric sample is observed. If the score is greater than the previous output score the changes applied on the biometric sample are preserved. The goal is to achieve the match threshold. This attack can be prevented if repeated trials are not allowed. According to Ulugad *et al.* [78] this type of attack can be cast as an attack in point T2 or point T4 in *Figure* 2.2.

Swamping Attack. This attack tries to exploit weakness in the algorithm to obtain matches of incorrect data. For example for a fingerprint system the attacker might try to submit a print with a lot of minutiae hoping that sufficiently many of them will clear the threshold. The weakness in the algorithm is that it accepts such a representation. [16].

Piggy-back Attack. The attacker tries to gain physical or logical access simultaneously with a legitimate user [16].

CONCLUSIONS. Biometric systems have a lot of weak points. Most likely, attacks occur during the live verification phase. An attack during *enrollment* is less expected, because this operation normally takes place in a secured environment. Therefore attacks made during the *authentication* are most likely, and therefore their effect should be limited.

2.3 3W-tree

The task of the biometric system security architect is to evaluate the effects and likelihood of the attacks described above as well as potential new attacks for a given biometric system, and ultimately to find and implement adequate countermeasures. A crucial step in this evaluation is the identification and classification of vulnerabilities. The result of this step is a classification of all known vulnerabilities in the system, the threats that can exploit them, and attacks that may result. The most adequate taxonomy for evaluating the risks associated with the biometric system has to be selected. This is in our opinion a difficult task, since security taxonomies in the literature do not capture the threats related to biometric systems well. Among the taxonomies studied we could not find one that could give us the assurance that all the relevant threats are indeed identified and which could help in developing the threat model for a biometric recognition system.

During our research for suitable taxonomies we observed that computer security taxonomies themselves can be classified. We propose to use this meta-classification to assist in identifying a proper threat model. Our meta-classification will prove to be useful in choosing, the right taxonomy or if there is no appropriate taxonomy at least provide a guidance to the process of building a new one.

The most general classification that we propose is the division of security taxonomies presented in the literature as: *specific area taxonomies* and *general taxonomies*. *Specific area taxonomies* are developed for restricted domains in computer science. We have found taxonomies for DoS attacks, Unix systems, software bugs, secure devices,[62] etc. *General taxonomies* are applicable in any computer science area. General taxonomies can be further divided into *atomic taxonomies* and *process oriented taxonomies*. *Atomic taxonomies* classify attacks based on one "fundamentum divisionis" or ground of distinction. The main grounds of distinctions used in atomic taxonomies are:

THE WHO. Is used by taxonomies that classify attacks according to various characteristics of the attacker. Anderson's Penetration Matrix, [50] covers the types of penetrators, based on whether they are authorized to use a resource. Abraham *et al.*[50] identify three classes of adversaries relative to the position of the attacker into the system. Rae and Wildman [62] assemble a

structured taxonomy of attacks as a basis for defining the access required by an attacker. In other papers [87] the motivation or the skill required to mount a successful attack, is taken into account.

THE WHAT. A considerable number of taxonomies group attacks on "modus operandi" or attack methods used during an attack. Neumann and Parker, [50] identify 9 distinct procedures of conducting an attack like *external, hardware misuse, masquerading, pest programs, bypasses, active misuse, passive misuse, inactive misuse, indirect misuse*. Lindqvist and Jonsson [50] extend the work of Neumann and Parker, refining the classification allowing the number of classes to increase from 9 to 26. For example the *hardware misuse* class is decomposed in *logical scavenging, eavesdropping, interference, physical attack, physical removal* and *masquerading* is broken down in *impersonation, piggybacking attacks, spoofing attacks and network weaving*. Lindqvist and Jonsson also introduce the concept of the dimension of an attack, which states that for every attack there is a result. Jayaram and Morse's develop a *taxonomy of security threats to networks* and the classes identified are *physical, system weak spots, malign programs, access rights, communication based*. We notice that some classes as *malign programs* overlap *pest programs* in Neumann and Parker's taxonomy, but the *communication* based class is new. Other taxonomies of this type are extensively covered in the PhD thesis of Lough [50], who also lists the similarities between taxonomies.

THE HOW. Taxonomies of this type have as ground of division the exploited vulnerability. Howard's CERT Taxonomy distinguishes three types of vulnerabilities *implementation vulnerability, design vulnerability, configuration vulnerability*. Other taxonomies identify a vulnerability as belonging to one of the following categories of attack: *specification weakness, implementation weakness, brute force attack*. One of the most interesting taxonomies proposed is Knight's Vulnerability Taxonomy [50]. He defines a vulnerability as being a quintuple of the form (*fault, severity, authentication, tactic, consequence*). In 1976 Stanford Research Institute collected 355 security breaching incidents and divided them into 7 violation categories [50]. We note that some *specific area taxonomies* are vulnerability taxonomies like 'Beizer's' bug taxonomy that is a software bug taxonomy [50] or Richardson's DoS taxonomy that classifies *denial of service* attacks according to three different categorizations.

Each atomic taxonomy represents only one dimension of the attack. An attack, however, is rarely caused by a single vulnerability in a system and is rather a

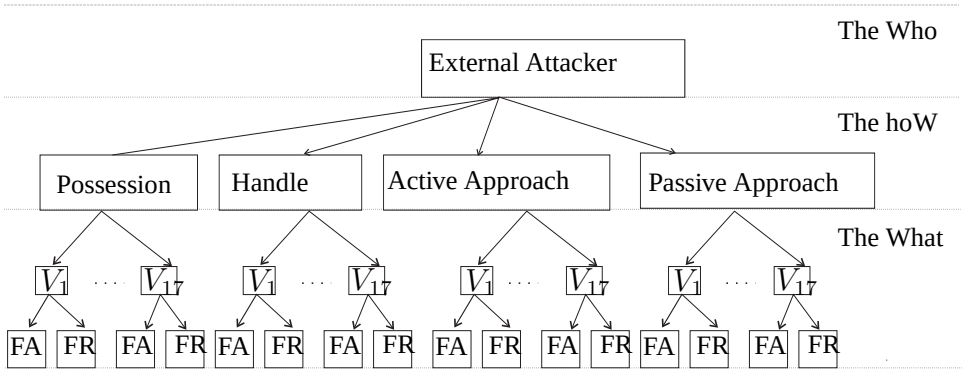


Figure 2.3: *3W-tree of attacks on biometric systems for the external attacker. V1-V17 are points of attack shown in* Figure *2.2 and summarized in* Table *2.2. In a practical analysis the same structure is repeated for the internal attacker.*

function of different characteristics of the system. Three relevant grounds of distinctions are identified in the general security taxonomies in the literature, namely the *who*, the *how* and the *what*. Therefore we propose to use one taxonomy from each of the identified classes (who, how and what). This offers the possibility of identifying a broad range of attacks. Combining a taxonomy from each of these classes creates a nested taxonomy, which we call the *3W-tree* (Who, hoW and What). *Figure* 2.3 shows a 3W-tree built for the generic biometric architecture presented in *Figure* 2.2.

We note that creating a 3W-tree is in conformance with the general methodology of Threat Vulnerability and Risk Assessment (TVRA) [67], which has been designed as a threat, vulnerability and risk assessment method and tool for use whilst writing standards. The purpose of using the TVRA in standardization is to be able to identify vulnerabilities and mitigate the risks and then assess the vulnerabilities that exist in the system with the countermeasures applied. This process has to be applied iteratively, until the risk of unwanted incidents is reduced to an acceptable level. However, TVRA is a general purpose risk assessment method that applies to any security architecture.

A 3W-tree (Who, What, hoW tree) is an analysis tool specifically built for biometric systems, which can be used to identify critical attack scenarios. Analysis based on a 3W-tree leads to concrete questions regarding the security of the system. Questions raised by other methods do not lead to the same level of specific questions. For each ground of distinction (Who, hoW, What) there are several taxonomies one may choose from. In the 3W-tree presented below for each level we chose one particular security taxonomy that fits the biometric system we analyze in *Section* 2.4. We stress that our 3W-tree is flexible and supports any other taxonomy a security architect may find more appropriate for the particulari-

ties of another biometric system. Our goal is to identify as many relevant attacks as possible, from all relevant points of view while maintaining a comprehensive structure (the 3W-tree).

Compared to atomic taxonomies, *process oriented taxonomies* take one step further and view an attack as a process. We see the extension of 3W-tree to cover processes(combination of multiple vulnerabilities) as future work.

2.3.1 The Who taxonomy in 3W-tree

The first level of the 3W-tree is a classical *who* taxonomy from the attacker's position relative to the system [60]. In this taxonomy, attackers are divided into three classes.

Class I attackers, or *external* attackers, lack knowledge about the system and have moderately sophisticated equipment.

Class II attackers, or *internal* attackers, are knowledgeable insiders, who are highly educated and have access to most parts of the system.

Class III attackers are funded organizations with ample resources that are able to assemble teams and design sophisticated attacks. The general opinion is that a system is considered secure if it can withstand class I and class II attackers. It is widely acknowledged that there is no protection against class III attackers and we also do not consider them.

2.3.2 The hoW taxonomy in 3W-tree

The second level is a *how* taxonomy described first by Rae, *et al.*, [62] as a taxonomy for secure devices. The actions allowed for an attacker in this taxonomy are described below.

When the attacker *possesses* the device she can open it and break tamper evident seals with impunity. She may try different attacks, learn how the system works. For example the attacker may buy a biometric system identical to the one she intends to attack.

An attacker may *handle* the device physically, but cannot break tamper evident seals on the device. For example, she has access to the device for a limited amount of time, or under the watchful eye of the owner of the device.

In an *active approach* the attacker may interfere with the device (e.g. over a network) and transmit data to the device from either an insecure or a secure domain.

In a *passive approach* the attacker may be in the proximity of the device, but she cannot touch it. She may be viewed as eavesdropper.

The classes presented are related to one another. *Possessing* the device means that the attacker can *handle* the device and may *approach* the device. This relationship can be formalized as:

$$\textit{passive approach} \subseteq \textit{active approach} \subseteq \textit{handle} \subseteq \textit{possession}$$

An observation is that portable biometric devices are more likely to be attacked in *possession* and *handle* situations so there must be some methods to ensure the physical integrity and robustness of such devices. Fixed biometric devices are more likely to be attacked by means of a *passive approach* and an *active approach*.

2.3.3 The What taxonomy in 3W-tree

The third level of the 3W-tree, the *what*, deals with the threats a system might be subject to, which in case of a biometric system, is either a *false acceptance* or a *false rejection*.

A 3W-tree is useful for identifying attacks on a general biometric recognition system in the design phase, which allows to classify known attacks and to identify the possibility of new attacks in a systematic manner. This is the subject of the *Section* 2.4.

2.3.4 Threat Evaluation

The construction of a 3W-tree for a particular biometric system is the first step to determine the effects and likelihood of an existing threat. We use *attack scenarios* to describe and document each identified threat in the biometric security architecture. The attributes of an attack scenario as detailed in *Table* 2.3 allow the construction of a *risk assessment matrix* which assists the security architect in taking decisions regarding critical attack scenarios. An attack scenario is a path in the 3W-tree of *Figure* 2.3, named $xiyz$ where:

1. $x \in I, E$ where I stands for internal attacker and E stands for external attacker.

2. $y \in \{PA, AA, HA, PO\}$, PA stands for *passive approach*, AA stands for *active approach*, HA stands for *handle* and PO for *possession*.

3. $i \in \{1..17\}$, indicates vulnerability V_i, see *Table* 2.2.

4. $z \in \{A, R\}$, where A means an attack leading to a *false acceptance* attack and R means an attack leading to a *false rejection* attack.

I.Description	
Scenario:	Name of the evaluated scenario.
Tactics:	Describe a possibility to realize this attack
Name:	Name of the attack as it is known in the literature or a link to a paper that describes this attack (if known)
II. Evaluation	
Damage:	The estimated consequence of the attack for the device. The possibilities are: *minor*, *moderate*, *major*. An attack with minor consequences will temporarily damage the device. A moderate consequence attack will temporarily damage the device but it needs specialized personnel to repair it. An attack with major consequence will completely ruin the device, and the whole or parts of it need to be replaced.
Knowledge:	Lists the knowledge that an intruder must have to launch the attack. The categories are: common sense, high school education, expert.
Occurrence:	an educated guess of the probability that such an attack occurs. The estimators are: *low* (unlikely to happen), *medium* (it might happen), *high* (likely to happen)
III. Defense	
Countermeasures	some notes on how this attack might be prevented, or how at least to mitigate its consequence.

Table 2.3: *Detailed description of an attack scenario.*

Each path in the tree corresponds to an attack that has to be evaluated. For example, scenario IPO1A identifies the following: an internal attacker (denoted by the letter I) in the possession situation (denoted by the letters PO), vulnerability point V_1 (presenting a fake biometric/tampering with the sensor) to obtain a false acceptance (A).

To describe and evaluate scenarios we use three classes of attributes. The first class of attributes (denoted with I, Description) is a description of what is known about this attack in the literature. The second class of attributes (denoted with II, Evaluation) asses the impact, likelihood and skills required to realize an attack. The third class of attributes (denoted with III, Defense) describes possible countermeasures, which is particularly useful if there is more than one person, with

different backgrounds and knowledge, which evaluate the security architecture. When described in detail, it also provides a useful indicator of how much it costs to implement the countermeasures.

2.3.5 Attacks trees and the 3W-tree

Attack trees offer a method of analyzing attacks [58, 69]. The root of the tree is identified with the goal of compromising a system and the leaf nodes are ways of achieving that goal. The goals of the children of a node could be the compromise of a sub-system or a contribution thereof. There are two types of nodes: the goal of an *and*-node depends on the goals of all its children, and the goal of the *or*-node depends on at least one of the children [58]. There are commercial tools to support analysis working with attack trees; for example the Secur*IT*ree tool from `http://www.amenaza.com/`.

The main advantage of attack trees is that they help the designer to visualize possible attack scenarios and understand the different ways in which a system can be attacked. If there are many possible attacks, or if there are many components that are subject to attack, an attack tree may become large. In this case the visualization is ineffective. However, by attacker profile based pruning [66], support tools allow the designer to focus on attacks relevant to specific attacker profiles. Another useful feature of the tools is that while constructing a tree the designer can document the changes and also the reason for changes made by annotating nodes.

The main disadvantage of attack trees is that they provide only the choice between and-/or-nodes. This only provides a low level way of breaking up a goal up into sub-goals. The general recommendation is to think hard, which does not provide much guidance.

We propose to combine attack trees with 3W-trees. At the top level, the 3W-tree gives rise to concrete questions about the what, how and whom of an attack. To answer the question, an attack tree can be attached to each leaf of the 3W-tree. By constructing the attack tree for each leaf, the analyst is encouraged to answer specific, focused questions. In the detailed description of an 3W-tree in *Table* 2.3 an attack tree can be placed in the Tactics attribute.

As a conclusion 3W-tree offers an effective method to identify threats related to biometric systems. Once a threat is identified, one can make use of the attack tree method to find tactics of how the goal can be achieved by an attacker.

Figure 2.4: *SmartGun prototype, with a pieso-resistive pressure sensor on the grip.*

2.4 3W-tree Analysis of Biometric SmartGun

In this section we create a 3W-tree to identify the relevant security issues for a biometric SmartGun which is a biometric device intended to reduce casualties among police officers whose guns are taken during a struggle. In the context of the SmartGun research, our main concern is a *false rejection* as this would result in a police officer not being able to use his gun when necessary. A *false acceptance* attack would permit other persons than the owner of the gun to use it.

2.4.1 The Biometric SmartGun

People depend on police officers to protect their lives and property. Police pursue and apprehend individuals who break the law. Armed police officers are a common sight in many countries. They are trained to use their gun only in critical situations. Unfortunately it happens regularly that during a struggle the suspect captures the weapon from the police officer and shoots him. Research in the United States has shown that every year approximately 16% of police officers killed in the line of duty were shot with their own gun.

A smart gun is a firearm which can only be operated by its authorized owner. As such, the gun would be useless in the hands of anyone else who might intend to misuse the weapon. We propose to use biometric recognition to make a gun smart.

The grip of our SmartGun (see *Figure* 2.4) is covered with a grid of pressure

sensors that are protected against wear and tear. These sensors are capable of measuring the static pressure pattern as a function of place when the gun is being held (representing the positions and shapes of the fingers on the grip and the pressure exerted by them) and the dynamic pressure pattern (i.e. the pressure variation) as a function of place and time when the grip tightens prior to pulling the trigger.

During the enrollment phase, a template is securely stored in the weapon (and also in a central database operated by the police). The template could be a set of constraints that the measurements have to satisfy or it can be a prototype grip pattern to which the measured grip pattern must be sufficiently similar.

The template is compared to the measured grip pattern when the weapon is handled. If the stored template matches the current sensor measurements sufficiently well, the weapon is enabled, otherwise it remains disabled. Therefore, only the authorized user may fire the gun, and not someone who has taken the gun away from the authorized user. The type of biometric recognition for this application is verification. In domestic settings, one could also program the gun to reject certain people, i.e. children.

The biometric verification in the above application is transparent. By holding the gun, the user implicitly claims that he is authorized to use it. The biometric data is also presented implicitly, when the grip is settled. In this example the transparency contributes to safety and user convenience. This transparency contributes to safety, when immediate recognition is required in case of an emergency. Transparency also makes grip pattern biometrics convenient to use.

After discussions with police officers we realized that the biometric SmartGun must have the size and feel of the guns currently in use. The reason is that a gun with a different weight or size is hard to adapt to: it requires additional training to regain the same accuracy as with the old weapon. Also, police officers often work in small teams, and each team member should be able to handle the weapon of the other team members, for example when one of the weapons is lost, or fails.

An important requirement is a very low false-rejection rate, rendering it highly unlikely that the authorized user will not be able to fire the weapon. The current official requirement in the Netherlands is that a police gun must have a probability of failure less than 10^{-4}. The overall failure rate of the take-away protection must, therefore, remain below this value. The recognition must work correctly for right-handed and left-handed use, with and without gloves. Even if the false-acceptance rate is as high as 50%, this would make 50% of the take-away situations harmless, which is a significant improvement over the current situation where each take away may be fatal.

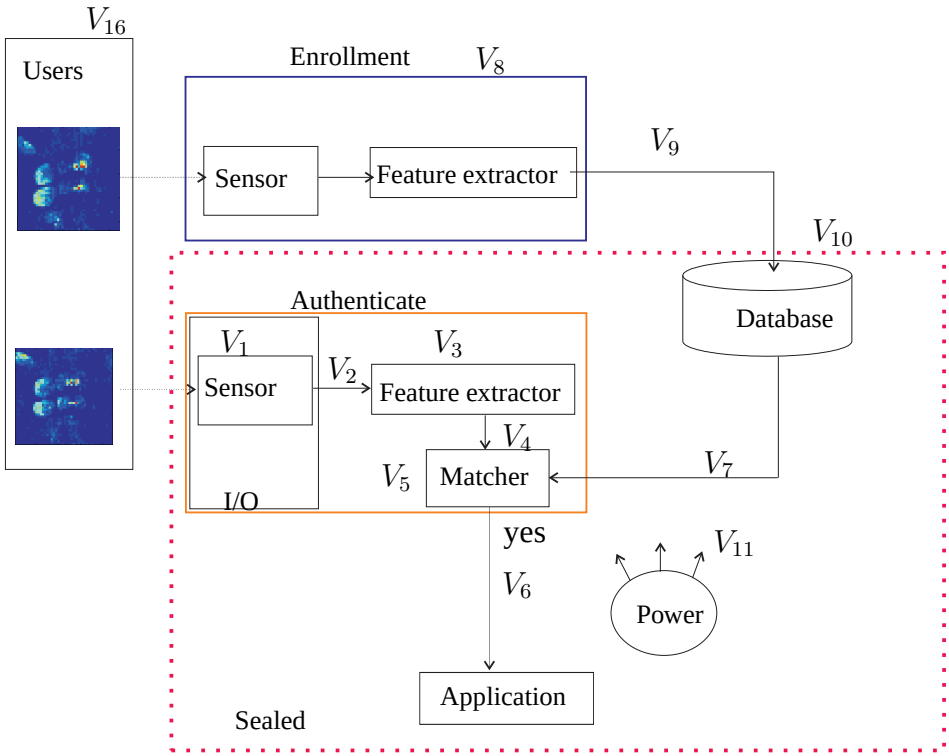


Figure 2.5: *Security Architecture of the SmartGun prototype. The dotted line shows the components that are inside the gun and which should be protected by a tamper evident seal. In the case of the SmartGun the matcher runs inside the processor and in the case of a favorable decision enables the gun. Due to the tamper evident seal the Environment has no significant influence on the components in the gun. For the same reason the Crypto component was removed from the architecture. The existence of the Audit Log component is still to be decided, thus at this point in time we removed this component from the architecture.*

2.4.2 3W-tree Analysis

In relation to the generic biometric architecture in *Figure* 2.2, in this section we present the SmartGun architecture and describe vulnerabilities. Next, we present a set of assumptions related to the intended use of the SmartGun, which was developed in close cooperation with the KLPD (Korps Landelijke Politie Dienst) and help of the 3W-tree analysis. We conclude this section with a set of research question and conclusions.

2.4.2.1 SmartGun Architecture

The SmartGun security architecture as shown in *Figure* 2.5 is a specialized version of the generic architecture of *Figure* 2.1. From a conceptual perspective the SmartGun verification system, as any biometric recognition system, can be subdivided into the following blocks: Sensor, Feature Extraction, Processor (Matcher), Memory (Template), Gun Control and Power.

The sensor that is used for this project is a custom design piezo-resistive pressure sensor. It is available in a size that fits the prototype gun but, which is that of a Walther P5, see *Figure* 2.4. This sensor consists of two layers of strong and flexible polyester foil. Each layer has 44 silver electrode strips deposited on one side. One layer has vertical and the other horizontal strips. A piezo-resistive ink is printed on top of the silver leads. This construction results in a network of silver strips with a resistive element at each crossing. The entire sensor array can be modeled as a 44×44 network of variable resistors. The resistive elements are sensitive to pressure. The grip pattern is measured by determining each resistor value. This is done by subsequently connecting the horizontal and vertical conductors to an analog measuring circuit. The connections can be altered by multiplexers controlled by digital logic.

The functionality and the vulnerabilities associated with the other components (Feature Extractor, Matcher, Storage, etc.) in *Figure* 2.5 are described in detail in *Section* 2.1.

2.4.2.2 Assumptions made for the 3W-tree Analysis

Assumptions create the general environment for describing attack scenarios. Before analyzing the threats our system is subject to, we make some realistic assumptions about the intended use of the SmartGun. After extensive discussion with the KLPD, the intended users of the SmartGun, we developed a list of assumptions which describe the environment in which the biometric recognition system will be used. The assumptions are important for motivating the presented threat model and they regard the operating environment including physical, personnel and connectivity issues.

1. Assumptions related to users:

   (a) (ADMIN.) Administrators of the system (police officers) are assumed to be non-hostile and trusted to perform their duty in a competent manner.

   (b) (TRAINING.) Getting used to the smart gun should not imply additional training for the police officers. However, it may take some time

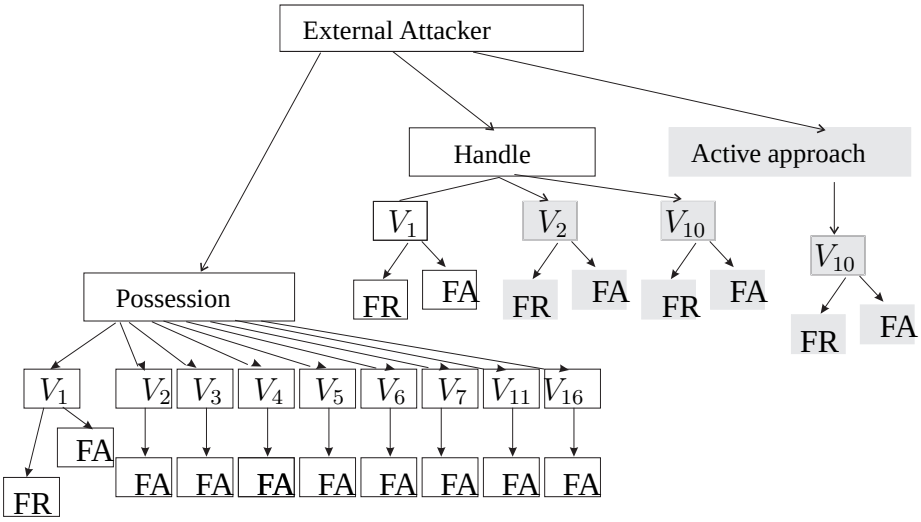


Figure 2.6: *3W-tree of attack scenarios relevant to the SmartGun biometric system. Components in grey represent threats that can not be analyzed at this stage. For example since we do not know how the template is transferred into the gun, a necessary step for the gun to recognize its owner, we leave its analysis for later. Details on the assumptions related to the SmartGun architecture are presented in subsection 2.4.2.2.*

to get used to the routine of using an electronic gun, like keeping batteries charged, auditing, training and template exchange to match team composition. This period should be as short as possible.

2. Assumptions related to the gun construction:

   (a) (SEAL.) Tampering with seal(s) on the gun, which secure the feature extractor, matcher and all the communication channels should be easy to detect and re-sealing should be hard to do.

   (b) (WEIGHT.) A new gun with a different weight is hard to adapt to: it requires additional training to regain the same accuracy as with the old weapon. It is desirable to approach the weight of the Walther P5 as much as possible so additional training is unnecessary.

   (c) (AUTHORIZATION.) The Walther P5 compatibility requirement stating that the gun must be safe while carrying a bullet in the chamber implies that the time needed between the choice to use the gun and the actual firing, must be as short as possible. Thus the time needed for authorizing someone must be as short as possible.

   (d) (TRANSPARENCY.) The mechanism is fully automated, i.e. the user does not need to perform additional actions to receive authorization.

I.Description	
Scenario:	EPO1R.
Tactics:	Jam or break the sensor.
Name:	Unknown.
II. Evaluation	
Damage:	Major
Knowledge:	High school. Which part of the gun is the sensor how to damage the sensor is easy to determine.
Occurrence:	High. It is easy to damage something that is not protected by a seal and is not tamper resistant.
III. Defense	
Countermeasures	The gun architecture should ensure that tampering with the sensor is obvious.

Table 2.4: *Detailed description of attack scenario EP01R.*

(e) (CONNECTIVITY.) The gun should be equipped with I/O interfaces for data. The sensor for reading the biometric grip pattern is an input interface. The nature of the output or input/output is to be decided by experience.

3. Assumptions related to algorithms:

   (a) (RELIABILITY.) It is "smart" (or "personalized") enough to identify and fire only if desired by the rightful owners and/or other authorized users.

   (b) (FAILURE.) Additional technology may not increase the failure rate of the current weapon, being once every 10.000 times. This includes the False Rejection Rate of the authentication procedure and the mechanical failure rate.

   (c) (AD-HOC SECURE INTERACTION.) The gun is able to recognize all members of a police patrol.

   (d) (ROBUSTNESS) The gun can identify authorized users when being in different states, from normal to stressed or even panicked.

4. Assumptions related to the system implementation:

   (a) (ENVIRONMENT.) The enrollment procedure takes place in a secure environment (the police station).

I.Description:	
Scenario:	EPO2A.
Tactics:	The attacker records a correct biometric signal and then injects the signal just before using the gun.
Name:	Replay attack.
II. Evaluation	
Damage:	Moderate
Knowledge:	Expert. Measure, record and store the biometric of an authorized user. If the attacker records the raw biometric then he must know the algorithm that produces the feature vector, the format of the feature vector and because the number of elements of the feature vector depends on the number of user registered to the system, he also has to know this number. He also needs to figure out a way of injecting the signal just before the gun is fired.
Occurrence:	Low. The attack requires inside knowledge of the system, the number of enrollees, and technical skills: recording the biometric or injecting the electronic signal
III. Defense	
Countermeasures	Encrypted communication, challenge-response protocol [64], perfect-matching checking [10], etc.

Table 2.5: *Detailed description for attack scenario EPO2A.*

Each of these assumption is motivated by the strict procedures to which police officers work. Other assumptions may be added as a result of the analysis of attack scenarios.

2.4.2.3 Risk Assessment for the Biometric SmartGun

The assumptions above simplify the 3W-tree analysis and lead to three important observations. Firstly, in the Who taxonomy or the first layer of the 3W-tree, the (ADMIN.) assumption (1(a) in the list above) allows us to restrict the threat analysis to the external attacker. We also assume the manufacturer to be trustworthy and that the biometric devices are certified by a certification authority.

Secondly, at this point in time we do not know what information (if any) is

transmitted on the wireless link, see *Figure* 2.5. Since this is the main point of attack for the external attacker in both *passive approach* and *active approach*, in the second layer of the 3W-tree at this stage we cannot analyze them. We remind the reader that 3W-tree analysis should be an iterative process and once new information is available or there are changes in the environmental condition the 3W-tree analysis should be revised, see *Chapter* 5.

Thirdly, a tamper resistant seal mounted on the gun handle (assumption 2(a) (SEAL.)) makes a false rejection attack in *posses* situation hard. We assume it is difficult for the attacker to disassemble the device, tamper with mechanical parts, re-seal the gun and then return the gun to the police officer without the tampering being noticed.

As a result of the above assumptions, from a total of 96 attack scenarios for the external attacker (12 vulnerabilities $\times$ 4 approach modalities (posses, handle, etc) $\times$ 2 goals (false acceptance, false rejection)) we are left with only 12 most relevant attack scenarios. *Figure* 2.6 shows the 3W-tree for the biometric SmartGun. Threats shown in grey, have to be dealt with once more information becomes available. The practical realization (sealed or not) of the physical link between the sensor and the feature vector (vulnerability V_2) determines the risks associated to an external attacker in handle situation. For this reason, in *Figure* 2.6 threats which we consider as relevant, but which cannot be evaluated at this stage are represented in grey. *Table* 2.4 and *Table* 2.5 list, as example attack scenarios EPO1R and EPO2A. Description for the other 11 threats can be found in the technical report [9].

Once the consequence of the damage and the frequency of occurrence have been estimated the final step is the risk assessment. The risk assessment matrix in *Table* 2.6 offers a means to categorize the risk associated with each attack scenario. We distinguish between three levels of risk. Level one risk is undesirable and requires immediate corrective action (any risk with major consequence and high frequency), level two risk is undesirable and requires corrective action but some management discretion is allowed (any risk with high/moderate damage and medium/high frequency) and level three risk is acceptable with review by management (any risk that produces minor damage and has low probability of occurrence).

The main conclusions of the risk analysis are: there is no level one risk corresponding to an external attacker relative to the SmartGun. There are four attack scenarios with level two risks which correspond to (1) scenario EPO5A, where an attacker in possession situation controls the memory of the gun, thus he has read and write access, (2) scenario EPO11A, where by emptying the battery, the gun can be forced to enter an unstable state, maybe allowing the gun to operate with-

Occurrence	Damage		
	Major	Moderate	Minor
High		EHA1A;EPO1A;	
Medium	EP05A;	EPO11A;	EPO16A;
Low	EPO7A;	EPO2A;EPO3A;EPO4A;EPO6A;	EHA1R;EPO1R;

Table 2.6: *Risk Analysis Matrix for the SmartGun for the scenarios in* Figure *2.6.*

out biometric recognition, (3) scenarios EPO2A and EPO1A where an attacker in handle or possess situation can be accepted as an authorized user of the SmartGun.

The conclusions above are preliminary. In the *Section* 2.4.2.4 we discuss several open questions regarding the architecture of the SmartGun. The choices between the alternative solutions of the open questions have a definite influence on the specialization of the architecture presented in *Figure* 2.5. We recommend a new 3W-treeanalysis of the SmartGun architecture once the open questions are answered, to refine the preliminary conclusions.

2.4.2.4 Open Questions

We discuss several open questions regarding the security architecture of the SmartGun. For each question several alternative solutions exist, in which case only experience can help chose the right one.

ENERGY CAPACITY. *How long should the power system be able to energize the electromechanical components before recharging?*

A large energy capacity makes the device more mobile. A recharge station can be centralized, at a police station for example. The downside of a large energy capacity is the size of the battery, which requires precious space. A choice can be made on whether to decrease the number of bullets, saving space for components like the battery. Another option is to implement the possibility to change batteries while on duty, meaning that the officer carries spare batteries. This would imply a reliable energy level indication, so that the officer is warned in time that the battery needs to be recharged or replaced.

Having a smaller energy capacity battery means more frequent recharging and smaller size. The necessity of frequent recharging could be combined with regular information exchange between the weapon and the supporting network.

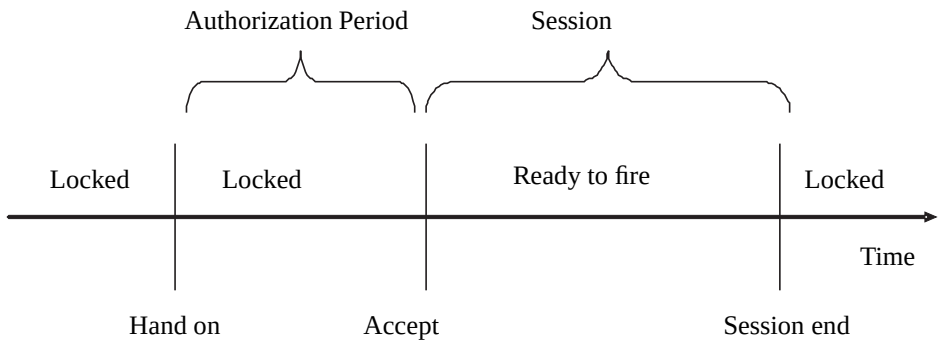


Figure 2.7: *Successful Authorization time line.*

CONNECTIVITY. *Should the information stored on the pistols be accessible remotely or not?*

Remote access provides more mobile smart guns. Standard maintenance and update operations can be done from a distance, making it unrestricted to a central location, such as a terminal. The disadvantage is that a hacker can use the wireless link too, if the communication is insufficiently protected.

Remote access also implies putting a radio in the gun, requiring space and putting an additional load on the limited energy supply.

Remote access requires a more complex communication protocol that guarantees security, reliability and atomicity for template updates. This places an extra burden on the computational resources of the mobile terminal. Extensive information on different forms of mobile communications and ad-hoc networks can be found in standard textbooks such as Rappaport [63].

AUTHORIZING. *A*fter successful authorization, when and how should the session end?

Before we discuss the dilemma implied by this question, we need to define the terms locked (the state of the weapon when it cannot be fired) and ready to fire (the state of the weapon when it can be fired). We also need to define the terms authorized user (a user who has his biometric identity stored in the gun) and impostor (a user who does not have his biometric identity stored in the gun and is fraudulently trying to use the weapon).

The default state of a gun is locked. Before an authorized user can use the gun an authorization must take place. This is a classifier based comparison between the stored biometric identity and the freshly measured biometric impression. A person can fire a gun if the two match. This period is called the authorization pe-

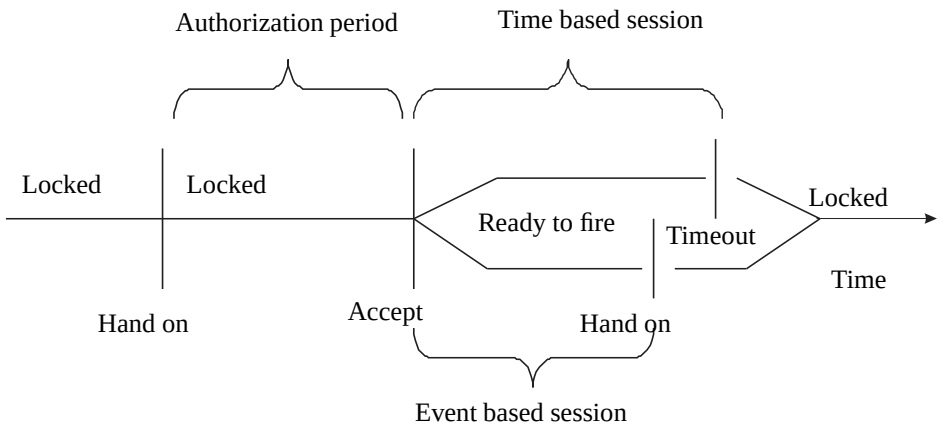


Figure 2.8: *Event and Timeout driven Session Ending.*

riod. A session is the period of time after a successful authorization period during which the gun is ready to fire.

A long session makes it possible that the gun can be used by another (unauthorized) user, for example when an accepted user is deprived of his gun. This safety gap clearly threatens the advantages of ”smart” guns. On the other hand, a short session length requires the gun to check the biometric pattern on a more frequent basis, which takes resources and distracts the officer.

Rather than choosing a time constraint for ending the session, it can also be event driven. Ending the session when the hand is no longer on the gun implies that the gun can only be fired when the authorized user has it in his hands, which has obvious advantages. Also, by choosing an event driven mechanism, there is no need for a clock on the gun, thus saving space. Even better is when both a time and event driven session end can work in conjunction, locking the gun when the hand is off the gun or when the gun is unlocked for a long period of time. This is represented by the following question:

After rejection of the supplied pattern, how long should the gun wait before another sample is taken and examined?

We define the term *cool down* as the minimum amount of time required between two consecutive authorizations. A short cool down period favors the officer in case he has been rejected. However, an imposter can perform several trials in a shorter time frame, so a longer cool down period works against an impostor. Additionally, a longer cool down saves resources due to less frequent authorizations.

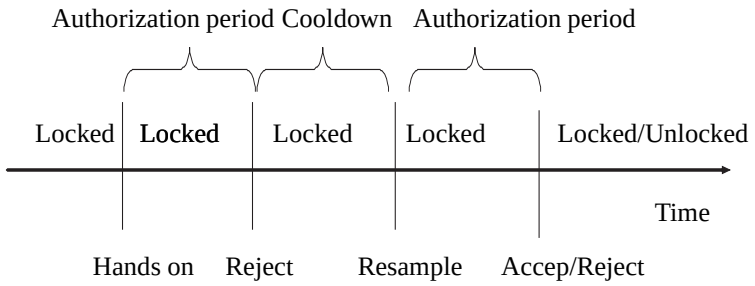


Figure 2.9: *Authorization cool down after rejection.*

POWER FAILURE. *If the power supply fails, should the gun be unlocked by default or locked and give a warning? If the electronic part of the gun fails, a choice has to be made on whether the gun should be locked or not.*

A police officer must be able to fire his gun even if the battery is empty. On the other hand, a fail safe mechanism that allows the gun to be fired even when the battery is empty can be used to bypass the biometric recognition phase. Illegal gun traders would find benefit in methods that disable the electronics, for example emptying the energy supply or frying sensitive circuits with electro magnetic pulses. Thus a failsafe mechanism comes at a price.

In the above we presented several open questions regarding the practical realization of the architecture of the biometric SmartGun. Only experience which of the alternative solution is best.

2.5 Conclusion

In this chapter we introduce the 3W-tree as a tool for threat analysis related to biometric systems and we use it for the analysis of the SmartGun. A SmartGun is a typical example of a biometric system, which is intended to replace the classical mechanical gun with a weapon that can authenticate the rightful owner.

We discussed the fundamental properties of a mechanical gun with police officers from KLPD. As a result of these discussions and the preliminary 3W-tree analysis associated to an external attacker we propose a set of recommendations for the security architecture of the SmartGun.

- LOW FALSE REJECTION RATE. One of the research challenges is that the False Rejection Rate (FRR) must be less than 10^{-4}, which is the accepted rate of misfire for a police weapon.
- LOW FALSE ACCEPTANCE RATE. Although not as critical as the FRR,

the SmartGun should have a low false accept rate (FAR). It is considered acceptable that once in ten trials the SmartGun may accept another user as the owner of the gun. This would reduce police casualties by 90%.

- SECURE SEAL. An attacker handling the gun should not be able to access components inside the gun. There should be a tamper evident seal on the gun handle and electronics. We note that the (SEAL) assumption only states that once the seal on the gun handle is broken, it should be difficult to restore it.

- ROBUST SENSOR. The sensor should be resistant enough to withstand an attacker who is trying to break it. If she succeeds it should be obvious for the police officer that the sensor is compromised.

- SECURE TEMPLATE STORAGE. It should not be possible to reconstruct the grip pattern from the template stored in the gun.

- SECURE TEMPLATE TRANSFER. Police officers work in teams. Each officer in the team must be able to fire the other officers weapon. Normally, teams are scheduled in advance so that appropriate templates can be loaded into the weapons at the police station. However, in emergencies this is not possible. In this case police officers have to team up unprepared and swap templates in the field. Police officers may work with colleagues from other departments, even from neighboring countries, so we may not assume a common key, or even a public key infrastructure. We present below a list of requirements that the secure template transfer protocol must possess to solve the above problem:

  - The initialization of the protocol should not require any special button on the gun handle;
  - Police officers should be able to perform the protocol without any available security infrastructure (shared keys, links to trusted third parties, etc.).
  - The protocol should be fast and easy;
  - Before loading a new template in the gun authorization of the gun owner is required;

The 3W-tree analysis should be an iterative process and when new information is available we recommend a new 3W-tree analysis. In the list above the first two items LOW FALSE REJECTION and LOW FALSE ACCEPTANCE depend on the performance of the biometric recognition algorithm which are the subject of the

forthcoming thesis of Xiaoxing Shang [71]. The SECURE SEAL and ROBUST SENSOR can be considered as engineering challenges which can be addressed in a follow up project. We consider SECURE TEMPLATE STORAGE in *Chapters* 3, 4 and SECURE TEMPLATE TRANSFER in *Chapter* 5 as new research directions and we concentrate on addressing these two challenges.

Chapter 3

Fuzzy Extractors for Continuous Distributions

The use of biometric features as key material in security protocols has often been suggested to avoid memorizing long passwords or keys. However, the use of biometrics in cryptography does not come without problems. It is known that biometric information lacks uniformity, and that biometric information is not exactly reproducible. This is the opposite of what is considered suitable for a cryptographic key. Fuzzy extractors allow cryptographic keys to be generated from noisy, non-uniform biometric data. They can be used to authenticate a user to a server without storing her biometric data directly. This is important because the server may not be trusted.

We show that a relation exists between the entropy of the keys extracted from biometric data and the quality of the biometric data. This information can be used a-priori to evaluate the potential of the biometric data in the context of a specific cryptographic application. We model the biometric data as a continuous distribution and we give a new definition for the fuzzy extractor that is suitable for this type of data. We propose a new construction called *cs*-fuzzy extractor which represents an extension to the classical fuzzy extractor to continuous source (*cs*) data. We apply the new definition to three schemes proposed in the literature for the protection of biometric templates.

Unprotected storage of biometric information is an example of a serious threat for the privacy of users because a biometric template may reveal sensitive personal information [80].

A fingerprint for example can be reconstructed from a stored biometric template as shown by Matsumoto, *et al.* [51]. Another privacy threat is tracking users across multiple databases. The usual solution of using different passwords in different systems does not apply for obvious reasons - a person only has a limited number of biometric identification available: ten fingers, two eyes, etc. Once a biometric template is compromised it cannot be re-issued.

Biometric template protection aims to protect the stored biometric identity of a user from abuse in two ways. Firstly, a protected template reveals almost nothing about the biometrics and if a database with protected biometric templates is compromised, the attacker cannot learn much about the biometric template. Secondly, if such an intrusion is detected the biometric template is not lost, since at any time the protection scheme can be reapplied on the original data.

The main challenge in protecting biometric templates using cryptographic techniques is coping with noise, which is always introduced into biometric samples during data acquisition and processing. Biometric template protection schemes can transform a noisy, non-uniform biometric template represented as a sequence of real numbers into a reproducible, uniformly-distributed binary string. There are many parameters that control this transformation, for example the length of the output binary sequence, the probability that two measurements coming from the same users will be mapped to the same binary sequence, etc.

Two abstractions, secure sketches and fuzzy extractors were proposed by Dodis, *et al* [32] to describe the process of transforming a biometric template into a reproducible, uniform binary sequence. A secure sketch can correct the noise between two biometric measurements coming from the same user by using some public information called a sketch. The result of a secure sketch is a reproducible sequence, which is not, however, uniformly distributed and thus not suitable to be used as cryptographic keys.

Fuzzy extractors can be used to extract randomness from biometric data to make the output of a secure sketch suitable for cryptographic keys. A fuzzy extractor, is a pair of two procedures. The first is the *generate* procedure, which is used once when the user generates a key for use on the untrusted server. The second is the *reproduce* procedure, which is used to authenticate the user to the server.

The *generate* procedure takes as input the noisy data x and then it outputs a public sketch p and a random binary sequence r. The generate procedure computes p in such a way that no significant information is revealed about the biometric data. The server will store the pair $\langle p, h(r)\rangle$, where $h(r)$ can for instance be a hash of the random binary sequence used for authentication.

The *reproduce* procedure takes as input a fresh measurement x' of the users biometric and the public sketch p, and outputs the random sequence r if x and x'

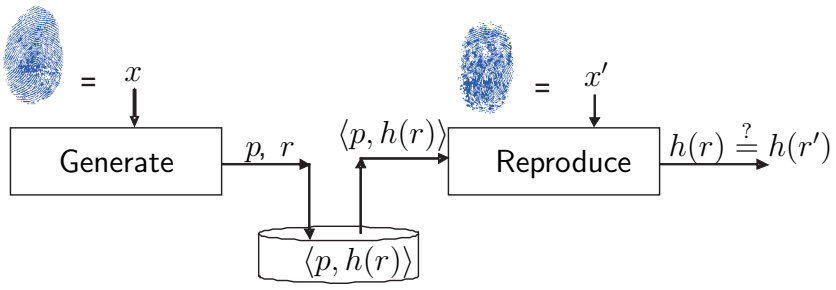


Figure 3.1: *User authentication scenario using a fuzzy extractor. During enrollment the* Generate *procedure outputs a random sequence r and a public string p. During authentication the hash of the output of the* Reproduce *procedure $h(r')$ is compared to the hash of the stored binary sequence $h(r)$. If the two match the user is authenticated.*

are similar enough. The similarity measure used is specific to the type of biometric and the algorithm used. The server can then verify that $h(r)$ matches the stored information. *Figure* 3.1 shows the architecture of a fuzzy extractor.

In the literature, fuzzy extractors are used for biometric data represented as an n point discrete vector, $x = (x_1, x_2, \ldots, x_n)$. However, when the noisy data originates from a continuous domain (i.e x is not a collection of discrete points, but is a probability distribution that describes the behavior of the each x_i) the general approach is a three step procedure applied on each vector element separately:

1. (Quantization.) A quantizer transforms the points modeled in the continuous domain into a suitable, discrete (binary) form of data, as is assumed in the fuzzy extractor model. After quantization, the *min-entropy* of the source data is fixed.

2. (Secure Sketch.) Apply error correcting techniques. No two biometric measurements are exactly the same, even when collected from the same person in two consecutive measurements. This step is used to compensate for the expected noise. The amount of noise, which can be compensated determines the reliability of the sketch. To improve the performance of the sketch, side-information also referred to as *helper data* [77] or a *public sketch* [32] is made public. The helper data, which is used for error correction reveals some information about the biometric string. The amount of information lost by publishing the helper data is known as the *entropy-loss*.

3. (Randomness Extraction.) Biometric data is not uniformly distributed. A randomness extractor will smoothen the distribution of the biometric data such that the output of the randomness extractor consists of nearly uniformly random bits.

Good fuzzy extractor constructions should produce high min-entropy random sequences, have high reliability and high security (i.e. low entropy loss). In each of the steps in the general approach one of these parameters is considered. After step 1, the available min-entropy can be estimated, during step 2 the sensitivity to changes in the continuous input values and the entropy-loss are determined and in step 3 determines how uniform the output binary string will be.

PROBLEM STATEMENT. The problem we see, is the gap which exists between the performance description of a biometric system, as seen by the biometric community and the notations used in the security community. The biometric community looks at error rates and sees the biometric data as continuously distributed. The most common performance measures are *false acceptance rate* (FAR) and *false rejection rate* (FRR). The former represents the probability that an attacker would be accepted by the biometric system as a legitimate user while the latter represents the probability that a legitimate user would not be recognized as such by the biometric system. As opposed to the biometric community the security community looks at *reliability*, *min-entropy* and *entropy-loss* and sees the biometric data as discretely distributed. Quantization is used as a means to bridge this gap. However, while steps 2 and 3 in the general approach are well understood, little is known about the effect of the choice of quantization strategy in step 1 and the effect of the quality of the biometric data for the overall construction.

CONTRIBUTION. In this chapter we extend the scope of the fuzzy extractors to continuous source data to bridge the gap between concepts like error rates (FAR and FRR) from biometrics and concepts like reliability, min-entropy and entropy-loss from security. We propose *cs-fuzzy extractors* as a unifying view on template protection schemes. This gives us new insights. We show that the key length of the binary sequence obtained by applying a fuzzy extractor to a biometric template depends on the amount of distinguishing information that exists in the biometric data or in other words on the error rates. We give an upper bound on the number of uniformly distributed bits that can be extracted from a given set of data characterized by FAR and FRR. This information can be used a-priori to evaluate the potential of a biometric data set in the context of a specific cryptographic application. To demonstrate the power of cs-fuzzy extractors we model existing template protection schemes in the new framework of cs-fuzzy extractors.

ASSUMPTIONS. In this chapter we make two important assumptions. The first assumption is related to the error model of our biometric data. We assume the additive noise model [28] for our data, where observations of each feature x_i can be perturbed by noise.

The second assumption is related to the existing knowledge, more precisely we assume the target FAR and FRR are chosen before the first step (quantization) in the general approach of a fuzzy extractor. Although this assumption is not very common in the world of security where one assumes limited knowledge regarding the probability distribution, the assumption *is* common in the world of biometrics. Assuming the $\mathrm{FAR}$and $\mathrm{FRR}$is known, is a reasonable assumption when considering the application scenario of a fuzzy extractor. As a pre-requisite for the fuzzy extractor design, the following steps are taken:

1. Collect biometric data from individual users. Each user will offer several measurements of their biometric data (e.g. fingerprint, face, voice, etc).
2. For each individual user a mean (or template) and a variance is estimated. The biometric data of the user is modeled as a continuous probability distribution. This continuous distribution is called *genuine distribution.*
3. The mean and the variance of the biometric data of *all* users is estimated. The obtained probability distribution is called the *impostor distribution.* It is expected that the variance of the impostor distribution is much larger compared to the genuine distribution.
4. A biometric classifier based algorithm is used to produce a *receiver operating curve* (ROC) by varying a discrimination threshold.
5. On the ROC curve a target FAR and FRR are chosen;

After the above steps are performed a generic fuzzy extractor scheme is applied on the biometric data using the discrimination threshold, which gives a direct link to target FAR and FRR.

ROADMAP. We look at related work in *Section* 3.1. Notation and background information is introduced in *Section* 3.2. The extension of the fuzzy extractor definition to continuously distributed data and the modeling of intrinsic relations between the error rates of biometric information and the parameters of a fuzzy extractor are presented in *Section* 3.3. Examples of practical template protection schemes modeled in the new framework of cs-fuzzy extractors are presented in *Section* 3.4. *Section* 3.5 concludes this chapter.

3.1 Related Work

In the literature the source of biometric data is considered to be either continuous or discrete. Therefore template protection schemes can be divided in two

classes. Representatives of the first class are continuous source shielding functions [49], the reliable component scheme [76] and the multi-bit scheme [25, 28]. The fuzzy vault [79] and the pin sketch [32] belong to the second class.

It is difficult to compare the performance of these schemes because there is no unified view on the security evaluation strategy. All authors estimate the error rate of their system in terms of FAR and FRR, but when it comes to evaluating the strength of the resulting binary sequence different authors have different opinions.

Monrose *et al.* [57] and Uludag *et al.* [81] compute the guessing entropy or the number of trials an attacker has to make to find the correct binary sequence while Zhang *et al.* [90] and Chang *et al.* [25] estimate the number of effective bits in the resulting key and propose a weighting system for choosing the best combination.

Chang *et al.* [24] analyze the security of a sketch by investigating the remaining entropy of the biometric data, given that the sketch is made public. The same approach is taken by Boyen *et al.* [19].

Dodis *et al.* [32] use both min-entropy and entropy loss.

Chen *et al.* [28] use as security measure the entropy of the output binary string and the mutual information between the output binary string and the published helper data. Tuyls *et al.* [76] estimate the information leakage using the conditional min-entropy between the public string and the binary string.

This brief summary highlights the importance of developing a unified theory that supports a thorough analysis of all schemes mentioned.

3.2 Preliminaries

Before we delve into the differences between discrete and continuous source biometrics, we need to establish some background. We start by giving our notation, as well as some basic definitions. Secondly, we summarize the fuzzy extractor for a discrete source as given by Dodis *et al.* [32] and Boyen *et al.* [19]. Thirdly, we briefly discuss the chosen model of the continuous source and its implications. Lastly, we remind the reader of the definitions of biometric error rates commonly used in the literature.

NOTATION. By capital letters we denote random variables while small letters are used to denote observations of random variables.

A random variable A is endowed with a probability distribution $f_A(a)$. When distinguishing between discrete and continuous random variable we use superscripts. With A^d we denote the random variable endowed with a discrete probability distribution $f_{A^d}(a)$ while A^c is used to denote the random variable endowed

with the continuous probability distribution $f_{A^c}(a)$. With $a \leftarrow A$ we denote an observation a of the random variable A.

It is common in the biometric literature [16] to model both the biometric data for all users in the target group (i.e. all users that the biometric recognition system is intended to recognize) and the biometric data for one generic user in the target group. In the rest of the paper we use the random variable Γ when referring to the target group, its distribution $f_\Gamma(\gamma)$ is known in the literature as the *impostor distribution* or the *background distribution* [16]. We use the random variable X when referring to biometric data which describes one user, where $f_X(x)$ is known in the literature as the *genuine user distribution* [16].

In the literature a biometric measurement is represented as an n point feature vector $x = (x_1, x_2, \cdots x_n) \in X$. However, without reducing the generality, in the remainder of this chapter, when referring to X we consider a one dimensional feature vector representation which captures all aspects of a template protection scheme.

We use the random variable P when referring to public data (the sketch) and R for random binary strings, which can be used as cryptographic keys and $\mathcal{U}_l$ to denote the set of uniformly distributed binary sequences of length l.

MIN-ENTROPY. When referring to cryptographic keys the strength of the key is measured as the probability that an adversary predicts the value of the secret key. The adversary's best strategy is to guess the most likely value. The *min-entropy* or the *predictability* of a random variable A denoted by $H_\infty(A)$ is defined as:

$$H_\infty(A) = -\log_2(\max_{a \leftarrow A} \Pr(A = a)).$$

Min-entropy can be viewed as the "worst-case" entropy [32].

FUZZY EXTRACTORS. For modeling the process of randomness extraction from noisy data Dodis, *et al.* [32] define the notion of a fuzzy extractor, see *Figure* 3.2. A fuzzy extractor extracts a uniformly random string r from biometric measurement x of user X in a noise tolerant way with the help of some public string p.

Enrollment is performed by the generate procedure, which on input of the biometric x extracts the binary string r and computes a public string p. During authentication, the reproduce procedure takes as input another biometric measurement x' and the public string p and it will output the binary string r if the two biometric measurements x and x' are within a predefined distance.

For a discrete random variable Γ^d, over the discrete metric space $\mathcal{M}$ endowed with a metric d, the formal definition of a fuzzy extractor [19, 32] is:

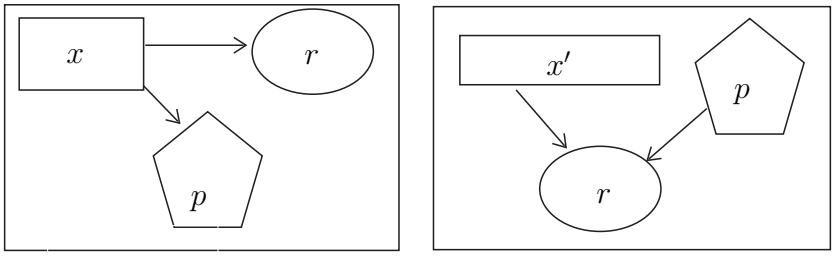


Figure 3.2: *A fuzzy extractor is a two step construction. The first step is the* Generate *procedure which on input of biometric x outputs a binary string r and a public string p. The* Generate *procedure is executed once. The second step is the* Reproduce *procedure which takes as input the biometric measurement x' and with the help of the public string p outputs r.*

Definition 1 *An $(\mathcal{M}, m, l, t, \epsilon)$ fuzzy extractor is a pair of randomized procedures, generate and reproduce, with the following properties:*

1. *The generation procedure on input of $x \in \mathcal{M}$ outputs an extracted string $r \in R$ and a public string $p \in P$.*
2. *The reproduction procedure takes an element $x' \leftarrow \Gamma$ and the public string $p \leftarrow P$ as inputs. The* reliability *property of the fuzzy extractor guarantees that if $d(x, x') \leq t$ and r and p were generated by $(r, p) \leftarrow \mathsf{Generate}(x)$, then $\mathsf{Reproduce}(x', p) = r$. If $d(x, x') > t$, then no guarantee is provided about the output of* Reproduce.
3. *The* security *property guarantees that for any distribution on the random variable Γ^d with min-entropy m, the string r is nearly uniform even for those who observe p: if $(R, P) \leftarrow \mathsf{Generate}(X)$, then $SD((R, P), (U_l, P)) \leq \epsilon$*

A fuzzy extractor is efficient *if* Generate *and* Reproduce *run in polynomial time.*

In other words, a fuzzy extractor allows to extract some randomness r from the biometric measurement x of a random user X^d chosen from the population of all users Γ^d. The reproduction procedure which uses the public string p produced by the generation procedure will output the string r as long as the biometric measurement x' is within distance t from the value x used during the generate procedure. This is the *correctness* property of the fuzzy extractor, the one we referred to earlier as *reliability*. The *security* property guarantees that the variable R looks uniformly random to an attacker and the probability that she guesses the value of r for from the first trial is approximately 2^{-m}. Security encompasses both *min-entropy* and the lack of uniformity of the random sequence.

We have two observations related to the above definition. Firstly, in the above definition $R = \{0,1\}^l$ thus a random binary string of length l. The public string $P = \{0,1\}^*$ which can be for example the syndrome of an error correcting code. However, there are template protection schemes that fit the model of the fuzzy extractors for which P is drawn from $\mathbb{R}$ [49] or $\mathbb{Z}$ [76]. Secondly, one can say that $f_X(x)$ has min-entropy only if it is a discrete probability distribution. Thus the above definition of fuzzy extractors works only when the biometric is represented discretely.

DISTRIBUTION MODELING. In the given examples, we model both the impostor, $f_{\Gamma^c}(\gamma)$ and the genuine distribution $f_{X^c}(x)$ as multivariate Gaussian distribution since it represents a common model for real world raw data. We write $f_{X^c}(x) = N(\mu_x, \sigma_x)$ and $f_{\Gamma^c}(\gamma) = N(\mu_\gamma, \sigma_\gamma)$. We emphasize that this assumption is not necessary for the definition of the *cs*-fuzzy extractors.

To estimate $f_{X^c}(x)$ multiple biometric measurements are collected from each user and the average μ_x and the standard deviation σ_x also known in the literature as *intra-class* variation, are estimated. The small perturbations between measurements hold important information. They represent an estimate on how far from the average other genuine samples will be. This is used to establish suitable probabilities of value acceptance and rejection area.

In controlling the error rates of a biometric classifier algorithm of interested is the *background distribution*, which represents the distribution of all the users enrolled in the biometric system. The background distribution is computed by estimating an average μ_Γ and a standard deviation σ_Γ on the data of all users. The background distribution is also known in the literature as *impostor distribution* since it assumed to be public information and an attacker can use this information produce a FAR or a FRR.

ERROR RATES. The error rates of a biometric system are determined by the accuracy with which the matching engine can determine the similarity between a measured sample x' and the expected value x of distribution $f_{X^c}(x)$ [16]. We can construct two hypotheses:
[H_0**]** the measured x' is coming from the authentic user X;
[H_1**]** the measured x' is not coming from the authentic user X;

The matching engine has to decide whether H_0 or H_1 is true. To express the accuracy of a biometric system the terms *false acceptance rate*, FAR and *false rejection rate*, FRR are used. The *false acceptance rate* is a Type I error and represents the probability that H_0 will be accepted when in fact H_1 is true. The *false rejection rate* is a Type II error and represents the probability that the outcome of the matching engine is H_1 but H_0 is true.

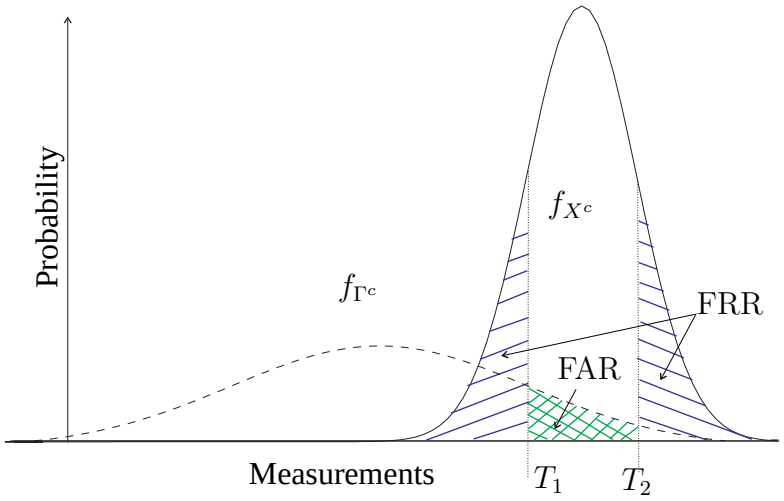


Figure 3.3: *Threshold $\langle T_1, T_2 \rangle$ determines acceptance and rejection regions.*

We have a false acceptance every time another user, from the distribution $f_{\Gamma^c}(\gamma)$ is generating a measurement which is in the acceptance region described by the interval $[T_1, T_2]$, see *Figure* 3.3. We can then write

$$\mathrm{FAR} = \int_{T_1}^{T_2} f_{\Gamma^c}(\gamma) d\gamma$$

Every time the user X with the distribution $f_{X^c}(x)$ produces a sample that is in the rejection area, see *Figure* 3.3 he will be rejected, thus

$$\mathrm{FRR} = 1 - \int_{T_1}^{T_2} f_{X^c}(x) dx.$$

Dodis *et al.* [32] assume the probability distribution associated to the random variable Γ^d to be discrete for the definition of fuzzy extractor. Therefore, the class of template protection schemes that use continuous sources do not fit this model. The subject of next section is the extension of fuzzy extractor definition to continuous source distributions.

FROM CONTINUOUS TO DISCRETE DISTRIBUTIONS. Definition 1 relies on a source random variable Γ^d with min-entropy m. How can we construct a source with min-entropy m out of a continuous distribution $f_{\Gamma^c}(\gamma)$?

A continuous probability distribution can be transformed into a discrete probability distribution by means of quantization. Quantization of a variable Γ^c means

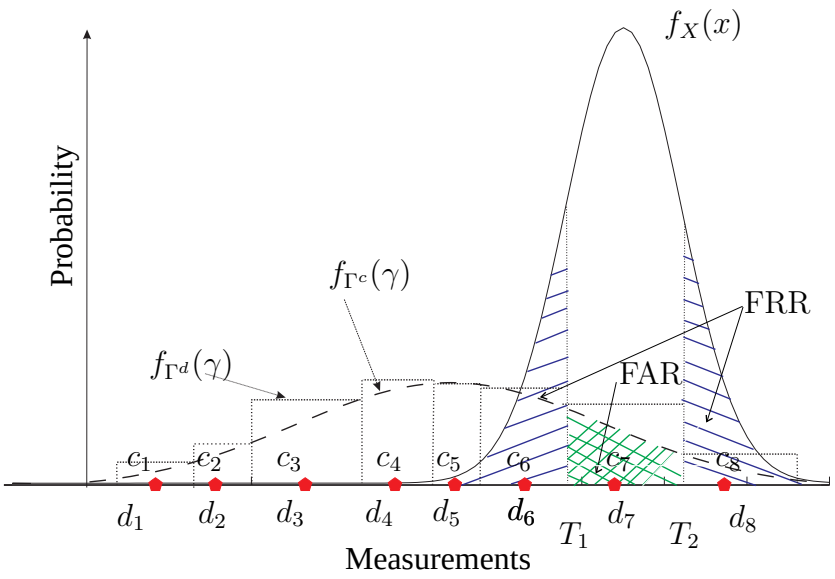


Figure 3.4: *Quantization of random variable* Γ^c *with continuous probability distribution* $f_{\Gamma^c}(\gamma)$ *into random variable* Γ^d *with discrete probability distribution* $f_{\Gamma^d}(\gamma)$. *The decision regions of the quantization functions are the intervals* $\{d_1, \cdots d_8\}$ *while the centroids are situated inside each interval and denoted with* $\{c_1, \cdots c_8\}$.

sampling its probability distribution $f_{\Gamma^c}(\gamma)$ and rounding the values of Γ^c to predefined points. By quantization the random variable Γ^c with probability distribution $f_{\Gamma^c}(\gamma)$, is transformed into random variable Γ^d with discrete probability distribution $f_{\Gamma^d}(\gamma)$, see *Figure* 3.4.

Formally, a quantizer is a function that maps each $\gamma \leftarrow \Gamma^c$ into the closest point $c_i \leftarrow \Gamma^d$, by

$$Q(\gamma) = \arg \min_{c_i \leftarrow \Gamma^d} d(\gamma, c_i)$$

where d is a suitable distance measure for the metric space of random variable Γ^d. The elements of Γ^d are known as *reconstruction point* $\{c_1, c_2, \ldots\}$. The subset of all elements $\gamma \leftarrow \Gamma^c$ that are closer to one reconstruction point than to any other reconstruction point is called a *decision region*.

When Γ^c and Γ^d are one dimensional, Q is called a *scalar* quantizer. In the scalar case, the length of the decision region is called the *step size.* If all decision regions of a quantizer are equal, the quantizer is *uniform*.

EXAMPLE. In the setting of *Figure* 3.4 the result of quantization is the discrete distribution f_{Γ^d} in this picture. The probability of selecting one decision region d_i

is computed as

$$p_i = \int_{d_i} f_{\Gamma^c}(\gamma) d\gamma.$$

As a result of quantization the continuous distribution $f_{\Gamma^c}(\gamma)$ has been transformed into the discrete distribution $f_{\Gamma^d}(\gamma) = \{p_1, \cdots p_8\}$.

After quantization of $f_{\Gamma^c}(\gamma)$ a user X with probability distribution function $f_{X^c}(x)$ is represented by one decision region, called the *authentic region*. From the perspective of the user there are two possible types of quantization strategies [41]. The first one is *user specific quantization* [25, 28] where the authentic region is chosen first such that it represents the user probability distribution function $f_{X^c}(x)$ as well as possible and the other decision regions and centroids are chosen afterwards. In this case $f_{\Gamma^c}(\gamma)$ is quantized differently for each user and for each a specific set of quantization boundaries are stored as public information. The second strategy is *user independent quantization* where the decision regions are chosen such that they are optimal for the average user. In this case there is only one set of quantization boundaries for all users [49].

The advantage of using a user specific quantization is a reliable template protection scheme. The quantization intervals are tuned to the specific distribution of each user and as a result the error rates are lower. The disadvantage however is that for each user, the quantization boundaries have to be computed independently and stored separately. As a result more data is stored in the database and more information (user-specific boundaries) is leaked about each user. A user independent quantization scheme stores less information and more importantly less user specific information. However, the cost is a lower reliability of the scheme.

Regardless of the quantization, the authentic region is chosen such that the probability associated to the authentic region, given the user probability distribution function $f_{X^c}(x)$, i.e.

$$p_{auth} = \int_{d_i} f_X^c(x) dx$$

is maximized, for a given d_i. Also, the authentic region determines both the FAR and the FRR.

In the example in *Figure* 3.4 the authentic region is d_7 and the FAR is represented by the double dashed area. The probability of a false rejection is determined by what is left from the probability distribution function $f_{X^c}(x)$ after removing p_{auth}, in *Figure* 3.4 the dashed area.

The min-entropy of the random variable Γ^d obtained after quantization, in the setting of *Figure* 3.4 is defined as $H_\infty(\Gamma^d) = -\log_2 p_{\max}$ where

$$p_{\max} = \max_i p_i.$$

In the rest of this chapter we extend the scope of the fuzzy extractors to continuously distributed data. We quantize the continuously distributed source and feed the result, the discrete source into the fuzzy extractor. We take one step further and generalize the connections between the parameters used for description of the biometric data (FAR , FRR) and the parameters of a fuzzy extractor.

MIN-ENTROPY FOR A CONTINUOUS SOURCE. For a random variable Γ^c with a continuous type of probability distribution function its min-entropy depends on the precision used to represent its elements. This topic is addressed in detail by Li *et al.* [48]. The min-entropy can be applied only to discrete type probability distribution functions, or after quantization as shown in the previous paragraph of the continuous probability distribution function:

$$H_\infty(Q(\Gamma^c)) = H_\infty(\Gamma^d) = -\log_2 \max_{\gamma \in \Gamma^d} \Pr(\Gamma^d = \gamma) \tag{3.1}$$

In the remainder of this chapter when referring to the min-entropy of a continuous random variable Γ^c we refer to relation 3.1.

3.3 Fuzzy extractors for continuous distributions

We show in this section that there is a natural link between the parameters of a fuzzy extractor $(\Gamma^d, m, l, t, \epsilon)$ and the error rates used for the description of biometric data.

3.3.1 Relating min-entropy m and FAR

The effective key space size of a biometric is linked to the FAR by O'Gorman [61], showing that the FAR (i.e. the probability that a person is accepted by the biometric system although he is not enroled) is the probability of an attacker performing a brute force password guessing attack. It is assumed the attacker has initial information about his own biometric and that the attacker has to guess the biometric of a legitimate member of the target group. However, O'Gorman [61] does not take into account the case when a template protection scheme is used for the biometric information. In this section we link the FAR to the min-entropy of the key *extracted* from the biometric data.

Quantization of a continuously distributed random variable Γ^c creates a tight relation between the min-entropy m of the random variable Γ^d after quantization and the error rates of the biometric system. For the variable R to have a high min-entropy and thus low probability that an attacker finds the correct value for

r one possibility is to lower the values of all the probabilities p_i. Unfortunately, by lowering p_i the FRR increases. The proposition below makes the connection between the error rates of the biometric data and the concept of min-entropy.

Proposition 1 *For a random variable Γ^d with probability density function $f_{\Gamma^d}(\gamma)$ the min-entropy m satisfies the relation $m \leq -\log_2 \mathrm{FAR}$ with equality when $p_{auth} = p_{\max}$.*

Proof: We take $p_{\max} = \max_i p_i$. Since $p_{\max} \geq p_{auth}$, we know that:

$$m = -log_2 p_{\max} \leq -log_2 p_{auth} = -log_2 \mathrm{FAR}$$

Corrolary 1 $\mathrm{FAR} \leq 2^{-m}$ *with equality when $p_{auth} = p_{\max}$.*

Proposition 2 *For a discrete random variable X, the min-entropy, $H_\infty(X)$ is maximized when the probability distribution of variable X is uniform.*

Proof: Assume that X and Y are two discrete random variables defined on the same support with n elements. Variable X has a uniform probability distribution and variable Y has a non-uniform probability distribution. The min-entropy of X is $H_\infty(X) = n$ bits.

We will prove the proposition by deriving a contradiction. Assume:

$$H_\infty(Y) > H_\infty(X)$$

this means

$$-\log_2 \max_{y \in Y} \mathsf{Pr}(Y = y) > -\log_2 \max_{x \in X} \mathsf{Pr}(X = x)$$

which means that

$$\max_{y \in Y} \mathsf{Pr}(Y = y) < \frac{1}{n}$$

which is impossible since the probabilities in Y have to sum up to 1 and both X and Y are defined on the same support. This proves the relation $H_\infty(Y) > H_\infty(X)$ is false and variable X with uniform probability distribution has maximum min-entropy.

Observation: m, the min-entropy of $f_{\Gamma^d}(\gamma)$ is maximized when the probabilities associated with the discrete distribution $f_{\Gamma^d}(\gamma)$ are uniform.

An example of an optimal quantization scheme from this perspective is given by Chen *et al.* [28], which is discussed in detail in *Section* 3.4.3.

3.3.2 Relating threshold t and FRR

According to *Definition* 1 the $\mathsf{Reproduce}(x', p)$ procedure will output the same binary sequence r as $\mathsf{Generate}(x)$ whenever x and x' are close. This means that x and x' probably belong to the same user X. In *Definition* 1 this is written as $d(x, x') < t$, where d is some metric, for example the Euclidian distance or the set difference metric. The value of t, does not say anything about the acceptance or the rejection probability of a user, which we feel, is more relevant. The probability of correctly identifying that two measurements belong to the same user is the opposite of a Type II error, thus the detection probability $P_d = 1 - \mathrm{FRR}$ is a suitable generalization of the threshold t.

3.3.3 CS-fuzzy extractors

The above relations lead us to the following definition of the fuzzy extractors for continuous sources.

Definition 2 *An* $(\Gamma^c, m, l, \mathrm{FRR}, \epsilon)$ cs*-fuzzy extractor (continuous source fuzzy extractor) is a pair of randomized procedures generate* $(\mathsf{Generate})$ *and reproduce* $(\mathsf{Reproduce})$*, with the following properties:*

1. $\mathsf{Generate}$ *is a (necessarily randomized) generation procedure, which on input* X^c *drawn from* Γ^c*, extracts a* private string $r \in \{0,1\}^l$ *and a* public string $p \leftarrow P$*, such that for any distribution* Γ_c *with min-entropy* m*, if* $(r, p) \leftarrow \mathsf{Generate}(X)$ *then* $SD((R, P), (U_l, P)) \leq \epsilon$*.*

2. $\mathsf{Reproduce}$ *is the reproduction procedure, which given a measurement* x' *sampled from* X *and a public string* $p \leftarrow P$ *outputs a string* $r \in \{0,1\}^l$*,* $r = \mathsf{Reproduce}(x', p)$*, where* $(r, p) \leftarrow \mathsf{Generate}(X)$*, with probability equal to the detection probability,* $P_d = 1 - \mathrm{FRR}$*.*

A cs-fuzzy extractor is efficient *if* $\mathsf{Generate}$ *and* $\mathsf{Reproduce}$ *run in polynomial time.*

A cs-fuzzy extractor preserves the mechanism of the generate and reproduce procedures as proposed in the original fuzzy extractor definition. The link between the parameters used in each model is described in the preceding sections, thus any fuzzy extractor is also a cs-fuzzy extractor.

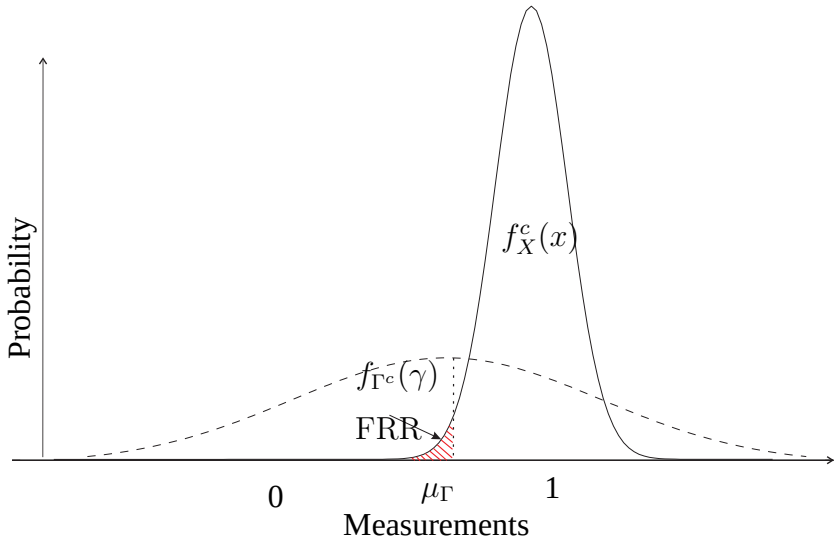


Figure 3.5: *Reliable component scheme.*

3.4 Examples

To demonstrate how one may use the cs-fuzzy extractor in practice, we take three prominent template protection schemes for continuous distributions from the literature and fit them in our model. As we discussed earlier, these template protection schemes cannot be described in terms of classical fuzzy extractors.

3.4.1 Reliable component scheme

One of the most intuitive schemes in the area of template protection is the *reliable component scheme* proposed by Tuyls *et al.* [76].

(Generate). During enrollment m samples $\{x^1, x^2, ..x^m\}$ are measured. This is followed by quantization, where a sequence $\{q^1, q^2, ..q^m\}$ is computed. During quantization each measured value x^j, $j = 1..m$ is compared to the imposter mean μ_Γ as shown in *Figure* 3.5. If $x^j \leq \mu_\Gamma$ then $q^j = 0$ else $q^j = 1$. A feature is *reliable* if all q^j are equal. Only in that case the feature will be used.

When x has many features, the public string p records the positions of the reliable components.

(Reproduce). During authentication, x' is measured and its value is compared to μ_Γ. The result of the comparison represents the key.

This scheme extracts 1 bit from every reliable component, with probabil-

ity equal to 1-FRR. We can characterize the reliable component scheme as a $(\Gamma^c, 1, 1, \mathrm{FRR}, 0)$ *cs-fuzzy extractor* where

$$\mathrm{FRR} = \begin{cases} \int_{-\infty}^{\mu_\Gamma} e^{\frac{-(x-\mu_X)^2}{2\sigma_X}} dx, & \mu_X > \mu_\Gamma \\ \int_{\mu_\Gamma}^{\infty} e^{\frac{-(x-\mu_X)^2}{2\sigma_X}} dx, & \mu_X < \mu_\Gamma. \end{cases}$$

The output bit is uniformly distributed, because the probability of a bit being equal to 0 is equal to the probability of the same bit being equal to 1. The main merit of this scheme is its reliability, because only the reliable components in the feature vector are chosen. The disadvantage is that many features are disregarded and depending on the quality of the data used the total length of the output key can be short.

3.4.2 Shielding functions

Linnartz *et al.* [49] were among the first to suggest how to get keys from continuously distributed sources. Their technique is inspired by watermarking. They propose a multiple quantization level system with odd-even bands, see *Figure* 3.6.

(Generate). As in the case of the previous template protection scheme, for each user X^c multiple measurements are taken and a mean μ_X and standard deviation σ_X are estimated. For one feature, the bit r is bound to the user X^c by shifting the mean of the user distribution, μ_X to the center of the closest even-odd interval, of length q if the value of the key bit r is a 1, or to the center of the closest odd-even q interval if the value of the key bit r is a 0, see *Figure* 3.6.

The public string p, also called the helper data is computed:

$$p = \begin{cases} (2n + \frac{1}{2})q - \mu_X & \text{when} \quad r = 1 \\ (2n - \frac{1}{2})q - \mu_X & \text{when} \quad r = 0 \end{cases}$$

Here $n \in \mathbb{Z}$ and is chosen such that: $-\frac{q}{2} < p < \frac{q}{2}$.

(Reproduce) is defined as:

$$\mathrm{Rep}(x', p) = \begin{cases} 1, & \text{when} \quad 2nq \leq x' + p < (2n+1)q \\ 0, & \text{when} \quad (2n-1)q \leq x' + p < 2nq \end{cases}$$

During authentication a noisy feature x' is extracted. The key bit is 1 if the sum of the noisy feature and the helper data is in an odd-even interval and is 0 otherwise.

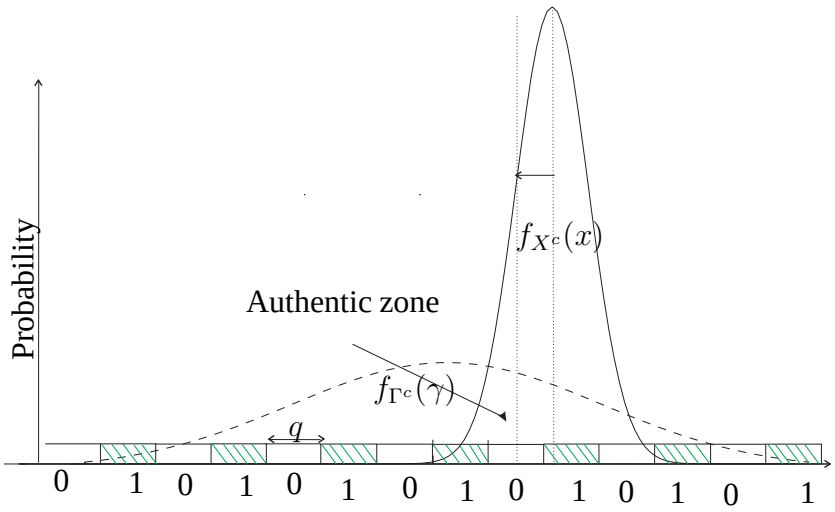


Figure 3.6: *Shielding function discretization, embedding a 0 value key bit.*

Whenever the measured value has an error greater than $\frac{q}{2}$ we can get an error in the key computation. This scheme can be written as a $(\Gamma^c, 1, 1, \mathrm{FRR}, 0)$ *cs-fuzzy extractor* where:

$$\mathrm{FRR} = \sigma_X \, 2\sqrt{2} \sum_{i=0}^{\infty} \int_{\frac{(1+4i)}{2\sqrt{2}}\frac{q}{\sigma}}^{\frac{(3+4i)}{2\sqrt{2}}\frac{q}{\sigma_X}} e^{-x^2} dx.$$

The FRR depends on the quantization step q. When q is large, the noise tolerance is high as well. On the other hand, if q is small, the FAR goes down. The output sequence is uniform in this scheme as well.

3.4.3 Chang multi-bit scheme

Chang *et al.* [25] select the distinguishing features from the biometrics of a user to extract multiple bits. For each feature the left and the right boundaries, $\mathcal{L}$ and $\mathcal{R}$ of the background distribution domain are selected so that with high probability a measurement from any user falls in this interval.

$(\mathsf{Generate})$. The selected FAR determines for each feature an authentic region delimited by T_1, T_2, see *Figure* 3.7. The whole region $\mathcal{L}$, $\mathcal{R}$ is divided in segments

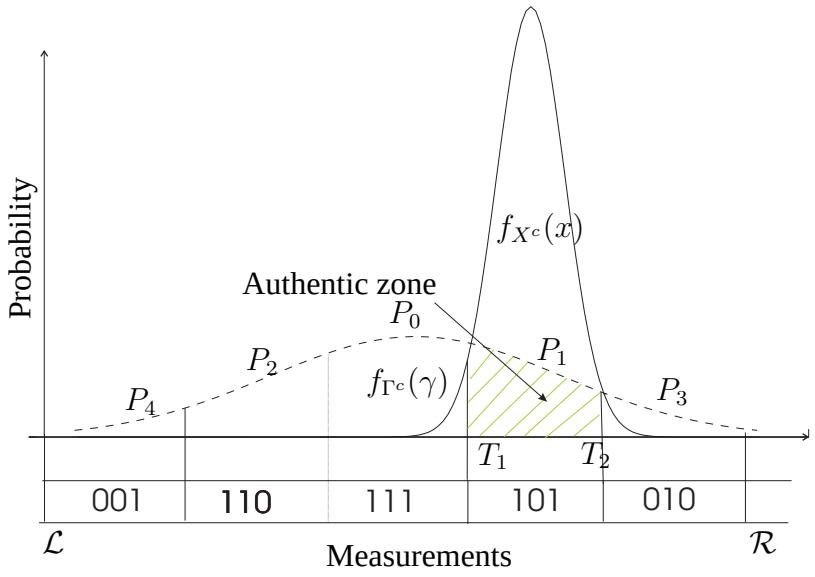


Figure 3.7: *Chang* et al. *[25] template protection scheme for continuously distributed biometric data.*

that have a length equal to the segment determined by T_1 and T_2. A label is associated with each segment. It can happen that some redundant segments are added to the left and to the right of $\mathcal{L}$ respectively $\mathcal{R}$ to use all labels of a given length. In *Figure* 3.7 three more segments with the labels 000, 100 and 011 can be added, here the genuine interval has label 101. The public string p contains the description of the intervals and the associated labels.

(Reproduce). Every time a user submits his biometric data to the system his feature will fall in one of the published intervals. The label associated with this interval represents the key of this user. An authentic user will be in the authentic area with probability 1-FRR.

This process is repeated for every user, for every feature. Thus they have defined an $(\Gamma^c, m, l, \mathrm{FRR})$ where

$$m = \log_2 \int_{\mu_\Gamma - \frac{|T_2 - T_1|}{2}}^{\mu_\Gamma + \frac{|T_2 - T_1|}{2}} f_{\Gamma^c}(\gamma) d\gamma$$

and $l = \log_2 \frac{|\mathcal{L}-\mathcal{R}|}{|T_2 - T_1|}$, and $\mathrm{FRR} = 1 - \int_{T_1}^{T_2} f_{\Gamma^c}(\gamma) d\gamma$.

The output sequence is not uniform and the consequences of this fact are analyzed in the next section.

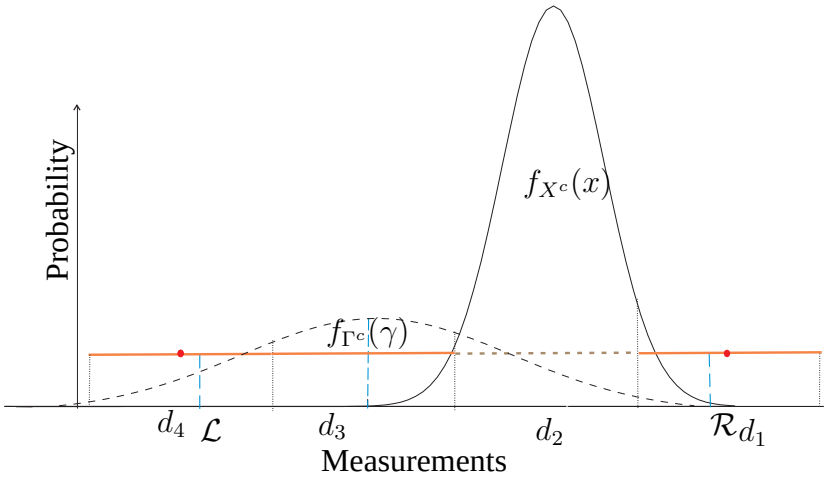


Figure 3.8: *In Chang* et al. *[25], the genuine interval can be guessed from one try.*

3.4.3.1 Comments on the distinguishable components

To generate stable cryptographic keys Chang *et al.* [25] propose to use only the distinguishable features for key generation. We show that in this case choosing the distinguishable feature makes life easier for an intruder and in a particular case she can almost certainly guess the authentic feature even from one try.

A feature is called distinguishable if the distance between the imposter mean μ_Γ and the authentic mean μ_X is sufficiently large. In the original paper a feature is distinguishable if $|\mu_\Gamma - \mu_X| > k_X \cdot \sigma_X$. Where k_X is a natural number chosen which determines the distinguishing degree of a feature. If k_X is large the feature is distinguishing, in other words characteristic to the user if k_X is small it is the other way around. In this scheme the authentic mean μ_X, due to the construction is always at the center of the authentic interval. The goal of an intruder trying to attack this scheme is to find the authentic interval with a minimal number of trials.

We model two types of intruders. Both intruders know the algorithm used for generating the key. However, we assume that the first imposter or the *type I imposter* knows the distribution of the population ($f_{\Gamma^c}(\gamma)$) while the second type intruder called, *type II* is stronger and also knows the parameters $\mathcal{L}$ and $\mathcal{R}$.

Type I The intruder knows that the authentic area of a user is far away from the global mean. In this case she can safely disregard the segment where the mean is situated. This leaves a new probability distribution $\frac{p_1}{1-p_0}, ..\frac{p_n}{1-p_0}$. as the central segment falls out.

Type II This attacker knows not only $f_{\Gamma^c}(\gamma)$ but she also knows the values of $\mathcal{L}$

and $\mathcal{R}$. In Chang *et al.* [25] these limits are computed as follows:

$$\begin{aligned} \mathcal{L} &= \min(\mu_\Gamma - k_\Gamma \sigma_\Gamma, \mu_X - k_X \sigma_X) \\ \mathcal{R} &= \min(\mu_\Gamma + k_\Gamma \sigma_\Gamma, \mu_X + k_X \sigma_X) \end{aligned}$$

Here k_Γ and k_X are natural numbers chosen by the designers of the system. For example Chang *et al.* [25] recommend for k_Γ the value 5 so that it covers almost the entire user distribution. If the margin $\mathcal{L}$ (and the reasoning is the same for $\mathcal{R}$) is somewhere situated in the right half of a segment we can safely eliminate that segment. According to the definition $\mathcal{L}$ will always be smaller than μ_X, which is in the middle of an interval. Thus the attacker can eliminate all intervals for which the middle value is smaller then $\mathcal{L}$.

If μ_X is in the the same segment as $\mathcal{L}$, according to the definition $\mathcal{L}$ will always be at the end of the authentic interval, never inside the interval. This leaves us the case where $\mathcal{L}$ is in the interval and in this case the attacker can safely eliminate this interval as well.

EXAMPLE. In *Figure* 3.8 we show how dangerous choosing the wrong combination of parameters can be. Assume the imposter distribution is divided in 4 intervals $\{d_1, d_2, d_3, d_4\}$. These intervals are published as helper data. The imposter has to guess which interval is the authentic one. It is assumed that the imposter distribution is known to the attacker.

The attacker can eliminate interval number d_3 because it contains the global mean μ_Γ and she knows that a distinguishable feature should be far away from the global mean. A type I attacker has 3 candidates for the correct authentic interval. However, the three intervals have different probabilities associated so the order of guessing will be: d_2, d_4, d_1. In this case she is lucky at the first trial. A type II attacker also knows the value of $\mathcal{L}$ and $\mathcal{R}$. The authentic mean is situated at the center of the authentic interval. The interval d_4 cannot be the authentic one since its middle value is smaller then $\mathcal{L}$. Thus the attacker can eliminate d_4. The same reasoning holds for $\mathcal{R}$ which eliminates d_1. As a result the intruder now has only one candidate for the authentic interval, namely d_2.

SOLUTION. A multi-bit biometric string generation algorithm that is not vulnerable to the above attacks is proposed by Chen *et al.* [28]. They propose a user-specific, likelihood ratio based quantizer which allows multiple bit extraction from each feature. The idea of using a likelihood ratio is driven by its optimal FAR versus FRR performance in many biometric applications. In the algorithm of Chen *et al.* [28] the quantization intervals are constructed such that they have equal probabilities. This gives an attacker (both type I and type II) no additional information regarding the genuine interval.

Chen *et al.* [28] carry out extensive experiments that compare the performance of their likelihood multi-bits quantization algorithm with the performance of the multi-bits quantization algorithm proposed by Chang *et al.* [25]. The main conclusion of these experiments is that when the user within-class variation is small the likelihood multi-bit quantization and multi-bit quantization have a similar performance, however, when the user within-class variation is large, which most often is the case in practice, likelihood multi-bit quantization outperforms the classical multi-bit quantization.

3.5 Conclusion and Future Work

A fuzzy extractor is a theoretical tool for modeling and comparing template protection schemes which use a discrete source. We generalize the definition to the cs-fuzzy extractor, which can also handle the continuous source data. We apply our model to three prominent template protection schemes in the literature.

Biometric recognition systems are evaluated using the false acceptance rate and the false rejection rate. The link between the two was hitherto not obvious even though they refer to the same data. In this chapter we show, that there is a natural connection between the false acceptance rate, false rejection rate and the parameters used to evaluate a template protection scheme implemented on the same data. We also show that the error rates have a direct influence on the length and robustness of the key extracted from the features of a user.

In this chapter we consider one dimensional or scalar quantization techniques. However, biometric data contains multiple features for each user. One approach towards the generalization to multiple independent features is to analyze each dimension independently. In this case, the relationship between the min-entropy and the FAR is as expected: the more dimensions we have, the lower the FAR is and the number of bits, which can be extracted increases. However, the FRR increases with the number of dimensions that are used.

Therefore, this may not be the best approach for aggregating multiple features. Zhang *et al.* [90] propose a better approach which can reduce both the FAR and the FRR by simultaneously analyzing all dimensions. In *Chapter* 4 we investigate the influence of various feature aggregation methods on the length and robustness of the key.

The contribution of this chapter is related to SECURE TEMPLATE STORAGE for continuous source data. We extend the theoretical model of fuzzy extractors to continuous source data in a new model we termed the *cs*-fuzzy extractor. In the new framework of *cs*-fuzzy extractors we relate the qualitative characteristics

of the input noisy data of a *cs*- fuzzy extractor and the properties of the uniform, reproducible string.

In the next chapter we extend the scope of the *cs*-fuzzy extractor to a practical construction termed the fuzzy embedder, which takes into account the problem of renewability of the uniform, reproducible sequence when the same noisy data is used for multiple applications.

Chapter 4

Embedding Renewable Cryptographic Keys into Noisy Data

When using a (cs-)fuzzy extractor in practice additional properties are needed, such as the renewability of the extracted strings, and the ability to use the fuzzy extractor directly on continuous input data instead of discrete data. Our contribution is threefold.

Firstly, we propose a *fuzzy embedder* as a generalization of both the fuzzy extractor and the *cs*-fuzzy extractor construction. A fuzzy embedder naturally supports renewability, as it allows a string to be embedded instead of extracted. It also supports direct analysis of quantization effects, as it makes no limiting assumptions about the nature of the input source.

Secondly, we give a general construction for fuzzy embedders based on the technique of quantization index modulation (`QIM`). We show that the performance measures of a `QIM`, translate directly to the security properties of the corresponding fuzzy embedder.

Finally, we show that from the perspective of the length of the embedded string, quantization in two dimensions is optimal. We present two practical constructions for a fuzzy embedder in the two-dimensional space. The first construction is optimal from a reliability perspective and the second construction is optimal in the length of the embedded string.

Cryptographic protocols rely on exactly reproducible key material. In fact, these protocols are designed to have a wildly different output if the key is only

perturbed slightly. Unfortunately, exactly reproducible keys are hard to come by, especially when they also need to have sufficient entropy. For example, one can hardly expect an average user to remember a password that consists of a string of 128 random bits. Luckily, it is relatively easy to find "fuzzy" sources, such as physically uncloneable functions (PUFs) [72] and biometrics [32]. However, such sources are inherently noisy and rarely uniformly distributed. The first, main difficulty in using the output of a fuzzy source as key material is the noise, which has to be corrected to produce the same key every time. To solve this problem, the notion of a secure sketch [48] has been proposed. The second difficulty lies in the fact that this output may have a non-uniform distribution, while it should be as close to uniform as possible to serve as a cryptographic key. A strong randomness extractor could be used to turn the reproducible output into a nearly uniform string. This naturally leads to the notion of a fuzzy extractor [32], which gives a reproducible, nearly uniform string as output. A common way of constructing fuzzy extractors is to combine a secure sketch with a strong randomness extractor.

However, when deploying a fuzzy extractor in practice, more difficulties arise. Firstly, even with the same input, it should be possible to generate many different keys. This is paramount when considering biometrics, where the number of possible inputs is limited (two eyes, 10 fingers etc.). To achieve renewability of the cryptographic key, the (fixed) output of the fuzzy extractor must be randomized, for instance by using a common reference string. Unfortunately, this falls outside the scope of the fuzzy extractor, even though it is recognized as an important and sensitive issue [19].

Secondly, as explained in the previous chapter the definition of a fuzzy extractor only accepts discrete sources as input. Existing performance measures for secure sketches, such as entropy loss or min-entropy, lose their relevance when applied to continuous sources [48]. This limitation can be overcome by quantizing the continuous input. Li, *et al.* [48] propose to define relevant performance measures with respect to the chosen quantization method. We argue that, instead of defining performance only after quantization, it is better to integrate the quantization into the definition, so that the intricacies of a continuous input can be studied.

CONTRIBUTIONS. Our contribution is threefold. Firstly, we propose a new primitive called a *fuzzy embedder*, which is a natural extension of a fuzzy extractor. A fuzzy embedder provides a randomized output, and handles arbitrary input sources.

The survey of template protection schemes presented by Uludag, *et al.* [80] divides known template protection schemes into two categories. The first category consists of constructions that extract a cryptographic key from a noisy input. Such

constructions are elegantly formalized by the notion of a fuzzy extractor. The second category consists of constructions that "bind" a cryptographic key to a noisy input. For this category only practical constructions are known, whereas formal models do not exist. The notion of a fuzzy embedder fills this important gap. A fuzzy embedder can be regarded as an extension of a fuzzy extractor, since it can embed a fixed string (for instance one obtained by applying a strong extractor to the input source) into a discrete source and thus achieve the same functionality, namely a randomized cryptographic key.

Interestingly, the fuzzy commitment [44] has a direct relation to a fuzzy embedder as well: removing the binding property from a fuzzy commitment scheme yields a fuzzy embedder, which suggests that a fuzzy commitment is more general than a fuzzy embedder.

Secondly, we propose a general construction for a fuzzy embedder, using data hiding techniques from the watermarking community. Our construction is based on Quantization Index Modulation (`QIM`), which is a watermarking method that can achieve efficient trade-offs between the information embedding rate, the sensibility to noise and the distortion [27]. The construction of a fuzzy embedder is intuitive as most of the properties of a fuzzy embedder can be reduced directly to the properties of the underlying `QIM`. The trade-offs of the used `QIM` give rise to similar trade-offs in fuzzy embedder performance measures. In this setting, shielding functions [49] can be regarded as a particular construction of a fuzzy embedder, as they focus on one particular type of quantizer. However, they only consider one-dimensional inputs.

Thirdly, we investigate different quantization strategies for high dimensional data, and we show that quantization in two dimensions gives an optimal length of the embedded uniform string. Finally, we focus on the two-dimensional case, and give two practical constructions, one being optimal from the perspective of sensitivity to noise, and the other being optimal from the key length perspective.

4.1 Related Work

Reproducible randomness is the main ingredient of a good cryptographic system. Good quality uniform random sources are rare compared to the more common non-uniform sources. For example, biometric data is easily obtainable, high entropy data. However biometric data is not uniformly distributed and its randomness cannot be exactly reproduced. Depending on the source properties several constructions have been proposed for obtaining cryptographic keys from noisy sources.

Dodis, *et al.* [32] consider discrete distributed noise and propose fuzzy ex-

tractors and secure sketches for different error models. These models are not directly applicable to continuously distributed sources. Linnartz, *et al.* [49] construct shielding functions for continuously distributed data and propose a practical construction which can be considered a one-dimensional `QIM` scheme. The same approach is taken by Li, *et al.* [48] who propose quantization functions for extending the scope of secure sketches to continuously distributed data. In chapter 3 we analyze the achievable performance of such constructions given the quality of the source in terms of the false acceptance rate and false rejection rate of a biometric system.

The process of transforming a continuous distribution to a discrete distribution influences the performance of the overall system, which uses fuzzy extractors and secure sketches. Quantization is the process of replacing analogue samples with approximate values taken from a finite set of allowed values. The basic theory of one-dimensional quantization is reviewed by Gersho [36]. The same author investigates the influence of high dimensional quantization on the performance of digital coding for analogue sources [37]. `QIM` constructions are used by Chen and Wornell [27] in the context of watermarking. The same authors introduce dithered quantizers [26]. Moulin and Koetter [59] give an excellent overview of `QIM` in the general context of data hiding. Barron, *et al.* [15] develop a geometric interpretation of conflicting requirements between information embedding and source coding with side information.

The concept of a fuzzy embedder might seem related to concepts developed in the context of information theoretic key agreement [52] more precisely to secure message transmission schemes based on correlated randomness [53]. However, the settings of the problem are different compared to ours. While in secure message transmission based on correlated randomness the attacker and the legitimate participants have a noisy share of the same source data, in the fuzzy embedder setting the attacker does not have access to the data source.

ROADMAP. The rest of this chapter is organized as follows. In *Section* 4.3 we present the definition of a fuzzy embedder and highlight the differences with fuzzy extractors and fuzzy commitment. In *Section* 4.4 we propose a general construction of a fuzzy embedder from any `QIM` and express the performance in terms of the geometric properties of the underlying quantizers. In *Section* 4.5 we present two practical constructions for the quantization of two-dimensional space, and compare the properties of these constructions with the existing square lattice packing. The last section concludes this paper.

4.2 Preliminaries

Before we delve into the differences between discrete and continuous source noisy data, we need to establish some background. We start by giving our notation, as well as some basic definitions. Secondly, we summarize the fuzzy extractor for a discrete source as given by Dodis *et al.* [32] and Boyen *et al.* [19]. Thirdly, we briefly discuss the chosen model of the continuous source and its implications. Lastly, we remind the reader of the definitions of error rates commonly used in the literature.

NOTATION. Let $\mathcal{M}$ be an n-dimensional discrete, finite set, which together with a distance function $d_{\mathcal{M}} : \mathcal{M} \times \mathcal{M} \rightarrow \mathbb{R}^+$ is a metric space. Similarly, let $\mathcal{U}$ be an n-dimensional continuous domain, which together with the distance $d_{\mathcal{U}} : \mathcal{U} \times \mathcal{U} \rightarrow \mathbb{R}^+$ forms a metric space. When the domain is clear from the context we use d and drop the subscript.

By capital letters we denote random variables while small letters are used to denote observations of a random variables. Continuous random variables are defined over the metric space $\mathcal{U}$ while a discrete random variable is defined over the metric space $\mathcal{M}$. A random variable A is endowed with a probability density function $f_A(a)$. We use the random variable P when referring to public sketch data and R for random binary strings, which can be used as cryptographic keys.

ENTROPY. When referring to cryptographic keys the strength of the key is measured as the min-entropy, i.e. the probability that an adversary predicts the value of the secret key from one attempt. The adversary's best strategy is to guess the most likely value. The *min-entropy* or the *predictability* of a random variable A denoted by $H_\infty(A)$ is defined as:

$$H_\infty(A) = -\log_2(\max_{a \leftarrow A} \mathbf{Pr}(A = a)).$$

Min-entropy can be viewed as the "worst-case" entropy [32]. For two (possibly correlated) random variables A and B, the *average min-entropy* is defined as

$$\begin{aligned} \tilde{\mathbf{H}}_\infty(A|B) &= -\log\left(\mathbb{E}_{b \leftarrow B}\left[\max_{a \leftarrow A} Pr(A = a|B = b)\right]\right) \\ &= -\log\left(\mathbb{E}_{b \leftarrow B}\left(2^{-H_\infty(A|B=b)}\right)\right), \end{aligned}$$

which represents the remaining uncertainty about A given B or the amount of uncertainty left about variable A when variable B is made public [32] (both A and B are discrete random variables).

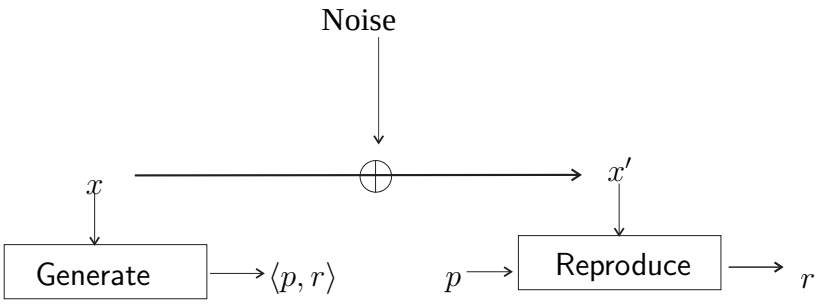


Figure 4.1: *A fuzzy extractor is a pair of two procedures* Generate *and* Reproduce*. The* Generate *procedure, which takes as input a noisy input x is executed first. The result is a random sequence r and a public sketch p, which is made public. The* Reproduce *procedure, which takes as input x' that is corrupted by noise and the public sketch p, will output r if x and x' are close.*

MUTUAL INFORMATION. By $I(A; B)$ we note the Shannon mutual information between the two random variables A and B, which is a measure of the mutual dependence between two random variable, in the following sense: $I(A; B) = 0$ if and only if A and B are independent random continuously distributed variables.

STATISTICAL DISTANCE. The Kolmogorov distance or *statistical distance* between two probability distributions A and B with the same domain is defined as:

$$SD(A, B) = \sup_v |\Pr(A = v) - Pr(B = v)|.$$

Informally, this is the largest possible difference between the probabilities that the two probability distributions can assign to the same event.

FUZZY EXTRACTORS. For modeling the process of randomness extraction from noisy data Dodis *et al.* [32] define the notion of a fuzzy extractor, see *Figure* 4.1. A fuzzy extractor extracts a uniformly random string r from a value x of random variable X in a noise tolerant way with the help of some public sketch p.

The Generate procedure takes a non uniformly random, noisy input x and produces two outputs: a public string p, and a key r. The key r is uniformly random given p, and according to the definition of p, reveals no information about the input x. However, one can reproduce r exactly when both p and x' (close to x) are presented to the Reproduce procedure.

For a discrete metric space $\mathcal{M}$ with a distance measure d, the formal definition of a fuzzy extractor [19, 32] is:

Definition 3 (Fuzzy Extractor) *An $(\mathcal{M}, m, l, t, \epsilon)$ fuzzy extractor is a pair of randomized procedures,* Generate *and* Reproduce*, with the following properties:*

1. *The generation procedure on input of $x \in \mathcal{M}$ outputs an extracted string $r \in R = \{0,1\}^l$ and a public helper string $p \in P = \{0,1\}^*$.*

2. *The reproduction procedure takes an element $x' \in \mathcal{M}$ and the public string $p \in \{0,1\}^*$ as inputs. The* reliability *property of the fuzzy extractor guarantees that if $d(x,x') \leq t$ and r, p were generated by $(r,p) \leftarrow \mathsf{Generate}(x)$, then $\mathsf{Reproduce}(x',p) = r$. If $d(x,x') > t$, then no guarantee is provided about the output of the reproduction procedure.*

3. *The* security *property guarantees that for any random variable X with distribution $f_X(x)$ of min-entropy m, the string r is nearly uniform even for those who observe p: if $(r,p) \leftarrow \mathsf{Generate}(X)$, then $SD((R,P),(N,P)) \leq \epsilon$ where N is a random variable with uniform probability.*

A fuzzy extractor is efficient *if* Generate *and* Reproduce *run in polynomial time.*

In other words, a fuzzy extractor allows to generate the random string r from a value x. The reproduction procedure which uses the public string p produced by the generation procedure will output the string r as long as the measurement x' is close enough. The *security* property guarantees that r looks uniformly random to an attacker and her chance to guess its value from the first trial is approximately 2^{-m}. Security encompasses both *min-entropy* and *uniformity* of the random string r when p are known to an attacker.

There are two shortcomings related to the above definition. Firstly, in the above definition $R = \{0,1\}^l$ thus a random binary string of length l. The public string $P = \{0,1\}^*$ which can be for example the syndrome of an error correcting code. However, there are template protection schemes that fit the model of the fuzzy extractors for which P is drawn from $\mathbb{R}$ [49] or $\mathbb{Z}$ [76]. Secondly, one can say that X has min-entropy only if it is a discrete probability density function otherwise its min-entropy depends on the precision or quantization used to represent the variable [48].

QUANTIZATION. A continuous random variable A can be transformed into a discrete random variable by means of quantization, which we write $Q(A)$. Formally, a quantizer is a function $Q : \mathcal{U} \rightarrow \mathcal{M}$ that maps each $a \in \mathcal{U}$ into the closest *reconstruction point* in the set $\mathcal{M} = \{c_1, c_2, \cdots\}$ by

$$Q(a) = \arg\min_{c_i \in \mathcal{M}} d(a, c_i).$$

where d is the distance measure defined on $\mathcal{U}$.

The *Voronoi region* or the *decision region* of a reconstruction point c_i is the subset of all points in $\mathcal{U}$, which are closer, with respect to a specific distance measure, to that particular reconstruction point than to any other reconstruction point.

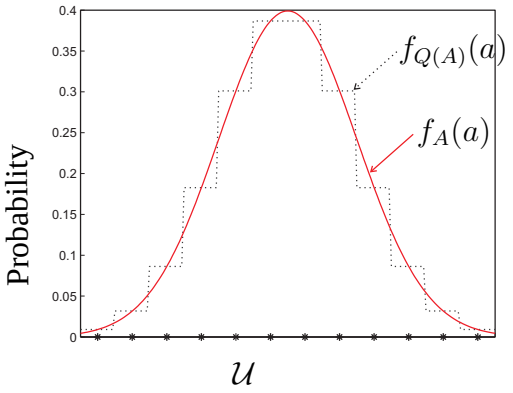


Figure 4.2: *By quantization, $f_A(a)$ (continuous line) is transformed into $f_{Q(A)}(a)$ (dotted line). We can write $Q(f_A(a)) = f_{Q(A)}(a)$.*

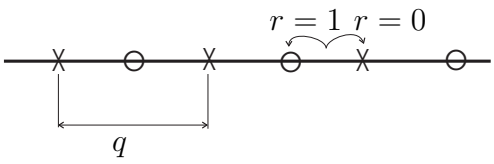


Figure 4.3: *Quantization of X with two scalar quantizers Q_0 and Q_1 both with step size q.*

We denote with V_{c_i} the Voronoi region of reconstruction point c_i. When A is one dimensional, Q is called a *scalar* quantizer. If all Voronoi regions of a quantizer are equal in both size and shape the quantizer is *uniform*. In the scalar case, the length of the Voronoi region is then called the *step size*. If the reconstruction points form a lattice the Voronoi regions of all reconstruction points are congruent.

By quantization the probability density function of the continuous random variable A, $f_A(a)$, which is continuous, is transformed into the probability density function $f_{Q(A)}(a)$, which is discrete (See *Figure* 4.2).

QUANTIZATION-BASED DATA HIDING CODES. Quantization based data hiding codes as introduced by Chen *et al.* [27] (also known as quantization index modulation) can embed secret information into a real-valued quantity. We start with an example of the simplest case.

Example 1. We want to embed one bit of information, thus $r \in \{0, 1\}$ into a real value x. For this purpose we use a scalar uniform quantizer with step size q, given by rounding $\frac{x}{q}$:

$$Q(x) = q \left\lceil \frac{x}{q} \right\rfloor .$$

The quantizer Q is used to generate a set of two new quantizers $\{Q_0, Q_1\}$ defined as:

$$Q_0(x) = Q(x + v_0) - v_0 \qquad \text{and} \qquad Q_1(x) = Q(x + v_1) - v_1$$

where

$$v_0 = \frac{q}{4} \text{ and } v_1 = -\frac{q}{4}.$$

In *Figure* 4.3 the reconstruction points for the quantizer Q_1 are shown as circles and the reconstruction points for the quantizer Q_0 are shown as crosses. The embedding is done by mapping the point x to one of the elements of these two quantizers.

For example, if $r = 1$, x is mapped to the closest $\circ$ point. The result of the embedding is the distance vector to the nearest $\times$ or $\circ$ as chosen by r. When during reproduction procedure x is perturbed by noise, the quantizer will assign the received data to the closest $\times$ or $\circ$ point, and output 0 or 1 respectively. The set of the two quantizers $\{Q_0, Q_1\}$ is called a `QIM`.

The amount of tolerated noise or the reliability is determined by the minimum distance between two neighboring reconstruction points. The size and shape (for high dimensional quantization) of the Voronoi region determines the tolerance for error. The number of quantizers in the `QIM` set determines the amount of information that can be embedded. By setting the number of quantizers and by choosing the shape and size of the decision region the performance properties can be finely tuned.

Formally, a *Quantization Index Modulation* data hiding scheme, can be seen as $\texttt{QIM} : \mathcal{U} \times R \rightarrow \mathcal{M}$ a set of individual quantizers $\{Q_1, Q_2, \ldots Q_{2^l}\}$, where $l = |R|$ and each quantizer maps $x \in \mathcal{U}$ into a reconstruction point. The quantizer is chosen by the input value $r \in R$ such that $\texttt{QIM}(x, r) = Q_r(x)$. The set of all reconstruction points is $\mathcal{M} = \bigcup_{r \in R} \mathcal{M}_r$ where $\mathcal{M}_r \subset \mathcal{M}$ is the set of reconstruction points of the quantizer Q_r.

We define the *minimum distance* $\sigma_{\min}$ of a `QIM`, as the minimum distance between reconstructions points of all quantizers in the `QIM`:

$$\sigma_{\min} = \min_{r_1, r_2 \in R} \min_{c^i_{r_1} \in \mathcal{M}_{r_1}, c^j_{r_2} \in \mathcal{M}_{r_2}} d(c^i_{r_1}, c^j_{r_2})$$

where $\mathcal{M}_{r_1} = \{c^1_{r_1}, c^2_{r_1}, \cdots\}$ and $\mathcal{M}_{r_2} = \{c^1_{r_2}, c^2_{r_2}, \cdots\}$. Hence, balls with radius $\frac{\sigma_{\min}}{2}$ and centers in $\mathcal{M}$ are disjoint.

Let ζ_r be the smallest radius ball such that balls centered in the reconstruction point of quantizer Q_r with radius ζ_r cover the universe $\mathcal{U}$. We define the *covering distance* $\lambda_{\max}$ as:

$$\lambda_{\max} = \max_{r \in \mathcal{R}} \zeta_r.$$

Any ball $B(c, \zeta_r)$ contains at least one ball $B(c_r, \sigma_{\min}/2)$ for $c_r \in \mathcal{M}_r, \forall r \in R$. Hence, balls with radius $\lambda_{\max}$ and centers in $\mathcal{M}_r$ cover the universe $\mathcal{U}$.

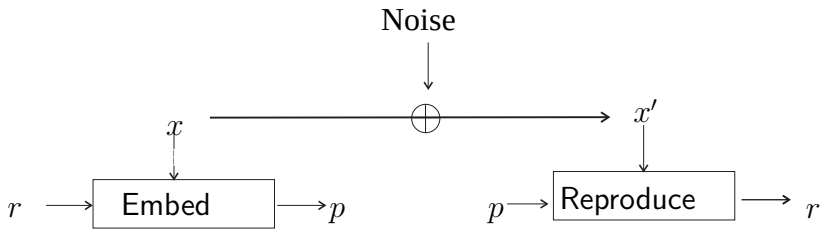


Figure 4.4: *A fuzzy embedder is a pair of two procedures* Embed *and* Reproduce. *The* Emded *procedure, which takes as input a noisy input x and a binary sequence r generated independently, is executed first. The resulting sketch p is made public. The* Reproduce *procedure, which takes as input a (possibly) an input x' which is corrupted by noise and the public sketch p, will output r if x and x' are close.*

A *dithered* `QIM` [26] is a special type of `QIM` for which all Voronoi regions of all individual quantizers are congruent polytopes (generalization of a polygon to higher dimensions). Each quantizer in the ensemble $\{Q_1, Q_2, \ldots Q_{2^l}\}$ can be obtained by shifting the reconstruction points of any other quantizer in the ensemble. The shifts correspond to dither vectors $\{v_1, v_2, \ldots v_{2^l}\}$. The number of dither vectors is equal to the number of quantizers in the ensemble.

Now that we have presented the necessary preliminaries, we are ready to present the notion of a fuzzy embedder in the next section.

4.3 Fuzzy Embedder

In this section we propose a general approach to embed cryptographic keys into noisy, continuous data. In addition, we show the relation between our new fuzzy embedder primitive and two related concepts, the fuzzy extractor and fuzzy commitment. It is worth stressing that the random key r is not extracted from the random x, but is generated independently, see *Figure* 4.4.

Definition 4 (Fuzzy Embedder) *A $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$-fuzzy embedder scheme consists of two polynomial-time algorithms $\langle$*Embed, Reproduce$\rangle$*, which are defined as follows:*

- Embed*: $\mathcal{U} \times R \to P$, where $R = \{0,1\}^l$. This algorithm takes $x \in \mathcal{U}$ and $r \in R$ as input, and returns a public string $p \in P$.*
- Reproduce*: $\mathcal{U} \times P \to R$. This algorithm takes $x' \in \mathcal{U}$ and $p \in P$ as input, and returns a string from R or an error $\perp$.*

Given any random variable X over $\mathcal{U}$ and a random variable R of size ℓ the parameters ρ, ϵ, δ are defined as follows:

- *The parameter ρ represents the probability that the fuzzy embedder can successfully reproduce the embedded key, and it is defined as*

$$\rho = \min_{r \in R} \max_{x \in \mathcal{U}} \Pr(\mathsf{Reproduce}(x', \mathsf{Embed}(x, r)) = r | x' \in X).$$

  In the above definition, the maximum over $x \in \mathcal{U}$ ensures that we choose the best possible representative x for the random variable X. In most cases, this will be the mean of X.

- *The security parameter ϵ is equal to the mutual information between the embedded key and the public sketch P, and it is defined as*

$$\epsilon = I(R; \mathsf{Embed}(X, R)).$$

- *The security parameter δ is equal to the mutual information of the noisy data and the public sketch and is defined as*

$$\delta = I(X; \mathsf{Embed}(X, R)).$$

A few notes are needed to motivate our choice of the security measures of a fuzzy embedder construction. Since the public sketch is computed both on X and R, ϵ measures the amount of information revealed about X (biometric or PUF) and δ measures the amount of information P reveals about the cryptographic key R.

When evaluating security of algorithms, which derive secret information from noisy data, entropy measures like min-entropy and average min-entropy or entropy loss are appealing since these measures have clear security applicability. However, these measures can only be applied to a variable that has a discrete probability density function. In the case of a continuous random variable these entropy measures depend on the precision used to represent the values of a random variable, as shown in the next example for min-entropy.

Example. Assume that all points X are real numbers between $[0, 1]$ and are uniformly distributed. Assume further that points in X are represented with 2-digit precision, which leads to a min-entropy $H_\infty(X) = \log_2 100$. If we choose to represent points with 4-digit precision the min-entropy of X becomes $H_\infty(X) = \log_2 10000$, which is higher than $H_\infty(X) = \log_2 100$ although in both cases X is uniformly distributed on the interval $[0, 1]$.

More examples related to average min-entropy and entropy loss can be found in Li *et al.* [48]. We chose mutual information measure, i.e $I(X; P)$ and $I(R; P)$ because it captures the measure of dependence between two random variables regardless of their type of distribution discrete or continuous. A similar measure for

the dependence of two variables is the statistical distance between their distribution. In this case our choice is motivated by the generality given by the information theoretical measure.

FUZZY EXTRACTOR AND FUZZY EMBEDDER. From *Definitions* 3 and 4, we argue that a fuzzy embedder is more general than a fuzzy extractor, due to the following reasons:

1. The fuzzy embedder scheme accepts continuous data as input and can embed different keys, while in a practical deployment, a fuzzy extractor scheme must be combined with quantization and re-randomization to achieve the same goals as a fuzzy embedder.

2. Given a $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$-fuzzy embedder, we can construct a fuzzy extractor as follows:

   - $\mathsf{Generate}'$: $\mathcal{U} \rightarrow P \times R$. This algorithm takes $x \in \mathcal{U}$ as input, chooses $r \in R$, and returns $p = \mathsf{Embed}(x, r)$ and r.
   - $\mathsf{Reproduce}'$: $\mathcal{U} \times P \rightarrow R$. This algorithm takes $x' \in \mathcal{U}$ and $p \in P$ as input, and returns the value $\mathsf{Reproduce}(x', p)$.

4.4 Practical Construction of a Fuzzy Embedder

In this section, the following three practical issues are presented. Firstly, we construct a fuzzy embedder using a `QIM`. Secondly, we analyze the performance of this construction in terms of reliability and security. Thirdly, we investigate optimization issues when $\mathcal{U}$ is n-dimensional.

`QIM`-FUZZY EMBEDDER. A fuzzy embedder can be constructed from *any* `QIM` by defining the embed procedure as:

$$\mathsf{Embed}(x, r) = \texttt{QIM}(x, r) - x,$$

and the reproduction procedure as the minimum distance Euclidean decoder:

$$\mathsf{Reproduce}(x', p) = \widetilde{Q}(x' + p),$$

where $\widetilde{Q} : \mathcal{U} \rightarrow R$ is defined as

$$\widetilde{Q}(y) = \mathsf{argmin}_{r \in R} d(y, \mathcal{M}_r).$$

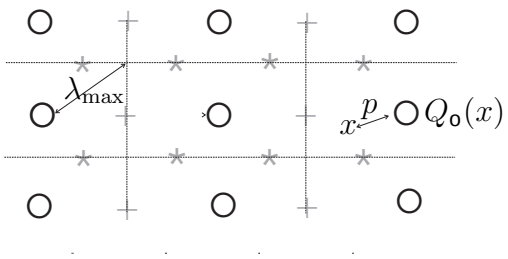


Figure 4.5: Embed *procedure of* `QIM` *fuzzy embedder*

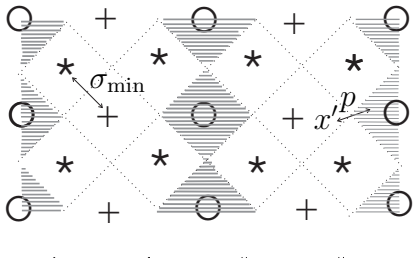


Figure 4.6: Reproduce *procedure of a* `QIM` *fuzzy embedder*

Example. Our construction is a generalization of the scheme of Linnartz, *et al.* [49]. *Figures* 4.5 and 4.6 illustrate the Embed respectively the Reproduce procedures for a `QIM` ensemble of three quantizers $\{Q_{\mathtt{o}}, Q_{+}, Q_{\star}\}$. During embedding, the secret $r \in \{\mathtt{o}, \star, +\}$ selects a quantizer, say $Q_{\mathtt{o}}$. The selected quantizer finds the reconstruction point $Q_{\mathtt{o}}(x)$ closest to x and the embedder returns the difference between the two as p, with $p \leq \lambda_{\max}$. Reproduction of p and x' should return `o` if x' is close to x, however, this happens only if $x' + p$ is close to $Q_{\mathtt{o}}(x)$ or in other words, if $x' + p$ is in one of the Voronoi regions of $Q_{\mathtt{o}}$ (hatched area in *Figure* 4.6). Errors occur if $(x' + p)$ is not in any of the Voronoi regions of $Q_{\mathtt{o}}$, thus the size and shape (for $n \geq 2$) of the Voronoi region parameterized by the radius of the inscribed ball $\sigma_{\min}/2$ determines the probability of errors.

4.4.1 Reliability

In the following lemma, we link the reliability of a `QIM`-fuzzy embedder to the size and shape of the Voronoi regions of the employed `QIM`.

Lemma 1 (Reliability) *Let* $\langle \mathsf{Embed}, \mathsf{Reproduce} \rangle$ *be a* $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$ *`QIM`-fuzzy embedder, and let* X *be a random variable over* $\mathcal{U}$ *with joint density function* $f_X(x)$*. For any* $r \in R$*, we define*

$$\rho(r) = \int_{\mathcal{V}_r} f_X(y - \mathsf{Embed}(X, r))dy,$$

where $\mathcal{V}_r = \bigcup_{c \in \mathcal{M}_r} V_c$ *is the union of the Voronoi regions of all reconstruction points in* $\mathcal{M}_r$*. Then the reliability is equal to*

$$\rho = \min_{r \in R} \rho(r).$$

Proof: Since $\rho(r)$ is exactly the probability that an embedded key r will be reconstructed correctly, the statement follows from the definition.

In most practical applications, noise has two main properties: larger distances between x and the measurement x' are increasingly unlikely, and the noise is not directional. Thus the primary consideration for reliability is the size of the inscribed ball of the Voronoi regions, which has radius $\sigma_{\min}/2$.

Corrolary 2 (Bounding ρ.) *In the settings of Lemma 1, the reliability ρ can be bounded by*

$$\min_{r \in R} \sum_{c \in \mathcal{M}_r} \int_{B(c, \frac{\sigma_{\min}}{2})} f_X(y) dy \leq \rho$$

where $B(c, r)$ is the ball centered in c with radius r.

Proof. The above relation follows from the definition of reliability, since $B(c, \frac{\sigma}{2}) \subset V_c$ and $y = x + \mathsf{Embed}(X, r)$ is always a reconstruction point.

Corollary 2 shows that reliability is at least the sum of all probabilities computed over balls of radius $\frac{\sigma_{\min}}{2}$ inscribed in the Voronoi regions. Thus the size of the inscribed ball is an important parameter, which determines the reliability to noise.

Example. In two dimensional space there are three regular polytopes, which tile the space: triangle, square and hexagon. If the size of the inscribed circle is equal for all three, in case of a spherically symmetric distribution like the normal distribution the hexagon has superior reliability performance compared to the other two polytopes because its shape is more close to a ball. The shape of the decision region that inscribes the ball is important as well as we show in *Section* 4.5.

4.4.2 Security

In this section we link the security of a fuzzy embedder to the covering radius, $\lambda_{\max}$ of the employed `QIM`.

We start this paragraph with one observation. If an attacker learns the value x she can reproduce the value r with the help of the public value p. However, if an attacker learns the secret key r, she could potentially circumvent the security altogether but cannot reproduce x. We illustrate this observation in the next example.

Example. In the fuzzy embedder example given in *Figure* 4.6, the attacker can choose between three different key values$\{\circ, +, \star\}$. Assume she learns the correct key, in our example $\circ$. To find the correct value for x she still has to decide which of the reconstruction points of the quantizer $Q_\circ$ is closest to x. Without any other information this is an impossible task since the quantizer $Q_\circ$ has an infinite number of reconstruction points.

The public sketch p leaks information about both the random string r (the amount of information revealed is δ) and the value x (the amount of information revealed is denoted with ϵ). We note that full disclosure of the string r is not enough to recover x.

We now consider how large δ, the leakage on the key can be in terms of P, which due to our construction is a continuous variable. We know that any $p \in P$ has the property that $p \leq \lambda_{\max}$. A technical difficulty in characterizing the size of P arises as P is not necessarily discrete. Tuyls ***et al.*** [77] show the following result, establishing a link between the continuous and the quantized version of P denoted here with P_d.

Lemma 2 (Tuyls *et al.* [77]) *For continuous random variables X, Y and $\xi > 0$, there exists a sequence of quantized random variables X_d, Y_d that converge pointwise to X, Y (when $d \to \infty$) such that for sufficiently large d, $I(X;Y) \geq I(X_d;Y_d) \geq I(X;Y) - \xi$.*

From the lemma above we have: $I(R;P_d) \leq H(P_d) \leq |P_d|$, P_d is a quantized representation, of the public sketch P, using a uniform scalar quantizer with step d. The reason for quantizing P is to make it suitable for a digital representation. $|P_d|$ represents the size, in bits, of the sketch.

To limit the information loss of the construction, which is the result of publishing the sketch P_d, it is best to have $|P_d|$ as small as possible. However a small representation of P_d implies that the cardinality of the set of values of P_d is small as well. There are two ways in which we can achieve a small representation for P_d. The first is to limit the support on which P is defined, while the second is to choose a higher value for the quantization step d. The second approach is not convenient since the quantization that is used for P has to be used for the noisy data X thus we concentrate on the first option: limit the support on which P is defined.

In our construction, we have $|P_d| \leq \lambda_{\max}$. Thus by bounding the size of p we bound the value of δ. In the rest of this chapter, for simplicity reasons we use P when referring the P_d.

4.4.3 Optimization

In this paragraph, we analyze the key length allowed by the restrictions placed by our performance criteria on the embed and reproduce procedures. Firstly, we take a look at the reproduce procedure which ties in directly with the reliability. The minimum size of an error to produce a wrong decoding is $\sigma_{\min}/2$. Thus, the collection of balls centered in the reconstruction point of all quantizers with radius $\sigma_{\min}/2$ should be disjoint.

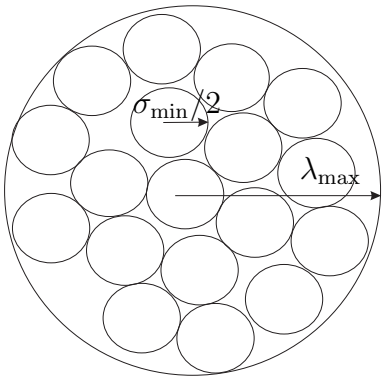


Figure 4.7: *Optimization of reliability versus security. Reliability is determined by the size of the ball with radius $\sigma_{\min}/2$. Each small ball has associated to its center a different key $r \in R$. The number of small balls inside the large ball with radius $\lambda_{\max}$ is equal to l, the number of elements in R. To have as many keys as possible we want to increase the number of small balls, thus we want* dense (sphere) packing. *The size of the public sketch $p \in P$ is at most $\lambda_{\max}$. Since for any $x \in \mathcal{U}$ we want to be within λ_{max} distance to a specific $r \in R$, large balls should* cover optimally *the space $\mathcal{U}$. When the point x falls in a region, which does not belong to any ball the* Reproduce *procedure gives the closest center of a small ball, thus* we want polytopes which tile the space.

Secondly, the result of the embed procedure for any arbitrary point x and any key $r \in R$ has to be smaller than the covering distance $\lambda_{\max}$. Hence, for each key r the collection of balls centered in the reconstruction points of Q_k and with radius $\lambda_{\max}$ should cover the entire space $\mathcal{U}$.

These two radii can be linked as follows:

Lemma 3 *The covering distance of a* `QIM` *, $\lambda_{\max}$ is bounded from below by:*

$$\lambda_{\max} \geq \sqrt[n]{N}\frac{\sigma_{\min}}{2}$$

where n represents the dimension of the universe $\mathcal{U}$ and N is the number of different quantizers.

Proof: As noted above, all balls with radius $\sigma_{\min}/2$ centered in the centroids of the whole ensemble are disjoint. Each collection of balls with radius $\lambda_{\max}$ centered in the centroids of an individual quantizer gives a covering of the space $\mathcal{U}$, see *Figure* 4.7.

Therefore, a ball with radius $\lambda_{\max}$, regardless of its center, contains at least the volume of N disjoint balls of radius $\sigma_{\min}/2$, one for each quantizer in the ensemble. Comparing the volumes, we have

$$s_n \lambda_{\max}^n \geq s_n N (\frac{\sigma_{\min}}{2})^n$$

where s_n is a constant only depending on the dimension.

The main conclusion of *Lemma* 3 is that for a `QIM`-fuzzy embedder to produce a long random string r, thus the length of r depends on the number of small balls which can be placed into a large ball.

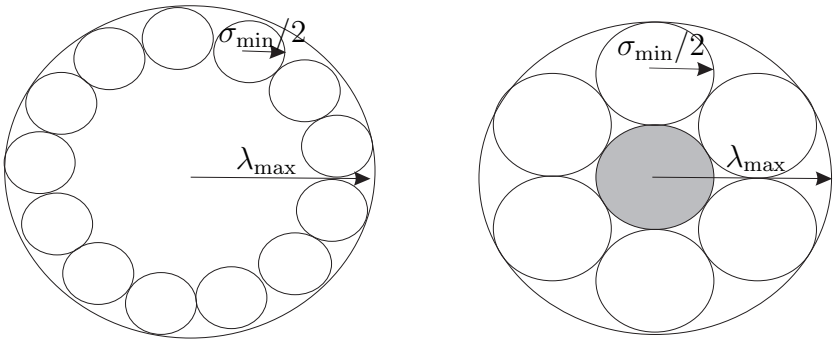
Consider the case when an intruder has partial knowledge about the random variable X. For example, she could know the average distribution of all (fingerprint) biometrics, or the average distribution of the PUFs. This average distribution is known in the literature as the *background distribution*. While any `QIM`-fuzzy embedder achieves equiprobable keys if the background distribution on $\mathcal{U}$ is uniform, the equiprobability can break down when this background distribution is non-uniform and known to the intruder. A legitimate question is: *how can a `QIM`-fuzzy embedder achieve equiprobable keys when the background distribution is not uniform?*

In the literature [25, 28, 49] it is often assumed that the background distribution is a multivariate Gaussian distribution. We make a weaker assumption, namely that the background distribution is not uniform but spherically symmetrical and decreasing. In other words, we assume that measurement errors only depend on the distance, and not on the direction, and that larger errors are less likely.

Thus, to achieve equiprobable keys given this background distribution, the reconstruction points must be equidistant as for example the construction in *Figure* 4.8 (a). Note that putting more "small" balls inside the "large"ball is not possible since they are not equiprobable. The problem with the construction in *Figure* 4.8 (a) is the size of the sketch which becomes large.

The natural question, which arises is: *what is the minimum sketch size attainable such that all keys are equiprobable for a given desired reliability?*

This question leads us to consider the kissing number $\tau(n)$, which is defined to be the maximum number of white n-dimensional spheres touching a black sphere of equal radius, see *Figure* 4.8 (b). The radius of the "small" balls determines



(a) (b)

Figure 4.8: *(a) Construction which yields equiprobable keys in case the background distribution is spherical symmetrical in the two dimensional space. (b) Optimal construction which results in minimal public sketch size and has equiprobable keys in the two dimensional space.*

reliability and the minimum $\lambda_{\max}$, such that a `QIM`-fuzzy embedder can be built is equal to the radius of the circumscribed ball as shown in *Figure* 4.8(b).

The next question we ask is: *for a minimum sketch size and a given reliability, are there dimensions which are better than others?* For example why not pack spheres in the three dimensional space where the kissing number is 12. For the same reliability: is it possible to obtain more keys? For most dimensions, only bounds on the kissing number are known [45, 89]. Assuming a spherically symmetrical and decreasing background distribution, there are only so many different equiprobable keys one can achieve:

Theorem 1 (Optimal high dimensional packing.) *Assume the background distribution to be spherically symmetrical and decreasing. For a* $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$ *`QIM`-fuzzy embedder with* $\dim(\mathcal{U}) = n$ *with equiprobable keys and* minimal sketch size*, we have that* $\ell \leq \tau(n)$.

Proof: The target reliability ρ will translate to a certain radius σ. In other words, we need to stack balls of radius σ optimally.

In *Figure* 4.9 we have three possible constructions for the `QIM`-fuzzy embedder, with different choices of number of quantizers in the set versus the size of the public sketch.

The construction in *Figure* 4.9 (a) cannot be used for data hiding since there is only one quantizer in the set. To achieve the maximum number of equiprobable keys without the sketch size getting too big, the best construction is to center the background distribution in one such ball, and to assign a different key to each

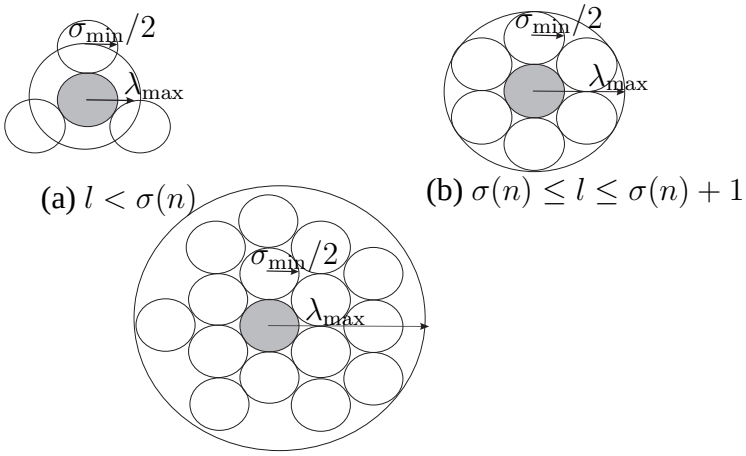


Figure 4.9: *Different choices for the number of quantizers in relation to $\lambda_{\max}$ in a* `QIM`*-fuzzy embedder construction. (a) There is only one quantizer in the* `QIM` *set. This construction cannot be used for data hiding. (b) Number of quantizers in the* `QIM` *set is equal to $\sigma_n + 1$, when the middle ball has a different codeword then the neighboring balls (e.g. the 7-hexagonal construction) or precisely equal to the σ_n, when the middle ball has no codeword associated (e.g. the 6-hexagonal construction). (c) The* `QIM` *set has more quantizers then the kissing number.*

touching ball as in *Figure* 4.9 (b). Construction in *Figure* 4.9 (c) yields a higher value for $\lambda_{\max}$ and is not optimal from the perspective of the size of the sketch.

The trade-off between the number of quantizers (and thus the length of the output sequence) and the size of the sketch can be seen by comparing constructions in *Figure* 4.9 (b) and *Figure* 4.9 (c). As the number of quantizers increases so does the size of the sketch.

Thus the number of possible equiprobable keys, when the background distribution is spherically symmetric and decreasing, is upper bounded by the kissing number $\tau(n)$.

Combined with known bounds on the kissing number [45, 89], we arrive at the following, somewhat surprising conclusion:

Corrolary 3 *Assuming a spherically symmetrical and decreasing background distribution on $\mathcal{U}$ and equiprobable keys, for a $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$* `QIM`*-fuzzy embedder, the most equiprobable keys are attained by quantizing two dimensions at a time, leading to*

$$N(n) = 6^{\lfloor \frac{n}{2} \rfloor} 2^{(n - 2\lfloor \frac{n}{2} \rfloor)}$$

different keys.

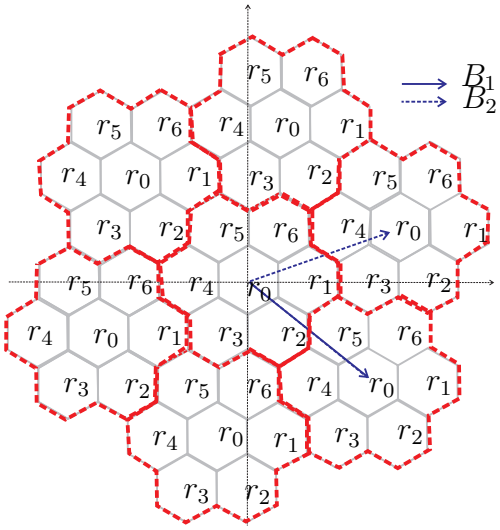


Figure 4.10: Reproduce *procedure of the 7-hexagonal tiling*

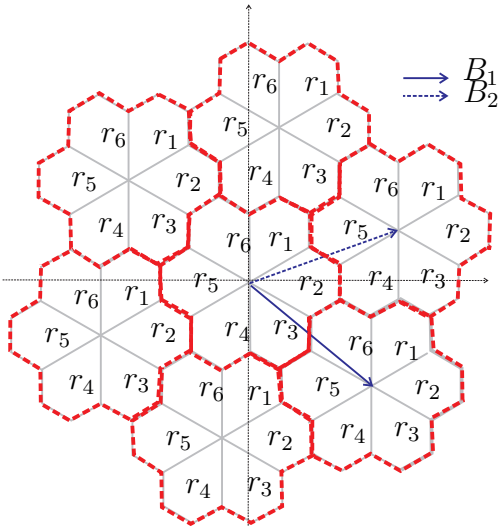


Figure 4.11: Reproduce *procedure of the 6-hexagonal tiling*

Proof: Known upper bounds [45] on the kissing number in n dimensions state that $\tau(n) \leq 2^{0.401n(1+o(1))}$. This means that $N(n) \geq \tau(n)$ in all dimensions, since $N(n) \approx 2^{1.3n}$ and small dimensions can easily be verified by hand. Also note that $N(n_1 + n_2) \leq N(n_1)N(n_2)$. Thus quantizing dimensions pairwise gives the largest number of equiprobable keys for any spherically symmetric distribution.

Example. Given a vector $X = (X_1, X_2, \cdots X_{10})$ there are several choices when considering quantization. One possibility is to quantize each of the elements $X_i, i \in \{1, 10\}$ independently. A second choice is to quantize pairs of elements (X_i, X_j) where $i \neq j$ and $i, j \in \{1, 10\}$. Another option is to quantize three elements at a time (X_i, X_j, X_s) where $i \neq j \neq s$ and $i, j, s \in \{1, 10\}$. We illustrate in this example that the two-dimensional quantization is optimal in the sense of *Corollary* 3. *Table* 4.1 shows the effect of quantization on the key space for different dimension choices.

- For *two-dimensional* quantization (*Table* 4.1), the kissing number is equal to 6, the 10 elements of vector X are grouped in 5 subsets of 2 elements each. For each subset, we can embed at most 6 keys and for the 5 pairs we have in total a key space of 6^5 possible keys.

- For *three-dimensional* quantization (*Table* 4.1), kissing number is 12, the 10 elements of X can be grouped as 3 pairs of 3 elements and there is one vector element left which can only be quantized in one dimension. The number of possible keys is $12^3 \times 2$.

Dimension	σ_n	Subsets	Key Space
1	2	1×10	$2^{10} = 1024$
2	6	2×5	$6^5 = 7776$
3	12	$3 \times 3 + 1$	$12^3 \times 2 = 3456$
4	24	$4 \times 2 + 2$	$24^2 \times 6 = 3456$
5	40	5×2	$40^2 = 1600$
6	72	6+ 4	$72 \times 24 = 1728$
7	126	7+ 3	$126 \times 12 = 1512$
8	240	8+ 2	$240 \times 6 = 1440$
9	272	9+ 1	
10	> 336	10	

Table 4.1: *Different choices for quantization and its effect of the key space (maximum number of bits that can be embedded) for a 10-dimensional vector X. In the first column we have the number of dimensions that are quantized at a time, the second column gives the value of the kissing number for the chosen dimension. The third column gives the particular choice for grouping the subsets and the forth column shows the size of the key space.*

The result of *Corollary* 3, confirmed by our example shows that the best strategy for quantization is the two-dimensional quantization. As this result points us to two dimensions, we will give two practical constructions for the two-dimensional case in the next section.

4.5 Practical constructions in two dimensions

In this section we present two optimal constructions for the `QIM`-fuzzy embedder in the two dimensional space. The first, 7-hexagonal tiling, is optimal from reliability point of view while the second is optimal from the number of equiprobable keys it can embed and the sketch size. We choose a hexagonal lattice to represent reconstruction points for the `QIM`, since this gives both the smallest circle covering (for the Embed procedure) and the densest circle packing (for the Reproduce procedure).

The first construction, the *7-hexagonal tiling*, can embed $n \times \frac{\log_2 7}{2}$ bits, where n is the dimensionality of random variable X. This construction is optimal from the reliability point of view. However, in this construction keys are not equiprobable, when the background distribution is not flat enough. The second construction, the *6-hexagonal tiling*, fixes this problem, but achieves a slightly lower key length of $n \times \frac{\log_2 6}{2}$ bits.

In our constructions the reconstruction points of all quantizers are shifted ver-

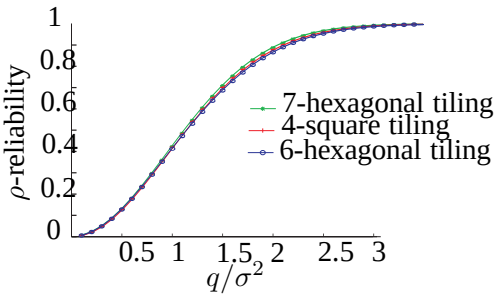


Figure 4.12: *Reliability of the three* `QIM`*-fuzzy embedder constructions.*

sions of some base quantizer Q_0. A dither vector $\overrightarrow{v_k}$ is defined for each possible $r \in \mathcal{R}$. We define the *tiling polytope* as the repeated structure in the space that is obtained by decoding to the closest reconstruction point. It follows from this definition that the tiling polytope contains exactly one Voronoi region for each quantizer in the ensemble. In *Figures* 4.10 and 4.11 the tiling polytopes are delimited by the dotted line.

The n-dimensional variable $X = (X_1, X_2, \cdots X_n)$ is partitioned into $\frac{n}{2}$-two dimensional subspaces (X_1, X_2). Each subspace is considered separately. On the x-axis in *Figure* 4.10 we have the values for X_1 and on the y-axis we have the values of X_2. Along the z-axis (not shown in the figure) we have the joint probability density $f_{X_1X_2}(x)$.

We start our construction by choosing the densest circle packing existing in the two dimensional space which is the hexagonal packing. All circles have equal radius and the center of the circle is the reconstruction point. With each reconstruction point a key value is associated. However, the circles do not tile the space. As a result when x, the realization of X, falls into the non-covered region, it cannot be associated with any reconstruction point. We need to approximate the circle with some polygons that tile the two-dimensional space. In the two dimensional space the Voronoi region for the hexagonal lattice is a hexagon.

In the two dimensional space, there are only three such regular polygons: triangles, squares and hexagons. Since we assume a spherical symmetrical distribution for $f_{X_1X_2}$ the hexagon is the best approximation to the circle from reliability point of view. The next step is to associate a key value to each hexagon such that for any value of (X_1, X_2), any key label is at most at the given distance (sphere covering problem).

4.5.1 7-Hexagon Tiling

Thus our first construction is a dithered `QIM` defined as an ensemble of 7 quantizers. The reconstruction points of the base quantizer Q_0 are defined by the lattice spanned by the vectors $\overrightarrow{B_1} = (5, \sqrt{3})q$, $\overrightarrow{B_2} = (4, -2\sqrt{3})q$, where q is the scaling factor of the lattice. In *Figure* 4.10 these points are labeled k_0. The other reconstruction points of quantizers $Q_i, i = 1, \ldots, 6$ are obtained by shifting the base quantizer by the dither vectors $\{\overrightarrow{v_1}, \cdots, \overrightarrow{v_6}\}$ such that $Q_i(x) = Q_0(\overrightarrow{x} + \overrightarrow{v_i})$. The values for these dither vectors are: $\overrightarrow{v_0} = (0, 0)$,$\overrightarrow{v_1} = (2, 0)$, $\overrightarrow{v_2} = (-3, \sqrt{3})$, $\overrightarrow{v_3} = (-1, -\sqrt{3})$, $\overrightarrow{v_4} = (-2, 0)$, $\overrightarrow{v_5} = (3, -\sqrt{3})$, and $\overrightarrow{v_6} = (1, \sqrt{3})$. The embed and reproduce procedures work as in our construction in section 4.4. The reproduce procedure is shown in *Figure* 4.10.

4.5.2 6-Hexagon Tiling

Assume that the background distribution is a spherical symmetrical distribution with mean centered in the origin of the coordinates. In the construction above the hexagon centered in the origin will typically have a higher associated probability than the off-center hexagons. This effect grows as we increase the scaling factor q of the lattice. This construction eliminates the middle hexagon, to make all keys equiprobable (see *Theorem* 1). The key length is $\frac{\log_2 6}{2}$ bits. The tiling polytope is formed by 6 decision regions and thus there are only 6 dither vectors, see *Figure* 4.11. The same dither vectors, $\{\overrightarrow{v_1}, \cdots, \overrightarrow{v_6}\}$ are used to construct the quantizers, but the basic quantizer Q_0 itself is not used. The embed and reproduce procedure are defined as in *Section* 4.4.

4.5.3 Performance Comparison

We compare the two constructions proposed above, i.e. the 7-hexagonal tiling (*Figure* 4.10), and the 6-hexagonal tiling (*Figure* 4.11), in terms of reliability, min-entropy of the key and entropy loss to the scalar quantization scheme introduced by Linnartz *et al.* [49] on each dimension separately (we refer to this as 4-square tiling).

To perform the comparison we consider identically and independently distributed (i.i.d.) Gaussian sources. We assume the background distribution has mean $(0, 0)$ and standard deviation $\sigma_{X_1X_2}$. Without loss of generality we assume that for any random $(X_1, X_2) \in \mathcal{U}^2$, the probability distribution of $f_{X_1X_2}(x)$ has mean $\mu = (\mu_1, \mu_2)$ and standard deviation σ_x^2. This model comes from biometrics, where the background distribution (also called imposter distribution) describes all users, and the user distribution is the distribution of random variable X.

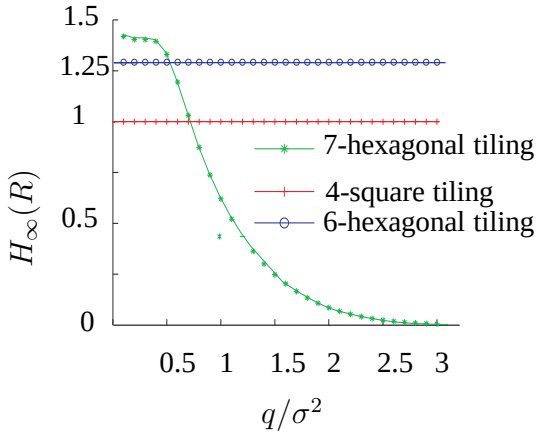


Figure 4.13: *Key length comparison for the three* `QIM`*-fuzzy embedder constructions- scaled to one dimension*

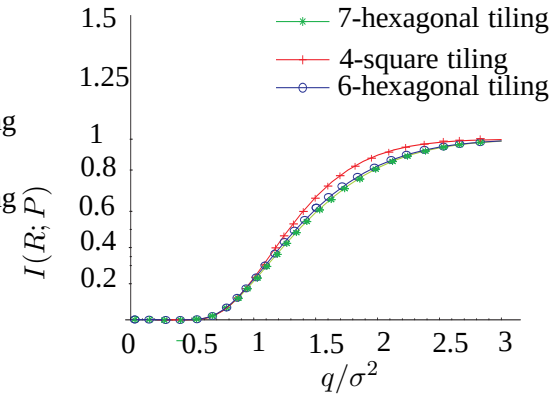


Figure 4.14: *Mutual information between the key and the public sketch for the three* `QIM`*-fuzzy embedders*

To evaluate the reliability relative to the quality of the source data (amount of noise, measured in the terms of standard deviation from mean) we compute probabilities associated with equal area decision regions, and the reconstruction point centered in the mean μ of the distribution $f_X(x)$. The curves in *Figure* 4.12 are obtained by progressively increasing the area of the Voronoi regions. The size of the Voronoi region is controlled by the scaling factor of the lattice, q. The best performance is obtained by the hexagonal decision regions. This is because the regular hexagon best approximates a circle, the optimal geometrical form for a spherical symmetrical distribution. However, the differences between reliability of the three `QIM`-fuzzy embedders are small.

The min-entropy in $r \in R$ is compared in *Figure* 4.13 among 7-hexagonal tiling, 6-hexagonal tiling, and 4-square tiling. Maximizing the min-entropy means minimizing the probability for an attacker to guess the key correctly on her first try. The key length for the 7-hexagonal tiling decreases rapidly with the increase of the lattice scaling factor q relative to $\sigma_{X_1X_2{}^2}$. While for a small lattice the scaling factor q one can approximate the background distribution as uniform, with the increase in scaling the center hexagon has a substantially higher probability associated with it, and thus one key value is more likely than all the others.

The 6-hexagonal tiling construction eliminates the middle hexagon and as a result all keys become equiprobable, at the cost of a somewhat lower reliability 4.12.

Finally, we evaluate the mutual information for the key when publishing the sketch for the three constructions compared. The results are shown in *Figure* 4.14. The values are scaled to the number of bits lost from each bit that is made public.

The results are somewhat surprising in the sense that the 4-square tiling looses more bits compared to our two new constructions. The reason is that while the size of the public sketch p is equal for all three constructions, thus they all lose the same amount of information but the key length differs.

4.6 Discussion: Putting it all together

A fuzzy extractor can transform a noisy, non-uniform discrete source of data, which is easily accessible into a reproducible, uniformly random string, which is suitable to be used as a cryptographic key. Basically, the fuzzy extractor performs two functions: the first is error correction, which compensates for the noise in the source data and the second is smoothing the non-uniform distribution of the source into a uniformly random distribution of the output.

When considering a fuzzy extractor construction in a practical scenario the two functions provided are not enough. Firstly, a fuzzy extractor is too limited because it accepts only discrete input data. Thus a procedure which transforms continuous data into discrete data is necessary. Our construction in *Chapter* 3, *cs*-fuzzy extractor is an extension of the fuzzy extractor construction in this sense. Secondly, a fuzzy extractor as pointed out by Boyen [19] needs to re-randomize its output such that one noisy source can be used in more than one application.

A typical fuzzy extractor implementation can be modeled as in *Figure* 4.15. In our view, there are four main building blocks: *quantization*, *error correction*, *randomness extraction* and *randomization*, which can be used in a typical fuzzy extractor implementation.

Each block in *Figure* 4.15 solves a specific problem and in the following we take a closer look at the purpose and requirements for each of the four blocks.

QUANTIZATION. The quantization block is used to transform continuously distributed data X with probability density function $f_X(x)$ into discretely distributed data Y with discrete probability density $f_Y(y)$. Examples of quantization schemes can be found in Chen *et al.* [28] and Zhang *et al.* [90] and in *Figure* 4.2. During quantization the public sketch denoted with P_1 in *Figure* 4.15 is computed and made public. The information leaked by the public sketch about the noisy source data is measured in terms of mutual information $I(X; P_1)$ between the source data X and the public sketch P_1.

ERROR CORRECTION. The error correction block adds redundant information to the input variable Y to increase the probability that its values are correctly repro-

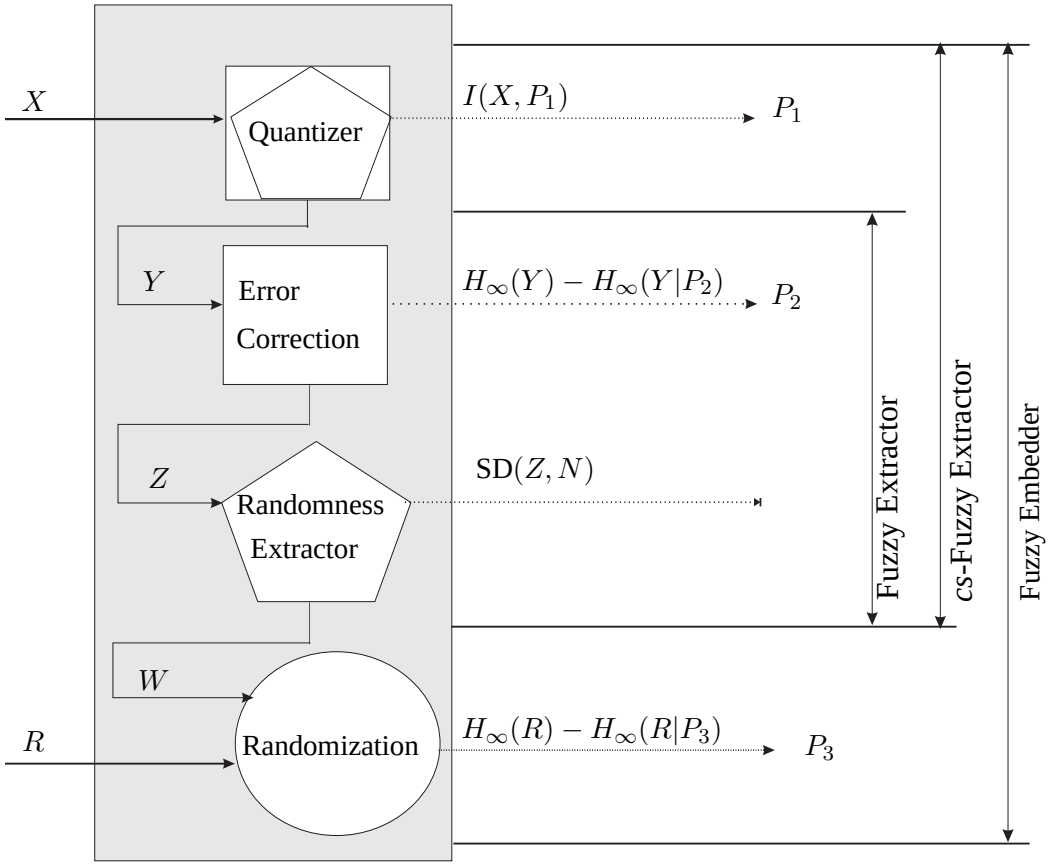


Figure 4.15: *Typical implementation of a fuzzy extractor. The shape of a block is a code for its purpose. Square blocks perform error correction, pentagonal blocks shape the distribution of the data, while the circle blocks are used to randomize the data. A fuzzy extractor can be constructed from an error correcting block and a randomness extractor. On the left-hand side of the figure the input variable (with capital letters, above the arrow). On the right-hand side of each block the security measure used to evaluate the performance of the block is presented.*

duced. The input variable $Y = (Y_1, Y_2, \cdots Y_n)$ is represented as a n-dimensional vector and its elements Y_i are called feature vectors.

There are two types of noise that can occur in Y. The first is *white noise* where elements of Y_i are perturbed by noise and the second is *replacement noise* where some features of Y can disappear and new features can appear between two consecutive measurements. Error correction schemes which correct white noise where proposed by several authors [28, 49, 90] while error correction schemes for replacement noise can be found in [79, 24].

To perform error correction a *public sketch* (also called *helper data*) is com-

puted for Y. If the helper data is made public, which is the case in most scenarios, it reveals information about the variable Y. The performance of an error correction scheme is measured in terms of how many *errors* it can correct and the amount of *leaked information*.

When the source data is continuous the leakage is measured in terms of mutual information, in *Figure* 4.15 $I(X; P)$, where P can be either P_1 or P_2. When the source data is discrete as in the error correction block in *Figure* 4.15 the amount of leaked information is measured in terms of min-average entropy $H_\infty(Y; P_2)$ and min-entropy $H_\infty(Y)$. The difference between the two is called the *entropy loss*.

RANDOMNESS EXTRACTOR. This block is used to transform *any* probability density function $f_Y(y)$ into a *uniform* probability function $f_Z(z)$, which is desirable for a cryptographic algorithm. A randomness extractor is used to "purify" the randomness coming from an imperfect source of randomness, it can efficiently convert a distribution that contains some entropy (but is also biased and far from uniform) Y into an almost uniform random variable Z.

The performance of a randomness extractor is measured in terms of the statistical distance between the distribution of the output variable Z and the distribution of a uniform random variable N, denoted in *Figure* 4.15 by SD(Z, N).

In the process of randomness extraction an external source of randomness must be present. Reducing the amount of required randomness in the external source and producing outputs, which are as close as possible to a uniform distribution is the main research topic in this area [14, 74, 75].

There are constructions known as *strong randomness extractors* [32] for which the output of the randomness extractor looks uniform even when the external source of randomness is made public, which are more convenient for the purpose for the scenario depicted in *Figure* 4.15.

RANDOMIZATION. This block is used to randomize the string which can be extracted from the noisy source. When biometrics is used as a noisy source, the purpose of randomization is protection of privacy for the biometric data. For example, from one fingerprint only one reproducible, uniform string can be extracted. The randomization ensures that from one fingerprint multiple random sequences, which can be used as cryptographic keys for more applications, can be produced.

We argue that the model described in *Figure* 4.15 covers most of the work done in the area of construction of cryptographic keys from noisy data. Theoretical work in the area usually covers the error correction block and randomness extraction [32, 33] whereas others, look at more practical aspects like quantization [5, 25, 28, 48, 90] or randomization [19, 20, 24].

The fuzzy embedder construction is intended as an all-encompassing theoretical model given the functionality of a fuzzy extractor. Thus, a fuzzy embedder is able to hide a key in any type of source data. This holistic view gives new insights in at least two ways:

- Not all four blocks in *Figure* 4.15 are necessary. For example the `QIM`-fuzzy embedder construction has only two blocks *quantization* and *randomization* [3].

- The order of the blocks in *Figure* 4.15 can be changed. Thus, the overall performance of the construction can be enhanced, e.g. using one error correction block instead of two error correction blocks, might limit the amount leakage.

Most of the work in the area of cryptographic use of noisy data focuses on optimizing one aspect, e.g quantization, randomness extraction, etc. Security measures used to quantify the performance in each block are different as they are studied in different research areas. In a practical scenario, when all these blocks are needed it is important to have an overall view of all the information that is leaked or the amount of errors that are corrected. The main purpose of the fuzzy embedder is to put things in perspective and define the overall security measures.

4.7 Conclusions

We propose the notion of a *fuzzy embedder* as a generalization of a fuzzy extractor. Fuzzy embedders solve two problems encountered when fuzzy extractors are used in practice: (1) a fuzzy embedder naturally supports renewability, and (2) it supports direct analysis of quantization effects. This is made possible by embedding a key instead of extracting one, and by making no limiting assumptions about the nature of the input source.

We give a general construction of a fuzzy embedder, using a `QIM` to construct the Embed and Reproduce procedures. The `QIM` performance measures (from watermarking) can be directly linked to the reliability and security properties of the constructed fuzzy embedder.

This construction gives a deep insight in the trade-offs between the parameters of a fuzzy embedder. We describe the key length-entropy loss tradeoff as a simultaneous sphere-packing / sphere-covering problem and we show that when considering equiprobable keys, quantizing dimensions pairwise gives the largest key length.

We also give two explicit, two-dimensional constructions, which can embed a longer key per dimension than existing (one-dimensional) schemes. The 7-hexagonal tiling scheme achieves the optimal probability of detection, but only performs well if the underlying background distribution is flat enough. We show that our 6-hexagonal tiling scheme is optimal from a key length perspective, given that each key is equiprobable. Using the 6-hexagonal construction we obtain $\frac{log_2 6}{2}$ bits per dimension of the input data, which is superior compared to the single bit obtained by the shielding scheme.

The contribution of this chapter is related to the problem of SECURE TEMPLATE PROTECTION. We propose a new, holistic model, the fuzzy embedder, which encompasses both the theoretical clarity and the practical needs of a template protection scheme. In the next chapter we use the fuzzy embedder, as a basic building block for secure pairing protocol, which is our solution for the SECURE TEMPLATE TRANSFER problem.

Chapter 5

Secure Pairing with Biometrics: SAfE

The *pairing problem*, described in *Chapter* 2 as SECURE TEMPLATE TRANSFER, is to enable two devices, which share no prior context with each other, to agree upon a security association that they can use to protect their subsequent communication. Secure pairing should offer guarantees of the association partner's identity and it should be resistant to eavesdropping or to a man-in the middle attack. We propose a user friendly solution to this problem. Keys extracted from images of the participants using the fuzzy embedder are used for authentication. Details of the SAfE pairing system are presented along with a discussion of the security features and a usability analysis.

Mobile devices are designed to interact anytime, anywhere. In many scenarios, however, is it desirable to associate devices in a secure way. For example when using a mobile phone to pay for tickets or when sharing private contact information via the wireless link in an unsecured environment. This problem is known in the literature as secure device association [46]. Solutions have to be specifically designed such that secure association can be realized between previously unassociated devices. Security means that the solution must offer guarantees of the association partner identity and must be resistant to eavesdropping and to a man-in the middle attack. The ideal solution must provide a balance between security and ease of use.

SCENARIO. When two users, Alice and Bob, meet at a conference and decide to exchange business cards or other documents, they talk for a while until they trust

each another sufficiently to exchange information. However, they do not wish other participants to eavesdrop on their communication or to tamper with their documents. At this stage the only secure association that they have is their trust in each other. To set up a secure association between their devices a protocol is needed that can transfer this trust to their devices. It is not enough for Alice's device to guarantee a secure pairing with device: 128.196.1.3. Alice needs to know that there is a secure association with Bob. Kindberg, *et al*. [46] use the term physical validation for physically verifying the identity of the other party in an association. For example when two devices are connected via a cable or an infrared channel. Kindberg [46] sees the physical validation as the physical counterpart of cryptographic authentication of identity. The strength of the physical validation depends on the length of the key established after pairing. Our solution is a protocol that can transfer the trust relation between people to a trust relation between devices using biometrics as the main tool, offering strong physical validation.

USER FRIENDLINESS. The most important reason why security often fails is the lack of user friendliness. To establish a secure communication, Alice and Bob have to agree on a key. From a usability point of view we want Alice and Bob to have minimal interaction with their devices, and the technical difficulty of the required task should be no worse than to dial a number on a mobile phone. Also we do not like the idea of Alice and Bob having to remember a password or a pin code for establishing the communication key. A user friendly solution is readily provided by appropriate use of biometrics, since a fingerprint or the image of a face is readily available, and has the advantage that it cannot be lost or forgotten and is thus always available.

CONTRIBUTIONS. We present a practical solution to the secure device association problem where biometrics are used to establish a common key between the pairing devices. Our approach has at least two major advantages. Firstly, it offers the possibility to transfer trust from humans to machines without any available security infrastructure. Biometric recognition offers physical validation, thus guaranteeing the identity of a device owner. Secondly, the process is short and should be user friendly. We propose a protocol in which the keys extracted from biometric data are combined to form a session key. The idea is both simple and effective. Suppose that two users wish to set up a secure communication channel. Both own a biometrically enabled handheld device. Both devices are equipped with a biometric sensor (a camera for face recognition) and a short range radio. Each device is capable of recognizing its owner. Then the users take each others picture. Each device now contains a genuine template of its owner and a measurement that approximates the template of the other user. The idea is that each device

calculates a common key from the owner template and the guest measurement. In our solution for Alice to set up a secure communication with Bob, she has to take a picture of him and let Bob take a picture of her. The protocol is even more general: it can be applied on any type of biometric channel. Our protocol is innovative compared to a key exchange protocol in the sense that legitimate users have to "find" the communication key by performing a related key search attack. The advantages are twofold. Firstly, fuzzy extractors can create a repeatable sequence out of biometric and our key search mechanism helps lower the error rates of the fuzzy extractor in a practical situation. Secondly, the key search mechanism uses the unpredictable randomness between two measurements as a random salt for the session key thus strengthening the key. The disadvantage is that key search takes time which leads to a trade-off between performance and security.

ROAD MAP. We start with a description of related work in section 5.1 to put the contribution of this chapter into perspective. Section 5.2 gives general background information regarding the particularities of the two biometric systems used later and describes the notation used in the rest of the chapter. Extracting keys from biometric data is an entire research field on its own; we dedicate section 5.3 to summarize the main results from this topic. In this section we describe how a reliable, uniformly random sequence can be extracted from noisy data such as biometrics highlighting the tradeoffs that have to be made and we give two examples that can be used in a practical setting. Section 5.4 is dedicated to the pairing protocol. In section 5.5 we look at security properties achievable against two powerful adversaries Eve and Charlie. Eve is an eavesdropper. She can record messages sent between Alice and Bob and try to find the key used to secure their messages. The other adversary, Charlie cannot search for the key but he has complete control over the communication environment so that he can listen, or modify any message. These two adversaries correspond to two different but complementary views on security: computational security and formal security. In section 5.6 we validate our protocol by experiments on real life biometric data. We look at two different flavors of biometric recognition: hand grip pressure pattern recognition and face recognition. Results obtained from these experiments are promising. Results of a usability study regarding the secure device association using face recognition are presented in section 5.7. Finally conclusions are presented in section 5.8.

5.1 Related work

Saxena, *et al*. [68] define the *pairing problem* as enabling two devices that share no prior context, to agree upon a security association that they can use to

protect their subsequent communication. Pairing is intensively studied in the area of pervasive and mobile computing. Most protocols for secure spontaneous interaction rely on two channels to perform the pairing process. The first, in-band channel, has high bandwidth but no security properties while the second, out-of-band, channel has limited bandwidth while offering additional security properties. There are two approaches in performing secure device association. The first approach uses the out-of-band channel to verify keys exchanged on the in-band channel with human assistance. We call this approach out-of-band verification. The second approach uses the out-of-band channel to send a secret but small message from which the common communication key is then derived and then the key is verified on the in-band channel. We call this approach in-band verification. Different flavors of out-of-band channels have been proposed that depend on the available hardware equipment, achievable bandwidth, offered security properties and requirements for user interaction with the devices. We summarize the history and evolution of the most well known out-of-band channels.

Stajano, *et al*. [73] brought the secure device pairing problem to the attention of the research community. They propose to use physical interface and cable as the out-of-band channel. The physical channel has a high bandwidth and offers confidentiality, authenticity and integrity. It is, however, impractical since all possible physical interfaces have to be carried around at all times.

Balfanz, *et al*. [13] propose to use a physically constrained channel (e.g. infrared) to establish a secure association between devices in close proximity. They advanced the state of the art by eliminating the need to carry around all the bulky interfaces. However, the disadvantage of this approach is the infrared channel which is slow, and which requires line-of-sight.

Bluetooth users can pair devices by introducing the same PIN, usually a 4 digit number in the paired devices. Shaked, *et al.* [70] show how a passive attacker can find the PIN used during pairing. The randomness and length of the PIN number influences the speed with which an attacker can perform this attack (a 4 digit PIN is cracked in less than 0.3 seconds). To make things worse Uzun, *et al.* [82] note in a usability study performed on different strategies for pairing that the choices of PIN numbers are not really random. We make the same observation in section 5.7.

McCune, *et al.* [55] propose to use the visual channel as an out of band channel. In their protocol, called *Seeing is Believing* (SiB), devices send their public key on the in-band channel while displaying the hash of the public key as a bar code. If the devices have no display, a sticker is suggested for displaying the hash of the public key. If mutual authentication is required both devices should have a camera to photograph the bar codes. SiB does not rely on the human ability to recognize the bar keys. Saxena, *et al.* [68] propose a variation of the SiB protocol which achieves secure pairing if one device is equipped with a light detector.

Goodrich,*et al.* [39] propose a human assisted authentication audio channel as the out-of-band channel. They use a text-to-speech engine for vocalizing a sentence derived from the hash of a device's public key.

Mayrhofer, *et al*. [54] propose accelerometer based authentication. Devices that need to be securely associated are shaken together and cryptographic keys are generated from data recorded by the two accelerometers. This approach is different from previous solutions in two ways. The first difference is that accelerometer data is used to produce cryptographic keys and the second difference is that the out-of-band channel is used to share the data from which keys are generated and not to authenticate keys. They report a key length obtained from accelerometer data between 7-14 bits for every second of shaking. By shaking longer the entropy may be increased.

We take a similar approach in the sense that cryptographic keys are transferred on the out-of-band channel. We propose to use biometrics as an out-of-band channel. The main advantage of biometrics over accelerometer data is the higher bandwidth that can be achieved, this can establish a key of length up to 60 bits (when we use face recognition biometrics) or 80 bits (when using hand grip pressure pattern biometrics).

5.2 Preliminaries

In this chapter we refer to two different biometric systems the first one uses face recognition. Face recognition analyzes the characteristics of a person's face image taken with a digital video camera. It measures the overall facial structure, including distances between eyes, nose, mouth, and jaw edges. The second biometric system is a hand grip pressure pattern where the image of the pressure pattern exerted while holding an object can be used to authenticate or identify a person.

We assume biometric measurements of a user to have a multivariate Gaussian statistical model. For face biometrics the number of elements of a feature vector, (N in our notation) can range between 30 features to about 280 features [38] while for hand grip pressure pattern N is equal to 40 features [85].

According to the statistical model a user is specified by a mean vector $t = (t_1, t_2, \cdots, t_N)$, termed in the rest of the chapter as the template and a standard deviation vector $\sigma = (\sigma_1, \sigma_2, \cdots, \sigma_N)$. By $x = (x_1, x_2, \cdots, x_N)$ we denote a noisy measurement. Due to differences in environmental conditions and user behavior (e.g. changes in the pose for face recognition or the presence of a ring for the hand grip pressure pattern) we expect that each x_i can be perturbed by a small amount of noise respective to t_i. The amount of noise depends on the value

of the standard deviation σ_i. If σ_i is small then we expect the difference between x_i and t_i to be small on the other hand if the value of σ_i is large then we expect the difference between x_i and t_i to be large as well.

The error rates of a biometric system are determined by the accuracy with which the matching engine can determine the similarity between a measured sample x and the expected value of the template t. We construct two hypotheses: **[**H_0**]** *x and t are sampled from the same probability distribution*; and **[**H_1**]** *x and t are not sampled from the same probability distribution*; The matching engine has to decide which of the two hypotheses H_0 or H_1 is true. To express the accuracy of a biometric system the terms *false acceptance rate,* FAR and *false rejection rate,* FRR are used. The *false acceptance rate* represents the probability that H_0 will be accepted when in fact H_1 is true. The *false rejection rate* represents the probability that the outcome of the matching engine is H_1 but H_0 is true.

5.3 Cryptographic keys from biometrics

Our protocol requires the construction of keys from biometric data. In raw form, biometric data is unsuitable to be used as cryptographic key material for two reasons. The first is its representation, usually the continuous real domain while cryptographic keys are represented in the discrete domain. The second reason is noise. Two consecutive biometric samples of the same individual will differ by a small, but unpredictable amount of noise while a cryptographic key should be exactly reproducible.

Chapters 3 and 4 of this thesis consider in detail the problem of extracting uniform and reproducible strings from noisy, non-uniformly distributed data. Dodis, *et al*. [32] propose a general construction termed fuzzy extractor, which in principle does two things: provides error correction to compensate for the unpredictable noise in the biometric and smoothing the non-uniform representation of biometric data.

There are two main components in a fuzzy extractor scheme: the generate and the reproduce. The generate procedure is used during enrolment (*Figure* 5.1 left) of a user X. As input it takes a low noise template t (for instance obtained by taking multiple low-noise measurements and averaging) of the biometric feature vector and a binary string $m = (m_1, m_2, \cdots, m_N)$ (which will be used as a cryptographic key later on), to compute the public sketch $w = (w_1, w_2, \cdots, w_N)$.

The binary string m can be extracted from the biometric data itself [76] as modelled by the fuzzy extractor or it can be generated independently [49] as modelled by the fuzzy embedder of *Chapter* 4. During authentication (*Figure* 5.1 right), the reproduce procedure takes as input a noisy measurement x of the user's

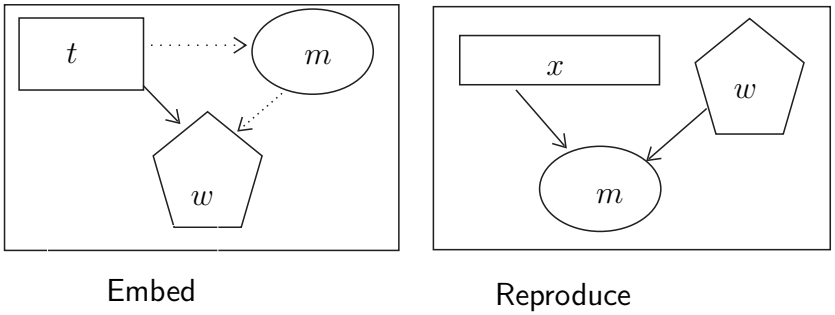


Figure 5.1: *A fuzzy embedder is a two step construction. The first step is the* Embed *procedure which is executed once when the device learns the identity of its owner. The second step is the* Reproduce *procedure which is executed each time a secure pairing is performed.*

biometric identifier(e.g. a photograph of the user for face biometrics) together with the public sketch w, and outputs the binary string m if the measurement is close enough to the original biometric. The exact reproduction of the binary string m is required to authenticate user X.

There is an important difference between creating the binary key m from biometric data (using a fuzzy extractor) versus creating the binary key independently of the biometric data (using a fuzzy embedder). In the first case the same sequence is extracted from the same biometric while in the second case different sequences can be embedded into the same biometric data for different protocol rounds. In our construction we prefer the second option, which is the fuzzy embedder because if the binary key is somehow compromised it is difficult to change the key, because this would mean changing the biometrics, i.e. changing one's face of fingerprint.

Both these algorithms operate componentwise on the feature vector. In other words, the noisy measurement will be processed to a feature vector $(x_1, \ldots, x_N)$. From each x_i and w_i the reproduce procedure outputs a binary string m_i (generally consisting of 0-3 bits). In particular, this means that even if some failures occur when processing the complete feature vector, the resulting bit string will still be close to the correct one. Later we show how this property can be used to improve the overall performance of a fuzzy embedder construction.

Two parameters are important for a fuzzy embedder construction. The first is the *reliability*, which represents the amount of noise tolerated between two measurements x and x' such that m is correctly computed by the reproduce. The second is security, which is determined by the *key length* (the length of m in bits) and the *entropy loss* [32], which measures the advantage that w gives to an adversary in guessing m. We require a fuzzy embedder to have long keys, high reliability and high security (i.e. low entropy loss). However, these are conflicting require-

ments. Usually the more secure (long key or small entropy loss) the less reliable (high values for the error rates FAR and FRR) the fuzzy embedder becomes.

The key length depends on the number of features available. The number of features is a function of the users enrolled in the system and the quality of the measurements. If there are N users in the system the maximum number of features that can be extracted is $N - 1$. However, if the collected data has poor quality the number of used features can be much less then the theoretical limit.

In the following we give two examples of fuzzy embedder schemes to illustrate how one can balance the reliability and key length in a practical setting.

As the first example let us consider the reliable components scheme of Tuyls, *et al*. [76] with security parameter s. This scheme assumes that a global estimate of the mean t is known. Enrollment is performed by taking s measurements of the user's biometric identifier. If the component i of each of those measurements is always bigger than a chosen threshold μ_i, we set $m_i = 1$. Otherwise, if all measurements are smaller then μ_i, we set $m_i = 0$. In all other cases, the component is not used. The public sketch w_i is set to 0 or 1 according to whether the component is used or not.

While the reliable component scheme described above achieves a high reliability, it may result in keys that are too short. Whether or not this method is satisfactory will have to be decided according to the intended use scenario. If a longer key is required, one should look at other fuzzy embedder constructions that embed one (or even more) bit(s) per component of the feature vector, like the schemes proposed by Chang, *et al*. [25]. However, a higher embedding rate does not come for free - it raises the FRR, or the longer key may not even have more entropy than the short one, meaning that it actually does not offer more security despite its greater length [5].

As second example we give the fuzzy embedder interpretation of the scheme proposed by Linnartz, *et al.* [49] known in the literature as the shielding scheme. The Linnartz construction is one of the first fuzzy embedder constructions that works on continuously distributed data as required for biometric data and is a particular case of the general `QIM`-fuzzy embedder construction proposed in *Chapter* 4. They propose to divide the probability density function of each feature component in odd-even bands of equal length q and label the odd-even bands with 1 and the even-odd bands with 0. The embedding of binary data is done by shifting the template distribution mean t_i to the center of the closest even-odd q interval if $m_i = 0$, or to the center of an odd-even q interval if $m_i = 1$. The public sketch w_i is the difference between the location of the mean t_i and the center of the chosen q interval, see *Figure* 5.2. During authentication the measurement x_i is shifted by the value of the public sketch w_i and the label of the corresponding interval is output. We describe this construction further in section 5.4.3. In the

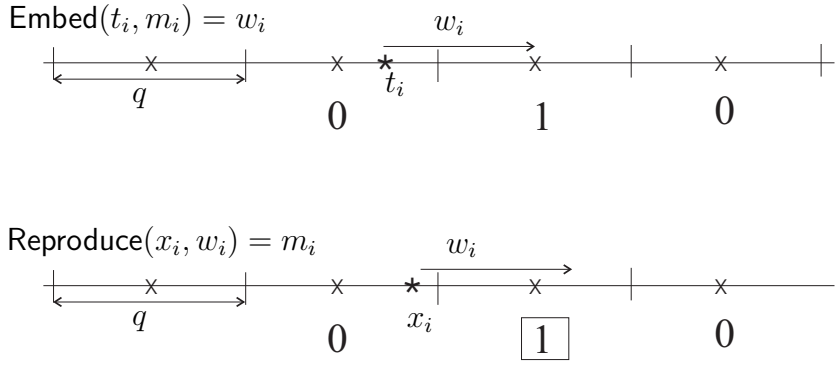


Figure 5.2: *The Linnartz* et al. *[49] fuzzy embedder for continuously distributed data. For embedding a bit $m_i = 1$ the* Embed *procedure outputs the public sketch w_i which is the difference between the template t_i and the closest middle of a 1 interval; The* Reproduce *procedure adds the measured x_i to the public sketch w_i and outputs the label of the result, in this case 1.*

shielding scheme construction the key length is fixed beforehand. More precisely it is equal to the number of features of the biometric template. A trade-off can be made between reliability and entropy-loss by varying the quantization step q [49].

The main difference between the reliable component scheme of Tuyls, *et al*. [76] and the shielding scheme of Linnartz, *et al.* [49] is the way the cryptographic key is generated. In the first case the biometric key is extracted from the biometric data whereas in the second case the cryptographic key is generated independently. The biometric data is used to unlock the value of the pre-generated cryptographic key. Thus, if the scheme is compromised a new key can be generated for the same biometric. That is our reason for choosing the shielding scheme in this work. As a conclusion, the properties of the biometric data and the selection of the embed and reproduce procedures determine the quality (in terms of randomness) of the cryptographic material that can be extracted from it. In the following we explain the authentication protocol and we analyze the impact of the key quality on the security of the protocol.

5.4 SAfE protocol

The SAfE protocol establishes a shared secret key between devices whose owners happen to meet and who have no prior security association. There are three phases in the lifetime of our protocol. The first (past), is the enrolment which can be regarded as a necessary precondition. The second (present), is the

SAfE protocol which is the action taken by Alice and Bob to achieve their goal which is secure communication (future) the third and final phase. We detail these phases below.

1. *Enrolment,* is performed once in the lifetime of the protocol. This step is performed by both participants Alice and Bob, independently, for example at home, and it is performed once. Each participant takes multiple (low-noise) measurements of his own biometric, and uses these to calculate the biometric template vector t. Next, each participant picks a random string m, and uses the embed procedure of the fuzzy embedder to calculate the matching public sketch w. To differentiate between the participants we use t_A, m_A, w_A for the template, key and public sketch of Alice and t_B, m_B, w_B respectively for Bob. After enrolment we have achieved that: (1) the identity of a user can be verified by her own device, and (2) a device is prepared to be paired up with another device on which the SAfE protocol has been implemented.

2. *Pairing,* is performed each time the users meet. The SAfE protocol is used to create a secure channel, a secret key is computed by the reproduce procedure of the fuzzy embedder. The protocol description below provides all the details of this step.

3. *Secure communication,* when the paired users send messages, documents etc. encrypted with the key derived by the SAfE protocol.

5.4.1 SAfE protocol details

The SAfE protocol uses two communication channels for key establishment as in the pairing model proposed by Balfanz, *et al.* [13]. One, the in-band channel, is used for authentication. This channel has a high bandwidth but offers no security guarantees. The second is the out-of-band channel used for pre-authentication. This channel has a low bandwidth but offers security guarantees like authentication, integrity and/or confidentiality. In the SAfE protocol we use the out-of-band channel to exchange a limited amount of information. Later, we use this information to establish a common key by exchanging messages on the in-band channel.

OUT-OF-BAND CHANNEL. In the SAfE protocol we use biometrics as the out-of-band channel. The first reason for our choice is that biometrics is a source of high entropy data which means high bandwidth compared to other out-of-band channels (e.g. infrared). The type and quality of the biometric modality used

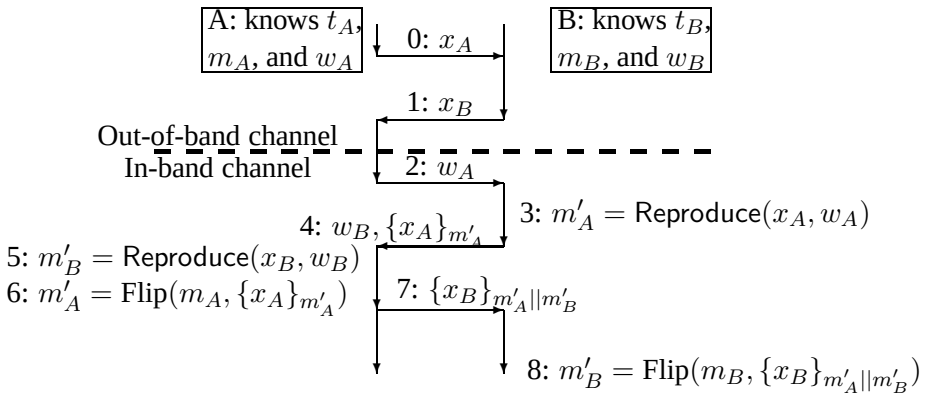


Figure 5.3: *Message flow for the SAfE protocol showing the steps taken by Alice to the left (5,6) and Bobs actions to the right (3,8) to pair their mobile devices. The steps in the middle represent the message exchange on the out-of-band channel (0,1) and the in-band-channel (2,4,7).*

(fingerprint, face, iris, palm print) determines the value of the bandwidth capacity for the out-of-band channel. We analyze, in section 5.6, the performance of two different biometric modalities: face and grip pressure pattern.

The second reason for biometrics as an out-of band channel is that it is easy to send messages on this channel since the main characteristic of biometrics is user friendliness (see section 5.7 for the results of usability analysis when face recognition biometrics is used as the out-of-band channel).

The security properties of the out-of-band channel depend on the properties of the biometric used. By default, biometric authentication offers authenticity and integrity. It offers authenticity because we know the source of the message and integrity since the message collected by Alice on the out-of-band channel cannot be changed by a third party. For some biometrics, like hand grip pressure pattern, retina or ear recognition we may even assume channel confidentiality because it is difficult for an adversary to collect a sample of the biometric without the user noticing. We discuss the implications of the properties of the out-of-band channel on the security guarantees of the SAfE protocol in section 5.5.

IN-BAND CHANNEL. The in-band channel is a broadcast channel (e.g. WLAN) thus all messages sent on this channel are public and can be manipulated.

MESSAGE FLOW. The message flow of the SAfE protocol is presented in Figure 5.3. Without loss of generality we may assume that Alice starts the protocol.

Figure 5.4: *Data transferred on the out-of-band channel for face recognition biometrics.*

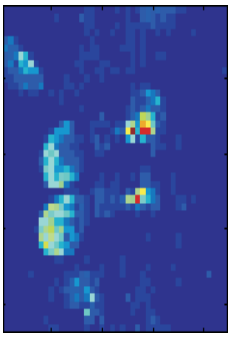


Figure 5.5: *Data transferred on the out-of-band channel for hand grip pressure pattern recognition biometric.*

We explain each of the steps:

0: Bob measures Alice's biometric. This is shown as a transfer of the measurement x_A from Alice to Bob on the biometric channel.
1: Similarly Alice takes a measurement of Bob's biometrics, yielding x_B.
2: Alice broadcasts her public sketch w_A on the wireless channel.
3: Bob feeds the public sketch w_A and the measurement x_A of Alice to the reproduce procedure of the fuzzy embedder to compute a key m'_A.
4: Bob broadcasts $w_B, \{x_A\}_{m'_A}$, i.e. the tuple consisting of w_B and the encryption of x_A using key m'_A.
5: Alice uses w_B received in plain in Step 4 and x_B received in Step 1 to compute m'_B with the reproduce procedure of the fuzzy embedder.
6: The second part $\{x_A\}_{m'_A}$ of the message is used to compensate for possible errors in reproducing m_A. We expect that due to noise or poor quality of the biometric sensor $m_A \neq m'_A$: However, due to their construction m_A and m'_A are close in terms of the Hamming distance so that Alice can perform an efficient key search algorithm to obtain m'_A from m_A. The key search algorithm systematically flips bits in m_A until $\{x_A\}_{m'_A}$ can be decrypted successfully (see the key search algorithm below for details). Since Alice can recognize a measurement of her own biometric, she can check the decryption results.
7: Alice broadcasts $\{x_B\}_{m'_A||m'_B}$.
8: Bob also performs a key search, flipping bits in the concatenation of m'_A and m_B until x_B can be decrypted successfully.

The action on the out-of-band channel "Bob takes a measurement from Alice" can be translated to: "Bob takes a picture of Alice" when face recognition biometric is used. In this case x_A represents the picture of Alice while x_B represents the picture of Bob (see *Figure* 5.4). The same action translates to "Bob hands

his mobile device to Alice who holds it firmly" in the case of hand grip pressure pattern generating x_A a grip pressure pattern (see *Figure* 5.5).

5.4.2 Key search algorithm.

In classical symmetric cryptography to decrypt a message encrypted with a key m one must know m exactly. In particular, with a key m' that differs only in one bit from m, decryption will fail. The SAfE protocol uses this apparent disadvantage of symmetric key cryptography as an advantage: m' is used to form the session key. The noise of the measurements is used as random salt [88] for the session key. The key search algorithm makes it possible to recover m'. Before the algorithm starts we decide on how many trials we make to discover the key. If we set the error threshold to τ bits the algorithm will try out at most $\sum_{i=0}^{\tau} \binom{N}{i}$ combinations before key search failure is declared. Then the protocol has to be restarted or the user gives up.

Alice starts the key search by assuming there are no errors in m'_A, and uses m_A to try and decrypt the encrypted message received in step 4. If decryption fails Alice assumes that there is a one bit difference between m_A and m'_A and so on until she has tried all combinations, i.e two bits, three bits etc. Finally, when Alice reaches the limit on the number of trials she assumes that the key is coming from an intruder and aborts the protocol. The recovery of m'_A is a related-key attack [56]. When the value of m'_A is discovered, Alice can decrypt the message encrypted with m'_A and recognize x_A by comparing it to t_A. The comparison can be performed by a classifier based matching algorithm designed for this particular biometrics.

A slightly less secure way is to use the reproduce functionality of the fuzzy embedder to recognize whether the decrypted result x is a measurement of Alice's biometric, by checking if $\mathrm{Dec}(x, w_A)$ is equal to m'_A. The advantage of this method is that the device does not need to store the sensitive template t_A, but only the (fixed) m_A and w_A. Since a fuzzy embedder is designed to correct errors in the (noisy) measurement, not for recognition, we expect this solution to be less secure since m_A is fixed for multiple protocol rounds. Bob performs the same search as Alice, but using m_B and m'_B.

We note that during the protocol both the devices of Alice and Bob have to perform the same amount of computation, which makes the protocol fair.

5.4.3 Smart Key Search.

When the key space is large the approach described above can become prohibitively expensive and unusable in practical situations. To increase the search

speed with which Alice finds m'_A from m_A we propose a method that computes weight coefficients on each of the key bits. The weight associated with a particular bit represents the probability of error for that bit. The vector of N weighting coefficients for a particular user is the *error profile*. The error profile gives, in fact the order in which bits are flipped. For example assume that 1 bit is changed in m'_A. Without error profile all N bits are equally likely to flip thus on average Alice will have to perform $\frac{N}{2}$ flips. On the other hand the error profile gives her the position of the most likely bit, giving an advantage.

There is another important reason for using error profile enhanced key search. Due to the nature of the protocol, Alice only has to find variations of her own key m_A and not keys coming from other parties. In particular, this means that we can reduce the false rejection rate without significantly increasing the false acceptance rate. We will see in section 5.6 how effective this approach can be.

The error profile computation is related to the specifics of the embed and reproduce procedure implementation. In the evaluation of our protocol we use the fuzzy embedder proposed by Linnartz, *et al.* [49] as described in section 5.3. To calculate the error profile we give the mathematical description of the embed and reproduce procedures below.

The public sketch is computed by the embed procedure as:

$$w_i = \mathsf{Embed}(x_i, m_i) = \begin{cases} (2n + \frac{1}{2})q - t_i & \text{when} \quad m_i = 1 \\ (2n - \frac{1}{2})q - t_i & \text{when} \quad m_i = 0 \end{cases}$$

Here $n \in \mathbb{Z}$ and is chosen such that: $-q < w_i < q$.

The reproduce procedure is defined as:

$$m_i = \mathsf{Reproduce}(x_i, w_i) = \begin{cases} 1 & \text{when} \quad 2nq \leq x_i + w_i < (2n+1)q \\ 0 & \text{when} \quad (2n-1)q \leq x_i + w_i < 2nq \end{cases}$$

ERROR PROFILE. Having described the fuzzy embedder above we remind the reader that embedders are not perfect, particularly because during key generation whenever the distance between the measured x_i and the expected t_i is larger than $\frac{q}{2}$ an error appears. The probability of an error is the probability of a measurement falling outside the chosen odd-even (labeled 1) or even-odd (labeled 0) interval of length q.

Figure 5.6 shows a feature with a normal distribution $N(t_i, \sigma_i)$ when the chosen interval is a 1. During encoding the public sketch w_i shifts the mean of the distribution to the closest 1 interval. The probability of error is then close to the probability of a measurement x_i shifted with the same w_i (the reproduce operation) falling in the neighboring 0 intervals, represented in *Figure* 5.6 by the hatched area. The error probability for this feature is computed as follows:

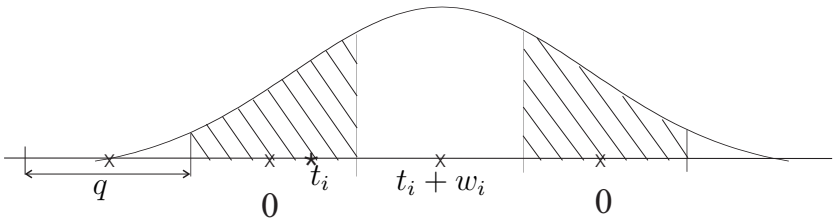


Figure 5.6: *Error computation for a feature element with normal distribution $N(t_i, \sigma_i)$, with quantization step q.*

$$\begin{aligned} E_i(\sigma_i, q) &= \sigma_i \, 2\sqrt{2} \sum_{j=0}^{\infty} \int_{\frac{(1+4j)}{2\sqrt{2}} \frac{q}{\sigma_i}}^{\frac{(3+4j)}{2\sqrt{2}} \frac{q}{\sigma_i}} e^{-x^2} dx \\ &> \sigma_i \, 2\sqrt{2} \int_{\frac{q}{\sigma_i \, 2\sqrt{2}}}^{\frac{3q}{\sigma_i \, 2\sqrt{2}}} e^{-x^2} dx. \end{aligned}$$

Here the integral represents the probability associated to one of the 0 labelled intervals of length q (one of the crosshatched intervals) and the summation is done over all the 0 intervals. If q is large enough we can approximate the error as being mostly determined by the two neighboring 0 intervals. Regardless of the chosen 0 or 1 labelled interval the error probability is computed exactly the same.

The error profile is the error probability of all N features of the template t.

In *Figure* 5.7 we show the error profile for the first 20 features computed on hand grip pressure pattern biometric data for two users named Alice and Bob. We can see that different users have different error profiles.

KEY SEARCH WITH ERROR PROFILE. When the template t and measurement x belong to the same user we expect a small number of errors to appear during the reproduction procedure. This means that even if m_A and m'_A are different, the difference should not be more then a few bits which can be further corrected using the error profile $e_A = (E_1(\sigma_1, q), \cdots, E_N(\sigma_N, q))$.

Now, the Flip function from *Figure* 5.3:

$$m'_A = \text{Flip}(m_A, \{x_A\}_{m'_A})$$

can be refined as:

$$m'_A = \text{SmartFlip}(m_A, \{x_A\}_{m'_A}, e_A).$$

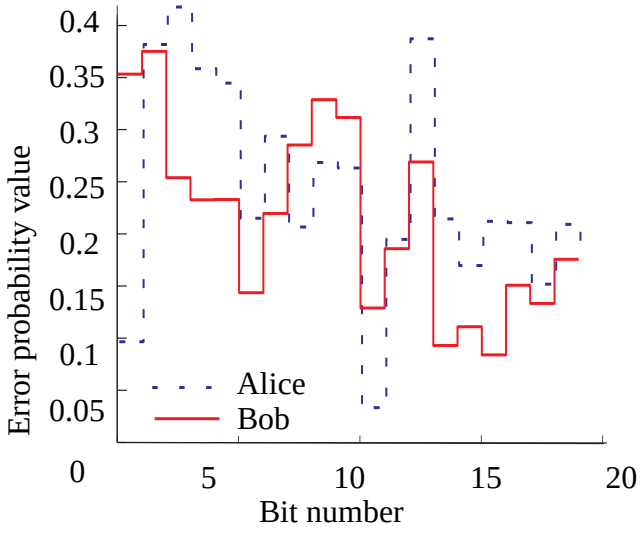


Figure 5.7: *Error profiles computed for Alice and Bob.*

We start the key search by assuming that there are no errors in m'_A, and we use m_A to decrypt the message $\{x_A\}_{m'_A}$. If decryption fails we assume there is a one bit error. We start flipping one bit of the key according to the position indicated by the largest component of e_A. If the operation is not successful we assume that two bits are wrong and we try combinations of the highest two components from the error profile. Finally if we reach the limit on the number of trials we assume that the key is coming from an intruder and the protocol is aborted.

5.5 Security Analysis

There are two distinct, rigorous views of cryptography that have been developed over the years. One is a formal approach where cryptographic operations are seen as black box functions represented by symbolic expressions and their security properties are modeled formally. The other is based on a detailed computational model where cryptographic operations are seen as strings of bits and their security properties are defined in terms of probability and computational complexity of successful attacks. In the following we look at both aspects of security to analyze the vulnerability of the protocol to two very different adversaries.

The first adversary, named Charlie is a Dolev-Yao [34] intruder who has complete control over the in-band communication channel. He can listen to, or modify messages on this channel. However, Charlie does not have computational capabilities. The actions of Charlie on the out-of-band channel depend on the properties of this channel. The second adversary, named Eve, is a passive adversary, i.e. eavesdropper. She can listen to the communication on the in-band channel and can perform a key search operation similar to Alice and Bob to find the communication key. If the out-of-band channel is not confidential she has access to a noisy version of the information sent on this channel. By modeling this adversary we try to answer the following question: "If both Alice and Bob have to guess the session key, how much more difficult is it for Eve to do the same?". We use the computational model to verify the vulnerability to an eavesdropper such as Eve.

From security point of view we realize that an adversary with the abilities of both Charlie and Eve is a potential threat and we should test the resilience of our protocol to such an adversary. Unfortunately as far as we know there is no formal approach that can handle such an attacker.

In section 5.5.1 we use the formal approach to verify the vulnerability of the protocol in the to a man-in-the-middle attack. This is an attack where Charlie is able to read, insert and modify at will, messages between Alice and Bob without either party knowing that the link between them has been compromised.

In section 5.5.2 we estimate how much effort is required for an adversary with the capabilities of Eve to find the common key established between Alice and Bob during a normal round of the SAfE protocol.

5.5.1 Formal verification (Charlie).

We have formally verified that SAfE satisfies mutual authentication and secrecy of messages exchanged after key establishment. The tool used for this purpose is the constraint based security protocol verifier CoProVe by Corin *et al.* [29]. An earlier version of the protocol was verified and found buggy, the version of the protocol in *Figure* 5.3 fixes the flaw found. A (security) protocol is normally verified using a model of the protocol, to avoid getting bogged down in irrelevant detail. The quality of the model then determines the accuracy of the verification results. The basic difference between a protocol and a model lies in the assumptions made when modeling the protocol. We believe that the following assumptions are realistic:

1. *No biometric errors.* We assume that the correction mechanism always works perfectly and thus the initiator knows the key used by the sender. Thus, we look only at complete protocol rounds. When the initiator cannot

work out the key the protocol is aborted. In this case we assume that Charlie does not get useful information from the aborted protocol messages.

2. *Modeling the out-of-band channel.* We have two types of out-of-band channels: (a) when hand grip pressure pattern biometric is used Charlie cannot listen, modify or send messages thus the out-of-band channel is authentic and confidential; (b) when face recognition is used Charlie cannot influence the picture Alice takes of Bob which makes the channel authentic. However, Charlie could himself take a picture of Bob. The picture Charlie takes of Bob will be slightly different from the picture Alice takes of Bob. Because systems without an equational theory such as CoProVe, do not have the notion of similarity we verify the protocol with the out-of-band channel in case (a) we leave this as future work. We assume that when the protocol starts Alice knows x_B the biometric of Bob and Bob has x_A the measurement of Alice biometric while Charlie knows neither.

We have verified the model in figure 5.3 with the assumptions above. We argue that the above abstractions do not affect the secrecy and the authentication property. Verification with CoProVe explores a scenario in which one of the parties involved in the protocol plays the role of the initiator (i.e. the party starting the protocol) and the other plays the role of the responder. A third party, the intruder learns all message exchanged by the initiator and the responder. The intruder can devise new messages and send them to honest participants as well as replay or delete messages. Should the intruder learn a secret key and a message encrypted with that key, then the intruder also knows the message.

Resilience to a man-in-the-middle attack depends on the assumptions made. Verification with CoProVe shows that the efforts of Charlie remain unrewarded when he does not have information about the biometric measurements x_A and x_B.

On the other hand if we assume that Charlie knows the biometric measurements of Alice and Bob, x_A and x_B respectively the protocol is broken. However, in real life this assumption is too strong since it is not possible to predict the noise in a biometric measurement and Charlie has no direct access to the measurements that Alice and Bob make. It is possible for Charlie to get an approximation of x_A and x_B. In the next paragraph we look at the security guarantees one can hope to achieve when the adversary knows some information about x_A and x_B but not all info.

5.5.2 Computational Analysis (Eve).

When the adversary has some useful initial knowledge as in the out-of-band channel case (b) we look at a different adversary, Eve. To derive keys from fuzzy

data we use a related-key attack in steps 6 and 8 of the protocol, to recover the session key. This approach raises two questions: "If both Alice and Bob have to guess the session key, how much more difficult is it for Eve (the intruder) to do the same?", and "What kind of guarantees is this protocol offering?" To answer these questions we study the following scenarios:

AE(0) No previous contact between Alice and Eve.

AE(1) Eve has a measurement of Alice's biometric. From the public string Eve constructs m''_A.

We denote by $W(x \to y)$ the average number of trials that Eve has to do to guess y when she knows x.

	AE(0)	AE(1)
BE(0)	$W(0 \to m'_A) \cdot W(0 \to m'_B)$	$W(m''_A \to m'_A) + W(0 \to m'_B)$
BE(1)	$W(0 \to m'_A) \cdot W(m''_B \to m'_B)$	$W(m''_A \to m'_A) + W(m''_B \to m'_B)$

Table 5.1: *Guesswork required for Eve to compute the session key.*

We analyze Eve's workload to guess m'_A in the two scenarios above. Alice (and the same holds for Bob) who knows m_A and who has to guess $m'_A = m_A + e$ where the Hamming weight of the noise e is $\mathtt{wt}(e) \leq \tau$, and where τ is an appropriate threshold. As the secret key length is N, there are $\binom{N}{i}$ different error patterns if the actual number of errors is i, thus on average Alice will have to guess (without knowing her error profile):

$$W(m_A \to m'_A) \approx \frac{1}{2} \sum_{i=0}^{\tau} \binom{N}{i}.$$

In scenario $AE(1)$, Eve knows m''_A and has to guess m'_A where $m''_A = m_A + e'$, thus $m''_A = m'_A - e' + e$. Since $\mathtt{wt}(e' - e) \leq 2\tau$, Eve has workload:

$$W(m''_A \to m'_A) \approx \frac{1}{2} \sum_{i=0}^{2\tau} \binom{N}{i}.$$

In scenario $AE(0)$ Eve has no information on Alice thus she has to brute force all possibilities. Thus the number of trials is approximately:

$$W(0 \to m'_A) \approx 2^{N-1}.$$

The scenarios for Bob are analogous:

BE(0) No previous contact between Bob and Eve.

BE(1) Eve records a measurement of Bob.

Eve's workload for guessing m'_B is equal to guessing m'_A in the analogous scenario.

To be able to listen on the communication channel Eve has to guess $m'_a||m'_b$ in all scenarios. Table 5.1 summarizes her workload. In each row we have the information that Eve knows about Bob and in the column the information that Eve knows about Alice. Due to the message flow in the protocol (see figure 5.3), Eve might have an advantage if she has information about Alice. Eve can intercept message 4: $w_B, \{x_A\}_{m'_A}$ and recover m'_A if the biometric allows for taking a decision on whether two measurements come from the same individual. This explains the plus sign between the work of guessing m'_A and the work of guessing m'_B in the columns where Eve has some knowledge about Alice. In the worst-case scenario, if Eve has had interactions with both Alice and Bob before, this means that Eve has to do a quadratic amount of work compared to either of the participants. In all other cases, there is at least one key that has to be recovered from scratch, making the attack infeasible.

We summarize why it is more difficult for Eve to guess the communication key compared to Alice and Bob:

- It is easier to start to guess $m' = m + e$ when m is available, as is the case for the legitimate participants Alice and Bob compared to guessing m' when $m'' = m' + e$ is available as is the case for Eve.

- A good quality camera for Eve will not improve her workload compared to a legitimate participant. Always Alice has as salt $m'_A = m_A + e_A$ while Eve will have $m''_A = m_A + e_E = m'_A - e_A + e_E$. With a good camera the best Eve can do is control e_E.

- Alice and Bob work in parallel to find the session key each computing their share while the best Eve can do is find the key sequentially, first find m'_A then find m'_B.

- Alice and Bob have an error profile that Eve does not have.

As a conclusion, the SAfE protocol can be assumed to be secure with respect to an eavesdropper for a short lived association as in the case with secure device association.

Figure 5.8: *Sample face images from FRGC database.*

5.6 Validation with real life data

We present experiments with two different sets of biometric data: hand grip pressure pattern data and face recognition data for validating the performance of the protocol. The goal of these experiments is to determine whether it is possible for Alice and Bob to determine their own key using the SmartFlip function knowing that biometric recognition is not perfect. We note that simulation results presented in this section were obtained in Matlab on real life data.

5.6.1 Face Recognition Biometrics.

For face recognition we report on three rounds of experiments on two different databases.

To verify the potential of constructing cryptographic keys from face data in the ad-hoc settings of our protocol we need a database with faces recorded with a mobile device. Since, as far as we know, such database is not publicly available we recorded our own "mobile" database. This database contains low-resolution images of 31 individuals, recorded in uncontrolled conditions. The first round of experiments was performed on this "mobile" database.

As a control for the first round of experimental results, we repeated the experiments on the Face Recognition Grand Challenge (FRGC) version 1 database

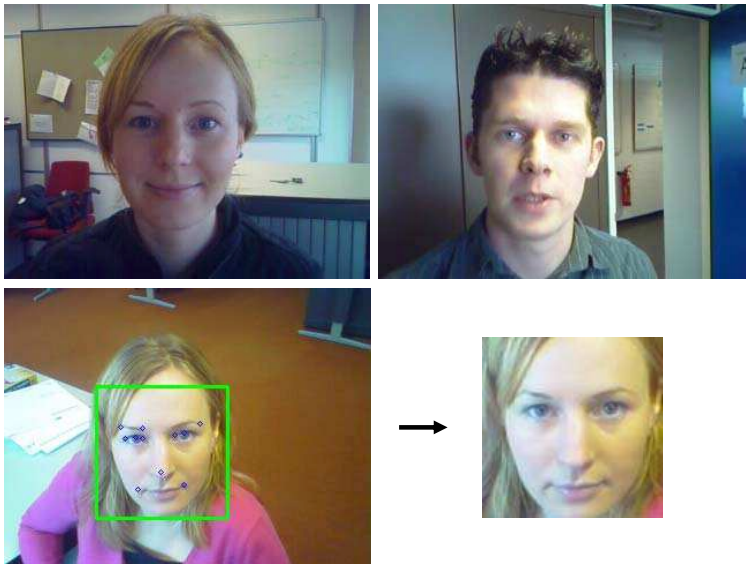


Figure 5.9: *Sample images from the mobile database.*

which contains 275 individuals. Images in the FRGC database are high resolution images, which can be divided into images obtained in controlled and uncontrolled situations. The difference between controlled and uncontrolled conditions can be seen in *Figure* 5.8 where the same person is captured in controlled conditions (right) and uncontrolled conditions (left). The second round of experiments was performed on the images taken in controlled situations and the third round of experiments was performed on the images taken in uncontrolled conditions.

One can see the three experiments as follows: the experiments with the mobile database show the success of Alice and Bob in performing the pairing protocol using face data recorded with existing mobile device technology. Experiments on the FRGC data set obtained in uncontrolled condition demonstrate the perspectives of the pairing algorithm in the near future when mobile devices are capable of capturing and processing high quality images. Experiments on the FRGC data set obtained in controlled condition represent the ideal case in terms of face recognition. One may hope to achieve them when changes in pose or in lighting conditions are no longer a problem.

MOBILE DATABASE. For each of the 31 individuals we recorded 4 video files using the same mobile device (ETEN M600+, which has a 2 mega-pixels camera). The four files were recorded in two sessions on two different days, each day we recorded two movies. On the first day each movie was approximately 10 seconds.

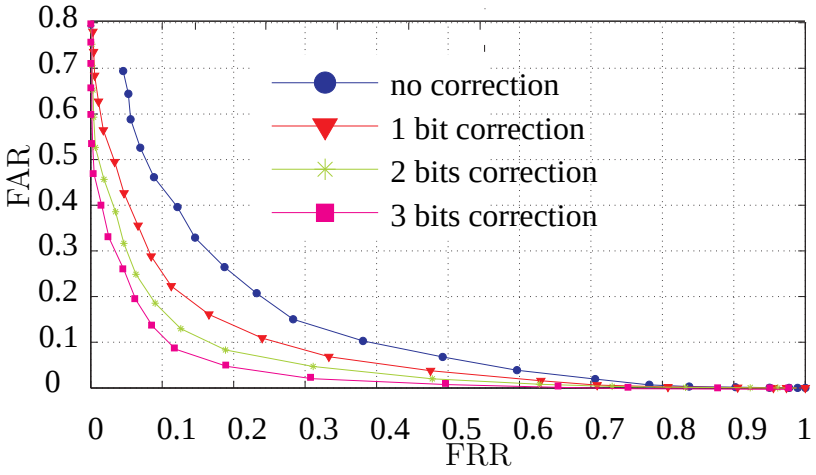


Figure 5.10: Experiment 1. ROC *curves for mobile data set, uncontrolled set.*

On the second day we recorded shorter movies of approximatively 5 seconds. Location of subjects (background), pose and light were different in the two sessions.

Also in this experiment we use the algorithm implementation proposed by Veldhuis *et al.* [84] for hand geometry and adapted for face recognition in [18]. The algorithm works as described below. We first trained a generic face model using the (FRGC) version 1 database. In the recorded movies, we extract frames which contain the face of the individuals. Movies recorded in the first session resulted into 5994 images that were used during enrollment. Movies recorded in the second session resulted into 2959 images that were used during testing. Images from our mobile database are shown in *Figure* 5.9 where the images on the top were recorded in the second session and thus were used for testing and the bottom images were recorded in the first session and were used for testing. In each of these images, we automatically located the faces using the face detection method of Viola-Jones [86] which finds facial landmarks like eyes, nose and mouth. These landmarks are used to align the faces (see the bottom images of *Figure* 5.9) We only used the first hundred correctly found faces for the recognition in both sessions. For each image the region of interest is selected, the background is removed (see *Figure* 5.9 bottom left) and the region of interest is normalized to zero mean and unit variance. The difference between the face in the image and the generic face model generated from the FRGC database is computed. As a result each biometric sample can be represented as N (in our case equal to 30) independent feature vectors. On this database, the face recognition is more difficult due to larger deviations in the pose of individuals, illumination and the low quality of the movies. The EER , using the face recognition algorithm without correction, is 15.7%.

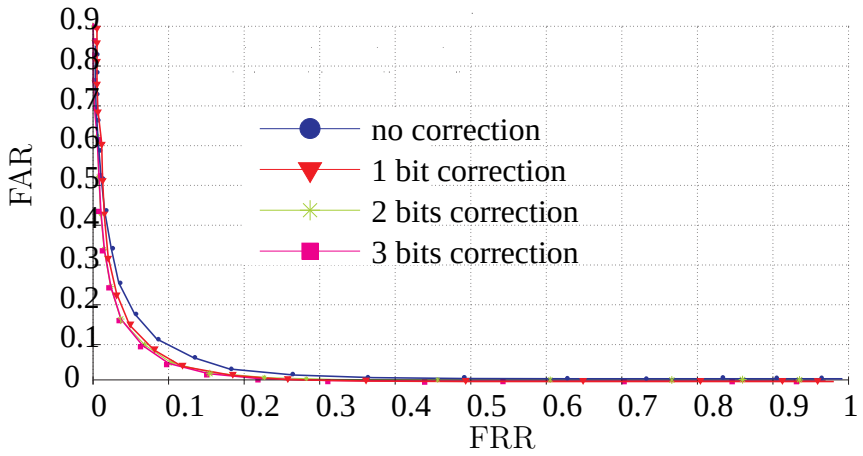


Figure 5.11: Experiment 2. ROC *curves on FRGC v1. database, uncontrolled set.*

At this stage we apply the shielding scheme fuzzy embedder proposed by [49] to extract cryptographic keys from face data. We use the data collected in the first session to estimate an average face template for each of the 31 users. We generate a random key of 30 bits length for each user. We use the embed procedure to generate the helper data and the error profile as described in section 5.4.3.

To estimate the FRR we do the following: for each user we use the biometric measurements from the second session and the helper data of each user as input to the reproduce procedure. The result of this operation is a binary key. We compare this result to the original key generated during enrollment. If they do not match exactly it means that we a have a false rejection. The FRR represents the percentage of the false rejections from the total number of trials.

To estimate the FAR we first choose a target of attack (one particular user). We apply the reproduce procedure to all the biometric measurements of the other users and the helper data of the target. The resulted key is compared with the target key. If they match we have a false acceptance. The FAR represents the percentage of false acceptance from the total number of trials where all users in the database were target.

By varying the quantization step q in the embed procedure we can tune the FAR and the FRR. *Figure* 5.10 shows the ROC curves obtained with and without correction. Of interest is the EER , which allows to evaluate the performance of the fuzzy embedder on the target data and the effect of the SmartFlip function. We notice that without any corrections the EER is around 29% with 1 bit correction the EER drops to approximately 19% and after further correcting 2 bits the EER is approximately equal to the one obtained by the biometric based classifier

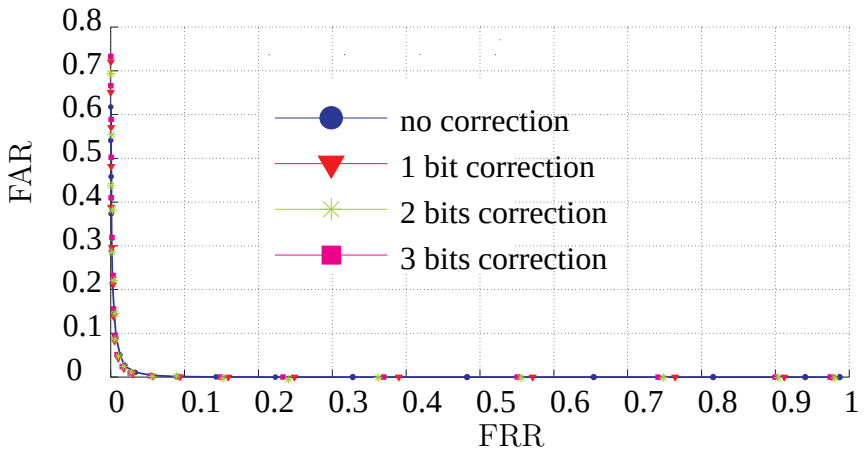


Figure 5.12: Experiment 3. ROC *curves on FRGC v1. database, controlled set.*

15%. By doing 3 bits correction we obtain an EER of approximately 12%.

FRGC DATABASE. In the FRGC database the position of eyes, nose and mouth are labelled, thus images can be easily aligned. For each image the region of interest which contains most of the face is selected and the background is removed.

A generic face model using all images in the FRGC database is trained. The difference between each face in the database and the generic face model is computed and stored as the feature vector. A combination of PCA and LDA algorithms is used on all feature vectors in the database. As a result each biometric sample can be represented as N (in our case equal to 50) independent feature vectors. In the FRGC database the data set obtained in controlled conditions contains 3772 images while the data set obtained in uncontrolled conditions contains 1886 images. In each experiment, the data set is randomly divided into two subsets, each consisting of approximately half of the images of each person. One subset is used for training and enrollment while the other subset is used for testing. The same algorithm for extracting cryptographic keys from face data and the same evaluation methodology is used as in the mobile database experiment.

The results of the experiment on the uncontrolled data set can be seen in *Figure* 5.11. Without any correction the EER is approximately equal to 9.2%. With 1 bit correction the EER is lowered to 8.7%. By doing 2 bit correction the EER can be lowered to 8.6%. Three bit correction, unfortunately cannot further improve the EER .

The results of the experiments on the controlled data set are shown in *Figure* 5.12. On this data set without any correction the EER is approximately 2.2%. With 1 bit correction the EER is lowered to approximately 1.8%. Also, in this case correcting more bits do significantly improve the EER .

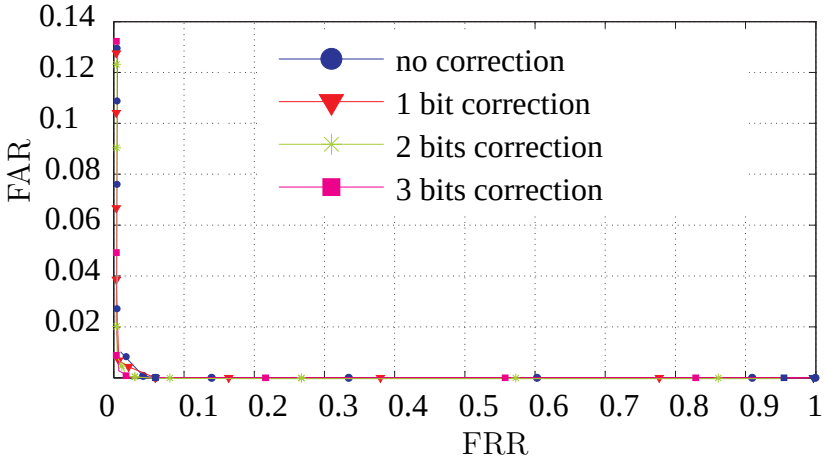


Figure 5.13: ROC *curves for on hand grip pressure data, controlled set.*

Summarizing, we did perform simulations for face recognition biometrics on three different data sets. The first is the mobile data set, which contains face data collected with a mobile device of 31 persons. The EER for this data set is around 12%. The second is the uncontrolled set of the FRGC v.1 data set which contains face data of 275 persons collected in uncontrolled situations. The best EER we obtained on this data set is 8.6 %. The third is the controlled set of the FRGC v.1 data set which contains face data of 275 persons collected in controlled situations. The lowest EER we obtained through simulations is 1.8 %. In the next section we look at a different biometric systems that uses hand grip pressure pattern to distinguish between individuals.

5.6.2 Hand grip pressure pattern biometric.

The evaluation is performed on real life grip pattern biometric data collected from 41 participants, in one session. A detailed description of this biometric can be found in Veldhuis *et al.* [85].

Each of the 41 participants contributed 25 different measurements. Approximately 75% of these samples(18), are used for training the algorithm and 25% (7) are used for testing. Firstly, we reduce the dimensionality of the data to maximum of 40 independent features. For training and testing we use the same data that is used for verification by the classifier based recognition algorithm. Secondly, we construct cryptographic keys using the fuzzy embedder as described above only this time the length of the key is 40 bits. *Figure* 5.13 presents the ROC obtained from the collected data. Without corrections the EER on the target data set is

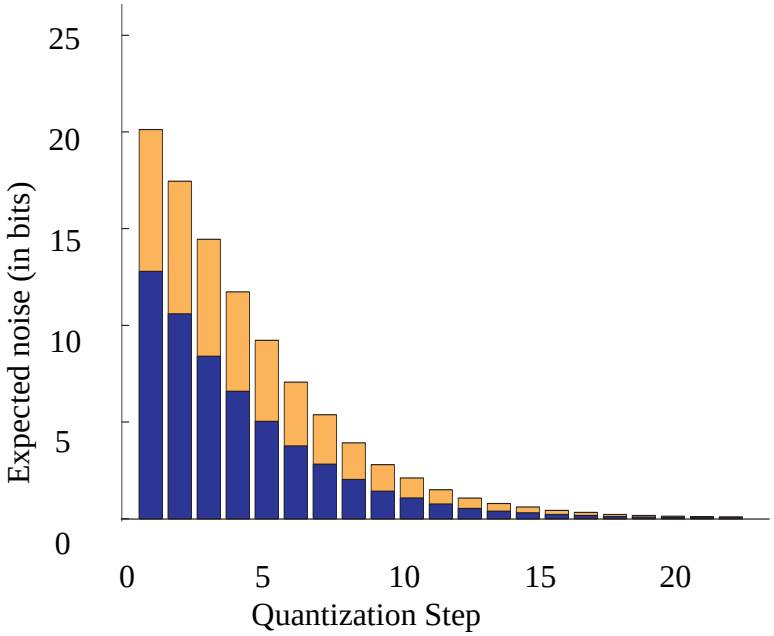


Figure 5.14: *Expected noise for Alice (dark-blue) and Eve (light-orange) for different quantization steps (different points on the* ROC *curve). Eve and Alice use the same type of camera.*

around 5%. After 1 bit correction, the EER drops significantly to 3.5% further after correcting 2 bits the EER goes down to 2.7% while correcting 3 bits further lowers the EER to approximately 2%. The EER values are better in the case of hand grip pressure biometric compared to the face data. One of the reasons is that hand data was collected in one session thus the variations between the training data used for enrollment and the testing data is not too large allowing for much better authentication performance.

Summarizing, after doing three bits of correction the EER we obtained through simulations is around 2%.

5.6.3 Practical Security Evaluation

We analyze in this paragraph how difficult it is for Eve to guess the communication key when the mobile data set is used to embed the communication key in the four scenarios described in section 5.5. We choose the mobile data set for the practical evaluation since this is the best case for Eve. The communication key has 60 bits when the mobile data set is used compared to 80 bits when the FRGC v.1 data set is used.

In this evaluation the most difficult problem is to give a realistic estimation of

the noise. By noise we understand a binary pattern which represents bits that are different between two binary strings or keys. By e we denote the noise expected for Alice and by e' we denote the noise expected for Eve when she takes a picture of Alice. However, when Eve is guessing the communication key the noise is $e - e'$, see section 5.5 for details. Our task is to evaluate from the experimental data the Hamming weights for e and $e - e'$. We make a few observations. As has been showed in section 5.5, Eve cannot lower her workload below that of Alice by using a good quality camera. Since Eve does not have the noise free key (m_A is never revealed during the protocol) her expected workload is larger then the workload of Alice. The noise between any two independent biometric measurements is also independent. The noise expected for Eve or Alice depends on the errors the biometric recognition algorithm can tolerate. Thus, for each point on the ROC curve in *Figure* 5.10 the amount of noise will vary.

For a realistic estimation of the noise we adopt the following solution. On the available data sets we compute the average number of bits that are different between the keys of all users for each point on the ROC curve. The average values are seen as the noise of the legitimate participants thus represent the Hamming weight of e.

The question now is: if we know e what is a realistic approximation for $e - e'$? We look at two cases: (a)-worse case scenario (for us) where Eve obtains exactly the same biometric measurements as the Alice and Bob, written formally as $e = e - e'$ and (b)-an average case scenario where the e and e' are not identical but they overlap. The overlap is estimated analytically as the percentage of the total length of the key that the Hamming weight of e represents. *Figure* 5.14 shows the Hamming weight of e versus the Hamming weight of $e - e'$ for different quantization steps. When the quantization step is relatively small (few errors are tolerated) the expected noise (the number of bits that are different) is relatively high for both Alice and Eve. The more the quantization step increases the more errors can be tolerated, the noise decreases and there is less work for Alice but also for Eve.

Figure 5.15 shows the number of trials that Eve has to perform versus the workload of Alice in the 4 scenarios described in section 5.5. When Eve has no information about Alice and Bob her workload is constant regardless the size of the quantization interval. In this scenario she will have to perform on average 10^{36} trials before she finds the correct key, *Figure* 5.15 (a).

We look at the quantization step where the EER is reported, in our case the EER is obtained when the quantization step is 10. At this point the workload of Eve in the scenario where she has no information about Alice but she has the picture of Bob is approximately 10^{18} trials in the worst case scenario and 10^{20} in the average case, see *Figure* 5.15 (b). When Eve has the picture of Alice but no

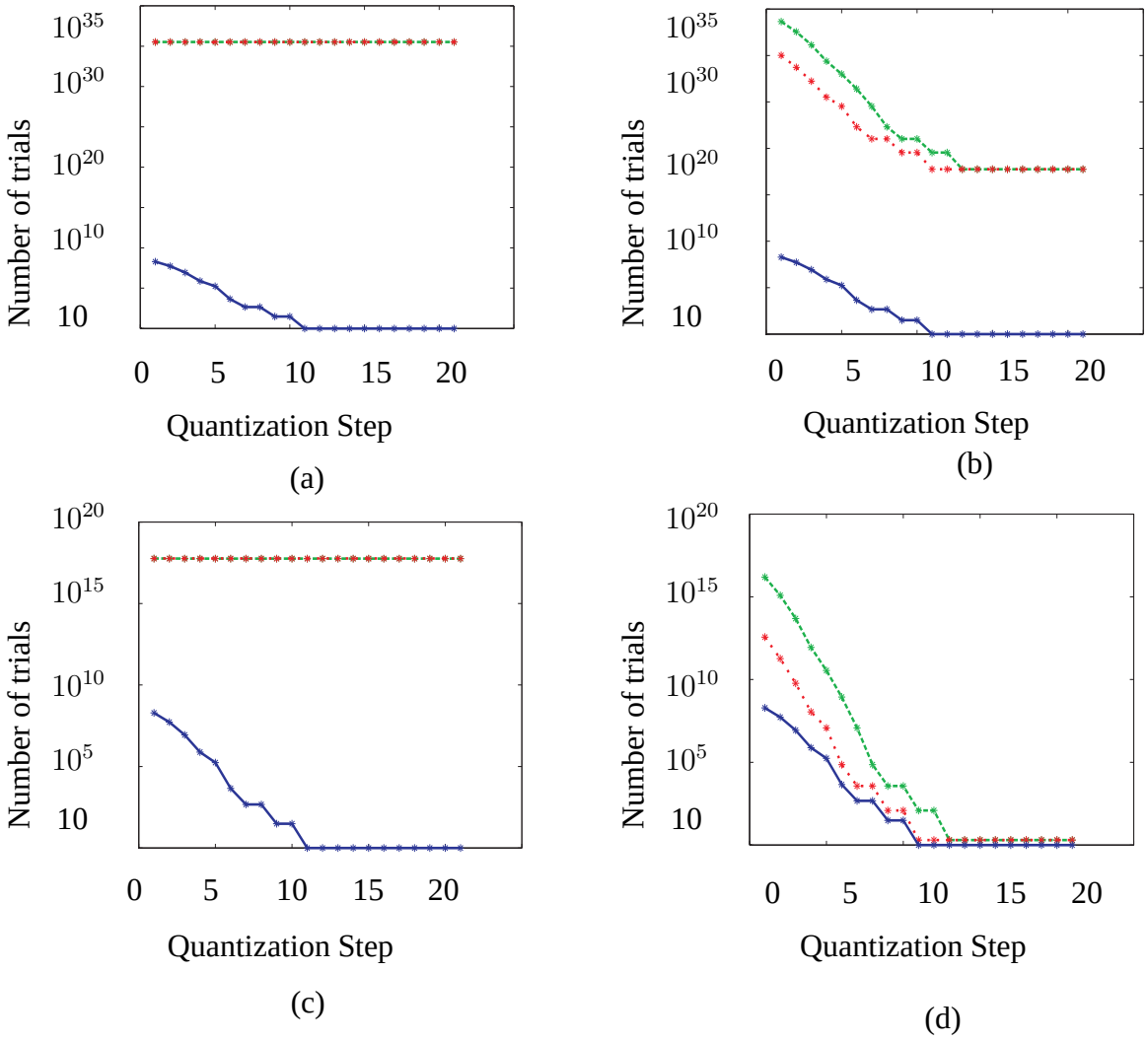


Figure 5.15: *Workload of Eve in worst case scenario (dotted) for Alice and Bob (and best case for her) and average case scenario (dashed) vs. the workload of Alice without using and error profile enhanced search (solid) when (a) Eve has no information about Alice or Bob, (b) Eve has no information over Alice and has the picture of Bob, (c) Eve has the picture of Alice and no information over Bob (d) Eve has the pictures of both Alice and Bob.*

Gender	Age	Education
Male: 60% Female: 40%	18−24: 10% 25−29: 56% 30−34: 20% 35−39: 7% 40+: 7%	High school: 7% Bachelor: 17% Masters: 46% Doctorate: 30%

Table 5.2: *Participant profile.*

information over Bob, due to the asymmetry of the protocol she has to perform approximately 10^{17} trials, *Figure* 5.15 (c). When Eve has both the picture of Alice and Bob she has to make in the worst case for us (and best case for her) the same number (in order of 10th) of trials as Alice and 10^4 in the average case, *Figure* 5.15 (d). In this case the workload of Eve is unacceptably low. A solution is to use another quantization step. For example when using quantization step number 3 Alice has to perform on average 10^7 trials while Eve has to make between 10^{10} (worst case) and 10^{14} (average case) trials.

Assume that Alice and Eve can perform one trial operation at the same speed. Assume further that it takes Alice 10 seconds to perform 10^7 trials (each trial implies setting a new key, a decryption operation and a comparison to decide whether the result is correct). In these settings it takes Eve in the worst case around 2.7 hours to find the communication key and 3 years in the average case.

VALIDATION EXPERIMENTS CONCLUSION. We offer four conclusions from the evaluation on the two sets of biometric data. The first conclusion is that error rates and thus performance of our protocol depends mostly on the quality of the collected biometric data, regardless of the biometric type of data. The second conclusion is that the influence of the correction algorithm is significant, however, the EER of the fuzzy embedder will be around the EER of the biometric based matcher. Increasing the number of bits that are corrected does not increase linearly the performance of the fuzzy embedder, the most significant improvement is obtained after the first bit of correction after which the improvement decreases. The third conclusion is that the correction mechanism is stable, meaning that the effect of correction is independent of the type of biometric The fourth conclusion is that it is possible to tune the workload of Eve compared to that of Alice such that security level is acceptable, even when Eve has the picture of both Alice and Bob.

5.7 Usability Analysis

Security only works if people use it therefore we conducted a comparative usability analysis between a PIN based pairing method and SAfE pairing. As a guideline we used the usability study by Uzun, *et al*. [82] for secure pairing methods. Our results are presented for a comparable target population.

TEST DESIGN AND PROCEDURE. Each subject was given a brief introduction to the secure device association scenario where people need to exchange sensitive information without having any prior security association. The researcher explained that the subject has to try two different pairing methods; one is the standard Bluetooth pin based pairing method and the other is our SAfE protocol. The subjects were asked to complete a background questionnaire first, so that we could learn about the subject demographics and mobile device usage history. Next, the subject was asked to try both pairing methods in a random order. For the SAfE protocol we wrote a program that implements only the user interaction part of the SAfE protocol. For the PIN based pairing we used the standard Bluetooth pairing method as provided in our device. Each subject was asked to choose a 4 digit PIN number and to enter it. For the SAfE protocol the subject was asked to take a picture of the researcher. All other actions with the PDAs were performed by the researcher. It was explained that only the steps required to perform the pairing are the subject of our experiment. After completing both pairing protocols subjects were asked to fill in the post-test questionnaire. The testing was done in a room with no disturbance and the testing time was around 20 minutes per subject with at least 15 minutes of free discussions. During both pairing protocols subjects were using the same ETEN M600+ PDA.

PARTICIPANT PROFILE. Our usability experiment had 30 participants from a university environment representing 13 different countries. The demographics such as gender, age and education for our subjects are presented in table 5.2. Most of our subjects have a computer science background.

The average computer usage history was around 15 years with an average of 9 computer hours per day. All participants have a mobile phone, a PDA or a laptop.

ANALYSIS AND DISCUSSIONS. The conclusions drawn from the experiment can be considered only as indicative due to the small number of participants and the (university) biased profile of our subjects.

The main purpose of our experiment was to discover whether users would find it easier to use SAfE protocol compared to a standard 4 digit PIN based pairing.

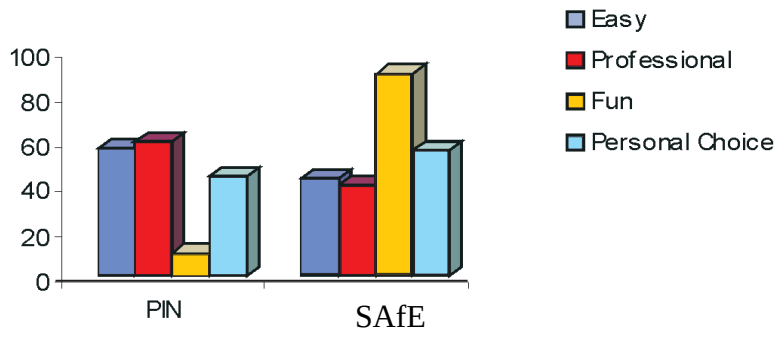


Figure 5.16: *Summary of participants opinion (in percent).*

As shown in figure 5.16 the score was tight with slightly more people preferring PIN pairing.

The explanation for the overall preference for the PIN based method is that subjects are familiar with PIN based security (ATMs, Bluetooth) and typing numbers is natural to subjects with a computing background. Some subjects used the adjective “easy” to describe the SAfE method. Others found it easy to understand how PIN based pairing method works but they used the word ’magic’ to describe the SAfE protocol. We did not try the experiment with a longer PIN and it is worth noting that approximately 80% of our participants choose the same PIN number(1234).

Most of our subjects, 90%, found it fun to perform the pairing using a camera and 73% would like to have both pairing methods on their mobile device (in figure 5.16 the percentage of only PIN or only SAfE choices are shown). Due to the “fun” effect of taking pictures the adjective “professional” was used more to describe PIN than SAfE.

A separate topic in the questionnaire concerned the privacy effect of giving away a photo to the researcher. To our surprise 56% of the subjects were not bothered to have their picture taken by a relative stranger. For those 44% who are bothered nothing changes if they have the photograph of the researcher. It was suggested that a privacy guarantee such as “picture deleted after pairing complete” would improve things significantly. To our satisfaction 87% of the users want to have security while communicating wirelessly. Summarizing, the usability experiment provides an indication that taking pictures provides a possible route towards creating security associations because it is fun. Whether people believe that taking pictures is professional enough to provide good security is an open question.

5.8 Conclusion

Secure device association is a challenging problem from both the technical and the user interface point of view. Firstly, users need to exploit a common secret source of randomness from which to extract a shared secret key. Secondly, it should be possible to link the device we connect to with the person who owns it. Thirdly, the process should be simple such that for any person with non technical background the protocol is easy to use.

In this chapter we propose the SAfE protocol which uses biometrics as the out-of-band channel. We analyze our protocol from three different perspectives. Firstly, we analyze the security of the protocol against two types of adversaries Eve which has computational capabilities and Charlie a Dolev-Yao attacker. We show that our protocol is not vulnerable to a man-in-the-middle attack and we analyze eavesdropping in four different scenarios both from theoretical and practical point of view. We show that in the average case when Eve has the biometric measurements of both Alice and Bob her workload is significant. Assume both Alice and Eve execute at the same speed 1 trial operation. Assume further it takes Alice 10 second to perform 10^7 trials. In these settings it would take Eve 3 years to perform 10^{14} trials, expected in an average case scenario. Of course Eve can use more powerful computers or execute operations in parallel. Since our protocol is intended for ad-hoc situations were confidential but not critical information is exchanged, as long as it would take Eve more than 7 days to find the communication key we consider our protocol secure. The workload of Eve, thus the security of the protocol can be increased but it would also increase the error rates. A convenient balance can be found on a case by case basis. It would have been extremely interesting to test the resilience of the protocol against an attacker who has both the abilities of both Eve and Charlie. Unfortunately we are not aware of any formal approach that can handle such an attacker.

Secondly, we evaluate the performance of the protocol with two types of real life biometric data: face recognition and hand grip pressure pattern. Binary keys are generated independently of the biometric data for each protocol round and combined with biometric information. This is a necessary approach since one has only one face, 10 fingerprints, etc. For face recognition we collected face data with a camera of a mobile device, in two different days in uncontrolled environment (light, face expression) as it would be the case in the real world. We obtain on this data set an EER of approximately 12% after applying a correction function that we designed. To consolidate our experiments we repeat the experiments on the FRGC v.1 database, which contains 275 individuals. Images in this database can be divided into two data sets: images obtained in uncontrolled conditions and images obtained in controlled conditions. These experiments are interesting

as they show the perspective of the pairing algorithm in the future. Results on the uncontrolled data set are relevant in the near future when mobile devices may capture and process high quality images. Simulations on this data set show that the EER without correction is approximately 9.2% and can be lowered by correction to 8.6%. Results on the controlled data set are relevant when changes in pose and lighting are no longer a problem for face recognition. Simulations on this data set show that the EER is 2.2 % without corrections and one can lower this value to 1.8 % by doing corrections.

On the hand grip pressure pattern biometric we obtained an EER that is approximately 2%. The main reason is the high quality of the data, all hand grip data were recorded in one session from trained individuals. As we noted before the quality of biometric data is the main factor that can lower the error rates. A carefully designed data acquiring interface is needed for good performance.

Thirdly, we look at our protocol from the perspective of the user. Our usability analysis shows that our subjects find the SAfE protocol fun to use, and that they would like to have the SAfE pairing available on their mobile devices. However, there are some situations where SAfE is not appropriate: (a) when the participants wish to communicate without drawing attention (such as in a restaurant or at a business meeting) (b) when the protocol fails (for example under bad lighting conditions). Therefore a back-up solution for SAfE is needed that is smoothly integrated with the system. The user would then have the choice of a more user friendly biometric based pairing method and a more robust alternative method.

The contribution of this chapter is related to the SECURE TEMPLATE TRANSFER recommendation, which is the result of the 3W-tree analysis in *Chapter* 2. Our solution to this problem is new, in the sense that biometrics is used for the first time as an out-of-band channel. Using biometrics makes the SAfE protocol, user-friendly and fun as pointed out by our usability analysis in *Section* 5.7. More importantly, it offers strong security guarantees, compared to other solutions in the literature. Since false acceptance and false rejection threat cannot be eliminated completely from any system that uses biometrics we hope that advances in the technology (more accurate cameras) and in the field of biometrics (algorithms resilient to environmental variations) can be reduced to an acceptable level.

Chapter 6

Conclusions

We now summarize the contributions of this thesis, in relation to the main Research Question described in *Chapter* 1. We also highlight future research directions in the area of cryptography with noisy data.

In the introductory chapter we formulate the following research question:

How can we mitigate *the risk of malicious errors in a biometric authentication system?*

Defense methods to mitigate risk are designed with a particular application in mind. Thus, we answer the above question in the context of designing the SmartGun architecture, a biometrical enabled weapon which can be fired only by an authorized user. In *Chapter* 2 using the 3W-tree, a novel threat analysis method specifically designed for the biometric system architecture, we propose a three step procedure, which consists of (1) identification, (2) classification and (3) analysis of biometric faults.

The result of the initial 3W-tree analysis for the biometric SmartGun gives six research directions. In this thesis we focus on the two security related recommendations. The first security direction is SECURE TEMPLATE STORAGE, which states that it should not be possible to reconstruct the biometric identifier from the data stored in the gun. We explore the challenges related to this topic, which are both theoretical and practical and we put forward solutions to many of the issues in *Chapters* 3 and 4. The second security direction is SECURE TEMPLATE TRANSFER, which states that it should be possible to transfer the biometric identifiers between two guns when no security infrastructure is available and when the users are no security experts. The solution we propose to this problem is explored in *Chapter* 5.

Contribution	Theoretical	Practical
Chapter 2 (Threat Analysis)		
Section 2.4	3W-tree	
Section 2.5		SmartGun Analysis
Chapter 3(*cs*-Fuzzy Extractors)		
Section 3.4	Link FAR- min-entropy *cs*-Fuzzy Extractors	
Section 3.5		Reliable Component Shielding Function Chang multi-bit scheme
Chapter 4 (Fuzzy Embedders)		
Section 4.4	Fuzzy Embedders	
Section 4.5	`QIM`-Fuzzy Embedder	
Section 4.6		6-Hexagonal Tiling 7-Hexagonal Tiling
Chapter 5 (SAfE Protocol)		
Section 5.5	SAfE Protocol Smart Key Search	
Section 5.6	Formal and Computational Analysis	
Section 5.7		Face Recognition and Hand Grip Experiments
Section 5.8		Usability Analysis

Table 6.1: *Theoretical and Practical contributions of this thesis.*

In this thesis we make progress in both research directions. We support this conclusion by describing our main contributions, which span a wide range of activities:

- We develop two novel template protection schemes, the 6 hexagonal tiling, which is optimal from security point of view and the 7 hexagonal tiling which is optimal from reliability point of view compared to their counterparts in the literature.

- We create novel definitions (of *cs*-fuzzy extractor, fuzzy embedder), which expand current models in the literature (fuzzy extractor).

- We show that the number of uniformly random bits that can be extracted

from a noisy source depends on the quality of the noisy data (i.e. biometric data) expressed in terms of FAR and FRR.

- We model mathematically the relationship between the security and reliability of template protection schemes as a dual sphere covering vs. sphere packing problem.

- We develop a new protocol for spontaneous device interaction using biometrics when no security infrastructure is available, which we demonstrate to be fast, user friendly and reliable.

An overview of the theoretical and practical results for each chapter is given in *Table* 6.1.

FUTURE WORK. The results in the thesis open several possible future research directions, both theoretical and practical.

- The theoretical results in *Chapter* 4 show that our new template protection schemes, the 6 hexagonal tiling and the 7 hexagonal tiling, are superior compared to other theoretical constructions in the literature. We would like to have these results confirmed in practice by results on real life data.

- In *Chapter* 4, we identify a few basic building blocks that can be used to construct a practical system, which extracts cryptographic keys from noisy data. Which blocks to use and in which order, is mostly determined by the "know-how" of the system engineer and the application context. So far no theoretical study was performed to determine any optimality criteria.

- We are working on a complete prototype that runs the SAfE protocol on two mobile devices. We are particularly interested in testing the influence of different environments on the key search failure since environmental effects such as changing the light conditions can seriously affect the face recognition performance.

- We verify the security of the SAfE protocol both formally, to prove that a man-in-the-middle attack is not possible and computationally, to estimate how much effort is required from an attacker who is actively involved in guessing the communication key.

  We leave the security verification against an adversary who can play the man-in-the-middle and who tries to guess the communication key at the same time as future work.

- False acceptance and false rejection threats are inherent to biometrics. Experiments show that the strength of a cryptographic key extracted from biometric data and the reliability with which the two legitimate users compute the key depends on the error rates of the biometric classification algorithm. At this point in time we find that face recognition algorithms for mobile device are not mature enough. We leave the exploration of other biometric modalities, such as fingerprints, iris or fusion of different biometric modalities, which might have superior performance as future work.

- Although biometric authentication is used to enhance security, storing biometric data, in a database introduces new security and privacy risks. In the literature there are several measures such as min-entropy, entropy-loss, relative entropy-loss, etc which are used to determine the security or privacy offered by a template protection scheme. In the future the right measures for evaluating both security and privacy scheme have to be agreed upon.

SMARTGUN. Finally, we present a perspective on the development of a SmartGun for the Dutch police. In *Chapter* 2 as a result of the 3W-tree analysis for the biometric SmartGun, we identify 6 general recommendations for the architecture of the biometric SmartGun.

The first two are LOW FALSE REJECTION RATE and LOW FALSE ACCEPTANCE RATE. These recommendations are the subject of intense research effort in *hand grip pressure pattern* biometrics, a new type of biometrics with interesting applications. Results are encouraging and more information can be found in the PhD thesis of Xiaoxing Shang. Currently for a FRR of 10^{-4}, which is the officially accepted failure rate in the Netherlands for a police weapon the FAR is approximately 30%, for public acceptance, however, the target is to have a FAR that is at the most 5%.

Development of a ROBUST SENSOR which is resilient to wear and tear is the research area of TSST (Twente Solid State Technology). From a technological point of view it is feasible to build a robust sensor, but tests in practical environment have not taken place. Solutions for the SECURE SEAL recommendation are considered as engineering challenges that will have to be dealt with by the gun manufacturers.

The security related recommendation, the SECURE TEMPLATE STORAGE and the SECURE TEMPLATE TRANSFER can be met by solutions put forward by this thesis. Finally, there are many, practical and still open problems to be solved such as interface the electronics to mechanical parts, making the battery last long enough, find space for the electronics in the gun butt and evaluate the reliability of the system.

When progress is made regarding the open problems, we recommend a new

3W-tree analysis of the SmartGun. The 3W-tree analysis should be an iterative process since new solutions may introduce new vulnerabilities that were not foreseen in the earlier 3W-tree analysis.

Author References

Journal Publication

[1] I.R Buhan, B. Boom, J.M. Doumen, P.H. Hartel, and R.N.J. Veldhuis. Secure ad-hoc pairing with biometrics: Safe. *International Journal of Security and Networks Special (IJSN), Special Issue on Secure Spontaneous Interaction*, 4(1):to appear, 2009. (Subsumed by Chapter 4 of this thesis.).

Refereed Conferences

[2] I.R. Buhan, A.M. Bazen, P.H. Hartel, and R.N.J. Veldhuis. A false rejection oriented threat model for the design of biometric authentication systems. In D. Zhang and A. K. Jain, editors, *2nd International Conference on Biometrics (ICB), Hong Kong, China*, volume 3832 of *Lecture Notes in Computer Science*, pages 728–736, Berlin, January 2006. Springer-Verlag. (Subsumed by Chapter 2 of this thesis.).

[3] I.R. Buhan, J.M. Doumen, and P.H. Hartel. Controling leakage of biometric information using dithering. In *Proceedings of the 16th European Signal Processing Conference (EUSIPCO), Lausanne, Switzerland*, EUSIPCO. European Association for Signal, Speech and Image Processing, EURASIP, August 2008.

[4] I.R. Buhan, J.M. Doumen, P.H. Hartel, and R.N.J. Veldhuis. Feeling is believing: A secure template exchange protocol. In Seong-Whan Lee and Stan Z. Li, editors, *Advances in Biometrics, International Conference (ICB'07), Seoul, Korea*, volume 4642 of *Lecture Notes in Computer Science*, pages 897–906. Springer, August 2007. (Subsumed in Chapter 5 of this thesis.).

[5] I.R. Buhan, J.M. Doumen, P.H. Hartel, and R.N.J Veldhuis. Fuzzy extractors for continuous distributions. In R. Deng and P. Samarati, editors, *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security (ASIACCS), Singapore*, pages 353–355, New York, March 2007. ACM. (Subsumed by Chapter 3 of this thesis, except examples.).

[6] I.R. Buhan, J.M. Doumen, P.H. Hartel, and R.N.J. Veldhuis. Embedding renewable cryptographic keys into continuous noisy data. In *(to appear) 10th International Conference on Information and Communications Security*

(ICICS), Lecture Notes in Computer Science, Birmingham, UK, October 2008. Springer-Verlag. (Subsummed in Chapter 4 of this thesis).

International Workshops

[7] I.R. Buhan, J.M. Doumen, P.H. Hartel, and R.N.J. Veldhuis. Secure ad-hoc pairing with biometrics: Safe. In *First International Workshop on Security for Spontaneous Interaction, Innsbruck, Austria*, pages 450–456, Innsbruck, Austria, September 2007. Ubicomp 2007 Workshop Proceedings.

[8] I.R. Buhan, J.M. Doumen, P.H. Hartel, and R.N.J. Veldhuis. A unifying view on template protection schemes. In R. N. J. Veldhuis and H. S. Cronie, editors, *Proceedings of the 28th Symposium on Information Theory in the Benelux, Enschede, The Netherlands*, pages 35–42, Eindhoven, May 2007. Werkgemeenschap voor Informatie- en Communicatietechniek. (Subsumed by Chapter 3 of this thesis.).

CTIT Technical Report

[9] I.R. Buhan and P.H. Hartel. The state of the art in abuse of biometrics. Technical Report TR-CTIT-05-41, University of Twente, Enschede, September 2005.

General References

Others

[10] A. Adler. Vulnerabilities in biometric encryption systems. In Takeo Kanade, Anil K. Jain, and Nalini K. Ratha, editors, *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication, (AVBPA 2005), Hilton Rye Town, NY, USA,*, volume 3546 of *Lecture Notes in Computer Science*, pages 1100–1109. Springer-Verlag, 2005.

[11] R.J. Anderson. *Security Engineering: A guide to building dependable distributed systems*. John Wiley & Sons Inc, New York, 2001.

[12] A. Avizienis, J.C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, 1(1):11–33, January-March 2004.

[13] D. Balfanz, D. K. Smetters, P. Stewart, and H. C. Wong. Talking to strangers: Authentication in Ad-Hoc wireless networks. In *Proceedings of the Network and Distributed Systems Security Symposium (NDSS), San Diego, California, USA*, San Diego, California, February 2002. The Internet Society, Reston, Virginia.

[14] B. Barak, R. Impagliazzo, and A. Wigderson. Extracting randomness using few independent sources. *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS'04), Roma, Italy*, 45:384–393, October 2004.

[15] R.J. Barron, B. Chen, and G.W. Wornell. The duality between information embedding and source coding with side information and some applications. *IEEE Transactions on Information Theory*, 49(5):1159–1180, May 2003.

[16] R. Bolle, J. Connell, S. Pankanti, N. Ratha, and A. Senior. *Guide to Biometrics*. Springer-Verlag, 2003.

[17] J. M. Bone and D. M. Blackburn. Biometrics for narcoterrorist watch list applications. Technical report, Crane Division, Naval Surface Warfare Center and DoD Counterdrug Technology Development Program Office, July 2003.

[18] B.J. Boom, G.M. Beumer, L.J. Spreeuwers, and R.N.J. Veldhuis. The effect of image resolution on the performance of a face recognition system. In *9th International Conference on Control, Automation, Robotics and Vision (ICARCV '06), Sinagpore*, pages 1–6, December 2006.

[19] X. Boyen. Reusable cryptographic fuzzy extractors. In V. Atluri, B.Pfitzmann, and P. D. McDaniel, editors, *Proceedings of the 11th ACM Conference on Computer and Communications Security (CCS 2004), Washinton DC, USA*, pages 82–91. ACM, October 2004.

[20] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith. Secure remote authentication using biometric data. In Ronald Cramer, editor, *Advances in Cryptology, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT 2005), Aarhus, Denmark*, volume 3494 of *Lecture Notes in Computer Science*, pages 147–163. Springer, May 2005.

[21] A.M. Bronstein, M.M. Bronstein, and R. Kimmel. Three-dimensional face recognition. *International Journal of Computer Vision*, 64(1):5–30, August 2005.

[22] UK Government Biometrics Working Group (BWG). Biometric device protection profile (bdpp). *2001 Crown Copyright*, Draft Issue 0.82(5):428–441, Sep 2001.

[23] UK Government Biometrics Working Group (BWG). Biometric security concerns. *v1.0*, v.0.1(5):428–441, Sep 2003.

[24] E.C. Chang and Q. Li. Hiding secret points amidst chaff. In Serge Vaudenay, editor, *25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Saint Petersburg, Russia*, volume 4004 of *Lecture Notes on Computer Science*, pages 59–72. Springer, May 2006.

[25] Y.J. Chang, W. Zhang, and T. Chen. Biometrics-based cryptographic key generation. In *IEEE International Conference on Multimedia and Expo (ICME'04), Taipei, Taiwan*, pages 2203–2206. IEEE Computer Society, June 2004.

[26] B. Chen and G.W. Wornell. Dither modulation: a new approach to digital watermarking and information embedding. *Proceedings of SPIE Security and Watermarking of Multimedia Contents*, 3657:342–353, April 1999.

[27] B. Chen and G.W. Wornell. Quantization index modulation methods for digital watermarking and information embedding of multimedia. *The Journal of VLSI Signal Processing, Springer Netherlands*, 27(1-2):7–33, February 2001.

[28] C. Chen, R.N.J. Veldhuis, T.A.M. Kevenaar, and A.H.M. Akkermans. Multi-bits biometric string generation based on the likelyhood ratio. In *IEEE Conference on Biometrics: Theory, Applications and Systems (BTAS'07), Washinton, DC*, pages 1–6. IEEE Computer Society, September 2007.

[29] R. Corin and S. Etalle. An improved constraint-based system for the verification of security protocols. In Manuel V. Hermenegildo and Germán Puebla, editors, *Proceedings of Static Analysis, 9th International Symposium (SAS 2002), Madrid, Spain*, volume 2477 of *Lecture Notes on Computer Science*, pages 326–341. Springer, September 2002.

[30] J. Daugman. How iris recognition works. In *Proceedings International Conference on Image Processing (ICIP 2002), Rochester, New York, USA*, volume 1, pages 1–31. IEEE Computer Society, September 2002.

[31] Germany DIN-Deutsches Institut Fur Normung E.V., Berlin. Information technology - security techniques - a framework for security evaluation and testing of biometric technology. Technical Report ISO/IEC JTC 1/SC 27 N 3806, DIN - Deutsches Institut fur Normung e.V. Berlin, Germany, 2003.

[32] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology ,Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT 2004), Interlaken, Switzerland*, volume 3027 of *Lecture Notes in Computer Science*, pages 523–540. Springer, May 2004.

[33] Y. Dodis and A. Smith. Correcting errors without leaking partial information. In Harold N. Gabow and Ronald Fagin, editors, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing(STOC), Baltimore, MD, USA*, pages 654–663. ACM, May 2005.

[34] D. Dolev and A.Yao. On the security of public key protocols. *IEEE Transactions on Information Theory*, 29(2):198–208, March 1983.

[35] S. Furui. An overview of speaker recognition technology. In *IEEE International Conference on Automatic Speech and Speaker Recognition: Advanced Topics (ICASSP'02), Orlando, Florida*, volume 4, pages 31–56. IEEE Computer Society, May 1996.

[36] A. Gersho. Principles of quantization. *IEEE Transactions on Circuits and Systems*, 25(7):427–436, July 1978.

[37] A. Gersho. Asymptotically optimal block quantization. *IEEE Transactions on Information Theory*, 25(4):373–380, July 1979.

[38] A.M. Bazen G.M. Beumer, Q. Tao and R.N.J. Veldhuis. Comparing landmarking methods for face recognition. In *Proceedings of the 16th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC 2005), Veldhoven, The Netherlands*, pages 594–597, November 2005.

[39] M. Goodrich, M. Sirivianos, J. Solis, G. Tsudik, and E. Uzun. Loud and clear: Human-verifiable authentication based on audio. In *26th IEEE International Conference on Distributed Computing Systems (ICDCS 2006), Lisboa, Portugal*, page 10. IEEE Computer Society, July 2006.

[40] A. Jain, L. Hong, and S. Pankanti. Biometric identification. *Communications of the ACM (CACM)*, 43(2):90–98, February 2000.

[41] A.K. Jain, K. Nandakumar, and A. Nagar. Biometric template security. *Journal on Advances in Signal Processing (EURASIP)*, 2008:17, 2008.

[42] A.K. Jain, S. Pankanti, S. Prabhakar, L. Hong, and A. Ross. Biometrics: A grand challenge. In *Proceedings of the 17th International Conference oN Pattern Recognition (ICPR' 04), Cambridge, UK*, volume 3, pages 935–942. IEEE Computer Society, August 2004.

[43] A.K Jain, A. Ross, and S. Prabhakar. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20, 2004.

[44] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *Proceedings of the 6th ACM Conference on Computer and Communications Security (CCS), Singapore*, pages 28–36. ACM SIGSAC, November 1999.

[45] G.A. Kabatiansky and V.I Levenshtein. Bounds for packings on a sphere and in space. *Problemy Peredachi Informatsii*, 1:3–25, 1978.

[46] T. Kindberg and K. Zhang. Secure spontaneous device association. In Anind K. Dey, Albrecht Schmidt, and Joseph F. McCarthy, editors, *In Proceedings of UbiComp 2003: Fifth International Conference on Ubiquitous Computing,(UBICOMP 03), Seattle, Washington*, volume 2864 of *Lecture Notes in Computer Science*, pages 124–131. Springer, October 2003.

[47] A. Kong, A. Griffith, D. Rhude, G. Bacon, and G. Shahs. Department of defense federal biometric system protection profile for medium robustness environments. Technical Report Technical Report Draft Version 0.02, U.S Department of Defense, 2002.

[48] Q. Li, Y. Sutcu, and N. Memon. Secure sketch for biometric templates. In Xuejia Lai and Kefei Chen, editors, *Advances in Cryptology 12th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2006), Shanghai, China*, volume 4284 of *Lecture Notes in Computer Science*, pages 99–113. Springer, December 2006.

[49] J.P. Linnartz and P. Tuyls. New shielding functions to enhance privacy and prevent misuse of biometric templates. In Josef Kittler and Mark S. Nixon, editors, *4th International Conference on Audio-and Video-Based Biometrie Person Authentication (AVBPA 2003), Guildford, UK*, volume 2688 of *Lecture Notes in Computer Science*, pages 393–402. Springer, June 2003.

[50] D.L. Lough. *A taxonomy of computer attacks with applications to wireless networks*. John Wiley & Sons Inc, New York, 2001.

[51] T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino. Impact of artificial "gummy" fingers on fingerprint systems. *Datenschutz und Datensicherheit*, 26(8):275289, 2002.

[52] U. Maurer. Perfect cryptographic security from partially independent channels. In *Proceedings of the 23rd ACM Symposium on Theory of Computing (STOC),New Orleans, Louisiana, USA*, pages 561–572. ACM Press, August 1991.

[53] U. Maurer. Secret key agreement by public discussion. *IEEE Transaction on Information Theory*, 39(3):733–742, May 1993.

[54] R. Mayrhofer and H. Gellersen. Shake well before use: Authentication based on accelerometer data. In Anthony LaMarca, Marc Langheinrich, and Khai N. Truong, editors, *Proceedings of the Fifth International Conference on Pervasive Computing (Pervasive'07),Toronto, Ontario, Canada*, volume 4480 of *Lecture Notes in Computer Science*, pages 144–161. Springer, May 2007.

[55] J.M. McCune, A. Perrig, and M.K. Reiter. Seeing-is-believing: Using camera phones for human-verifiable authentication. In *IEEE Symposium on Security and Privacy*, pages 110–124. IEEE Computer Society, May 2005.

[56] A.J. Menezes. *Handbook of Applied Cryptography*. CRC Press, 1997.

[57] F. Monrose, M.K. Reiter, Q.Li, and S. Wetzel. Cryptographic key generation from voice. In M.K. Reiter, editor, *Proceedings of the IEEE Symposium*

on Security and Privacy S&P, Oakland, CA, USA, pages 202–213. IEEE Computer Society, May 2001.

[58] A. P. Moore, R.J. Ellison, and R.C. Linger. Attack modeling for information security and survivability. Technical report, CMU/SEI-2001-TN-001, 2001.

[59] P. Moulin and R. Koetter. Data-hiding codes. *Proceedings of the IEEE*, 93(12):2083–2126, December 2005.

[60] P.G. Neuman and D.B. Parker. A summary of computer misuse techniques. In *12th National Computer Security Conference, Baltimore, MaryLand*, pages 396–407, 1989.

[61] L. O'Gorman. Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*, 91(12):2021–2040, December 2003.

[62] A.J. Rae and L.P. Wildman. A taxonomy of attacks on secure devices. *Australian Information Warfare and IT Security, Australia*, 0.1(5):251–264, November 2003.

[63] T.S. Rappaport. *Wireless Communications: Principles and Practice*. IEEE Press Piscataway, NJ, USA, 1996.

[64] N.K. Ratha, J.H. Connell, and R. M. Bolle. Enhancing security and privacy in biometric-based authentication systems. *IBM System Journal*, 40(3):614–634, September 2001.

[65] N.K. Ratha, J.H. Connell, and R.M. Bolle. Biometrics break-ins and band-aids. *Pattern Recognition Letters*, 24(13):2105–2113, September 2003.

[66] I. Ray and N. Poolsapassit. Using attack trees to identify malicious attacks from authorized insiders. In Sabrina De Capitani di Vimercati, Paul F. Syverson, and Dieter Gollmann, editors, *Proceedings of the 10th European Symposium on Research in Computer Security (ESORICS 2005), Milan, Italy*, volume 3679 of *Lecture Notes in Computer Science*, pages 231–246. Springer, September 2005.

[67] J.E.Y. Rossebø, S.C., and P. Sijben. tvra, a threat, vulnerability and risk assessment tool for europe. In Ketil Stølen, William H. Winsborough, Fabio Martinelli, and Fabio Massacci, editors, *Proceedings of the 4th International Conference on Trust Management (iTrust 2006), Pisa, Italy*, volume 3986 of *Lecture Notes in Computer Science*, pages 467–471. Springer, May 2006.

[68] N. Saxena, J. Ekberg, K. Kostiainen, and N. Asokan. Secure device pairing based on a visual channel. In *IEEE Symposium on Security and Privacy (S&P 2006), Berkeley, California, USA*, pages 306–313. IEEE Computer Society, May 2006.

[69] B. Schneier. Attack trees: Modeling security threats. *Dr. Dobb's Journal*, 24(12):21–29, 1999.

[70] Y. Shaked and A. Wool. Cracking the bluetooth pin. In Kang G. Shin, David Kotz, and Brian D. Noble, editors, *The Third International Conference on Mobile Systems, Applications, and Services (MobiSys 2005), Seatle, USA*, pages 39–50. ACM, June 2005.

[71] Xiaoxin Shang. *Grip-Pattern Recognition: Applied to a Smart Gun*. PhD thesis, University of Twente, 2008.

[72] B. Skoric, P. Tuyls, and W. Ophey. Robust key extraction from physical uncloneable functions. In John Ioannidis, Angelos D. Keromytis, and Moti Yung, editors, *Applied Cryptography and Network Security (ACNS 2005), New York, NY, USA*, volume 3531 of *Lecture Notes in Computer Science*, pages 407–422. Springer, June 2005.

[73] F. Stajano and R.J. Anderson. The resurrecting duckling: Security issues for ad-hoc wireless networks. In Bruce Christianson, Bruno Crispo, James A. Malcolm, and Michael Roe, editors, *Proceedings of 7th International Workshop on Security Protocols Workshop, Cambridge, UK*, volume 1796 of *Lecture Notes in Computer Science*, pages 172–194. Springer, April 1999.

[74] A. Ta-Shma. On extracting randomness from weak random sources (extended abstract). *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing (STOC 1996), Philadephia, Pennsylvania, USA*, 28:276–285, May 1996.

[75] L. Trevisan and S. Vadhan. Extracting randomness from samplable distributions. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science,Redondo Beach, CA, USA*, volume 41, pages 32–42. IEEE Computer Society, 2000.

[76] P. Tuyls, A. Akkermans, T. Kevenaar, G. Schrijen, A. Bazen, and R. Veldhuis. Practical biometric authentication with template protection. In Takeo Kanade, Anil K. Jain, and Nalini K. Ratha, editors, *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA 2005), Hilton Rye Town, NY, USA*, volume 3546 of *Lecture Notes in Computer Science*, pages 436–446. Springer, July 2005.

[77] P. Tuyls and J. Goseling. Capacity and examples of template-protecting biometric authentication systems. In Davide Maltoni and Anil K. Jain, editors, *Proceedings of International Workshop on Biometric Authentication (ECCV 2004), Prague, Czech Republic*, volume 3087 of *Lecture Notes in Computer Science*, pages 158–170. Springer, May 2004.

[78] U. Uludag and A.K. Jain. Attacks on biometric systems: a case study in fingerprints. In Edward J. Delp and Ping Wah Wong, editors, *Proceedings of SPIE on Security, Steganography, and Watermarking of Multimedia Contents (SSWMC 2004), San Jose, California, USA*, volume 5306 of *Proceedings of SPIE*, pages 622–633. SPIE, January 2004.

[79] U. Uludag, S. Pankanti, and A.K. Jain. Fuzzy vault for fingerprints. In Takeo Kanade, Anil K. Jain, and Nalini K. Ratha, editors, *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication, (AVBPA 2005) Hilton Rye Town, NY, USA*, volume 3546 of *Lecture Notes in Computer Science*, pages 310–319. Springer, July 2005.

[80] U. Uludag, S. Pankanti, S. Prabhakar, and A. K. Jain. Biometric cryptosystems: Issues and challenges. *Proceedings of the IEEE*, 92(6):948–960, 2004.

[81] U.Uludag, S.Pankanti, and A.K. Jain. Fuzzy vault for fingerprints. In Takeo Kanade, Anil K. Jain, and Nalini K. Ratha, editors, *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication, (AVBPA 2005) Hilton Rye Town, NY, USA*, volume 3546 of *Lecture Notes in Computer Science*, pages 310–319. Springer, July 2005.

[82] E. Uzun, K. Karvonen, and N. Asokan. Usability analysis of secure pairing methods. Technical report, NRC-TR-2007-002, Nokia Research Center, 2007.

[83] T. van der Putte and J. Keuning. Biometrical fingerprint recognition: Don't get your fingers burned. In Josep Domingo-Ferrer, David Chan, and Anthony Watson, editors, *Proceedings of the Fourth Working Conference on Smart Card Research and Advanced Applications (CARDIS 2000), Bristol, UK*, volume 180 of *IFIP Conference Proceedings*, pages 289–306. Kluwer, September 2000.

[84] R.N.J. Veldhuis, A.M. Bazen, W. Booi, and A. Hendrikse. A comparison of hand-geometry recognition methods based on low- and high-level features. In *Proceedings of the 15th Annual Workshop on Circuits Systems and Signal Processing (ProRISC 2004), Veldhoven, The Netherlands*. STW, 2004.

[85] R.N.J. Veldhuis, A.M. Bazen, J.A. Kauffman, and P.H. Hartel. Biometric verification based on grip-pattern recognition. In *Proceedings of SPIE the Security, Steganography, and Watermarking of Multimedia Contents VI, San Jose (SSWMC), California, USA*, volume 5306 of *Proceedings of SPIE*, pages 634–641. SPIE, January 2004.

[86] P.A. Viola and M.J. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), Kauai, HI, USA*, pages 511–518. IEEE Computer Society, December 2001.

[87] N. Weaver, V. Paxson, S. Staniford, and R. Cunningham. A taxonomy of computer worms. In *Proceedings of the 2003 ACM workshop on Rapid Malcode (WORM '03), Wyndham City Center Washington DC, USA*, pages 11–18, New York, NY, USA, October 2003. ACM.

[88] T.D. Wu. The secure remote password protocol. In *Proceedings of the Network and Distributed System Security Symposium (NDSS 1998), San Diego, California, USA*. The Internet Society, March 1998.

[89] K. Zeger and A. Gersho. Number of nearest neighbors in a euclidean code. *IEEE Transactions on Information Theory*, 40(5):1647–1649, September 1994.

[90] W. Zhang, Y.J. Chang, and T. Chen. Optimal thresholding for key generation based on biometrics. In *Proceedings of the IEEE 2004 International Conference on Image Processing (ICIP 2004), Singapore*, pages 3451–3454. IEEE Computer Society, October 2004.

Samenvatting

Biometrische beveiligingssystemen die de identiteit van een persoon verifiren door het scannen van vingers, handen, oog of gezicht worden steeds meer toegepast. Daardoor is de biometrie een van de snelst groeiende industrien. Toepassingen van biometrie omvatten nationale veiligheid (bijvoorbeeld het Europese paspoort), fysieke toegang tot diverse faciliteiten (banken, pretparken, kantoorgebouwen, computersystemen, etc.), gezondheidszorg en overheidsdiensten.

Het gebruik van biometrie voor authenticatie van personen is gemakkelijker dan bestaande methoden zoals passwords en PINcodes (er hoeft niets meegenomen of onthouden te worden). Nog een belangrijk voordeel van biometrische authenticatie is dat het gebeurtenissen aan een gebruiker verbindt (passwords of pasjes kunnen verloren of gestolen worden). Ook wordt het steeds meer maatschappelijk geaccepteerd en dalen de kosten. Biometrische authenticatie vereist het vergelijken van een geregistreerde biometrische opname (biometrische template) met een momentopname (bijvoorbeeld een vingerafdruk die bij het inloggen opgenomen wordt).

Biometrische authenticatie is echter niet perfect, en de uitvoer van een biometrisch authenticatiesysteem kan fouten vertonen door beperkingen van het classificatie algoritme, slechte kwaliteit van de opnamen, of manipulatie van het systeem door een indringer. Alhoewel biometrische authenticatie primair bedoeld is voor het versterken van de beveiliging, leidt het opslaan van biometrische gegevens in een database tot nieuwe beveiligings- en privacyrisico's, die toenemen als de database met een netwerk verbonden is. Dit is in de meeste praktijksituaties het geval.

De meest ernstige bedreigingen zijn: *identiteitsdiefstal*, waarbij een aanvaller templates uit een database steelt en een synthetisch biometrisch kenmerk maakt dat bij authenticatie geaccepteerd wordt; *onherroepelijkheid*, hetgeen betekent dat biometrische gegevens niet kunnen worden bijgewerkt of heruitgegeven wanneer zij gecompromitteerd zijn; *privacy*, hetgeen duidt op het vrijgeven van gevoelige persoonlijke informatie zonder toestemming van de eigenaar. Een oplossing voor deze bedreigingen is het toepassen van technieken voor templatebescherming, die

het moeilijk maken voor een aanvaller om de biometrische gegevens uit de templates te achterhalen.

Dit proefschrift beschouwt beveiligingsaspecten van biometrische authenticatie en draagt oplossingen aan om het risico te beperken dat een aanvaller misbruik maakt van biometrische gegevens of delen van biometrische systemen omzeilt om zijn kwaadaardige doelen te verwezenlijken.

Onze bijdrage bestaat uit drie delen. Ten eerste introduceren we de 3W-tree, een analyse-instrument om voor een biometrisch systeem *kritieke aanvalsscenario's te identificeren*. We passen het 3W-tree ontwerpinstrument toe op het SmartGun biometrisch herkenningssysteem met als doel het identificeren van kritieke beveiligingsproblemen. Ten tweede verkennen we de uitdagingen van *veilige templatebescherming*, die zowel theoretisch als praktisch zijn, en we dragen voor een gedeelte van de problemen oplossingen aan. Ten derde presenteren we een praktische oplossing voor *het veilig verzenden van templates*, wat het mogelijk moet maken de biometrische kenmerken tussen twee biometrische apparaten te versturen wanneer er geen beveiligingsinfrastructuur aanwezig is en de gebruikers geen beveiligingsexperts zijn.

Titles in the IPA Dissertation Series since 2002

M.C. van Wezel. *Neural Networks for Intelligent Data Analysis: theoretical and experimental aspects*. Faculty of Mathematics and Natural Sciences, UL. 2002-01

V. Bos and J.J.T. Kleijn. *Formal Specification and Analysis of Industrial Systems*. Faculty of Mathematics and Computer Science and Faculty of Mechanical Engineering, TU/e. 2002-02

T. Kuipers. *Techniques for Understanding Legacy Software Systems*. Faculty of Natural Sciences, Mathematics and Computer Science, UvA. 2002-03

S.P. Luttik. *Choice Quantification in Process Algebra*. Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2002-04

R.J. Willemen. *School Timetable Construction: Algorithms and Complexity*. Faculty of Mathematics and Computer Science, TU/e. 2002-05

M.I.A. Stoelinga. *Alea Jacta Est: Verification of Probabilistic, Real-time and Parametric Systems*. Faculty of Science, Mathematics and Computer Science, KUN. 2002-06

N. van Vugt. *Models of Molecular Computing*. Faculty of Mathematics and Natural Sciences, UL. 2002-07

A. Fehnker. *Citius, Vilius, Melius: Guiding and Cost-Optimality in Model Checking of Timed and Hybrid Systems*. Faculty of Science, Mathematics and Computer Science, KUN. 2002-08

R. van Stee. *On-line Scheduling and Bin Packing*. Faculty of Mathematics and Natural Sciences, UL. 2002-09

D. Tauritz. *Adaptive Information Filtering: Concepts and Algorithms*. Faculty of Mathematics and Natural Sciences, UL. 2002-10

M.B. van der Zwaag. *Models and Logics for Process Algebra*. Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2002-11

J.I. den Hartog. *Probabilistic Extensions of Semantical Models*. Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2002-12

L. Moonen. *Exploring Software Systems*. Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2002-13

J.I. van Hemert. *Applying Evolutionary Computation to Constraint Satisfaction and Data Mining*. Faculty of Mathematics and Natural Sciences, UL. 2002-14

S. Andova. *Probabilistic Process Algebra*. Faculty of Mathematics and Computer Science, TU/e. 2002-15

Y.S. Usenko. *Linearization in μCRL*. Faculty of Mathematics and Computer Science, TU/e. 2002-16

J.J.D. Aerts. *Random Redundant Storage for Video on Demand*. Faculty of Mathematics and Computer Science, TU/e. 2003-01

M. de Jonge. *To Reuse or To Be Reused: Techniques for component composition and construction*. Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2003-02

J.M.W. Visser. *Generic Traversal over Typed Source Code Representations*. Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2003-03

S.M. Bohte. *Spiking Neural Networks*. Faculty of Mathematics and Natural Sciences, UL. 2003-04

T.A.C. Willemse. *Semantics and Verification in Process Algebras with Data and Timing*. Faculty of Mathematics and Computer Science, TU/e. 2003-05

S.V. Nedea. *Analysis and Simulations of Catalytic Reactions*. Faculty of Mathematics and Computer Science, TU/e. 2003-06

M.E.M. Lijding. *Real-time Scheduling of Tertiary Storage*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2003-07

H.P. Benz. *Casual Multimedia Process Annotation – CoMPAs*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2003-08

D. Distefano. *On Modelchecking the Dynamics of Object-based Software: a Foundational Approach*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2003-09

M.H. ter Beek. *Team Automata – A Formal Approach to the Modeling of Collaboration Between System Components*. Faculty of Mathematics and Natural Sciences, UL. 2003-10

D.J.P. Leijen. *The λ Abroad – A Functional Approach to Software Components*. Faculty of Mathematics and Computer Science, UU. 2003-11

W.P.A.J. Michiels. *Performance Ratios for the Differencing Method*. Faculty of Mathematics and Computer Science, TU/e. 2004-01

G.I. Jojgov. *Incomplete Proofs and Terms and Their Use in Interactive Theorem Proving*. Faculty of Mathematics and Computer Science, TU/e. 2004-02

P. Frisco. *Theory of Molecular Computing – Splicing and Membrane systems*. Faculty of Mathematics and Natural Sciences, UL. 2004-03

S. Maneth. *Models of Tree Translation*. Faculty of Mathematics and Natural Sciences, UL. 2004-04

Y. Qian. *Data Synchronization and Browsing for Home Environments*. Faculty of Mathematics and Computer Science and Faculty of Industrial Design, TU/e. 2004-05

F. Bartels. *On Generalised Coinduction and Probabilistic Specification Formats*. Faculty of Sciences, Di-

vision of Mathematics and Computer Science, VUA. 2004-06

L. Cruz-Filipe. *Constructive Real Analysis: a Type-Theoretical Formalization and Applications*. Faculty of Science, Mathematics and Computer Science, KUN. 2004-07

E.H. Gerding. *Autonomous Agents in Bargaining Games: An Evolutionary Investigation of Fundamentals, Strategies, and Business Applications*. Faculty of Technology Management, TU/e. 2004-08

N. Goga. *Control and Selection Techniques for the Automated Testing of Reactive Systems*. Faculty of Mathematics and Computer Science, TU/e. 2004-09

M. Niqui. *Formalising Exact Arithmetic: Representations, Algorithms and Proofs*. Faculty of Science, Mathematics and Computer Science, RU. 2004-10

A. Löh. *Exploring Generic Haskell*. Faculty of Mathematics and Computer Science, UU. 2004-11

I.C.M. Flinsenberg. *Route Planning Algorithms for Car Navigation*. Faculty of Mathematics and Computer Science, TU/e. 2004-12

R.J. Bril. *Real-time Scheduling for Media Processing Using Conditionally Guaranteed Budgets*. Faculty of Mathematics and Computer Science, TU/e. 2004-13

J. Pang. *Formal Verification of Distributed Systems*. Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2004-14

F. Alkemade. *Evolutionary Agent-Based Economics*. Faculty of Technology Management, TU/e. 2004-15

E.O. Dijk. *Indoor Ultrasonic Position Estimation Using a Single Base Station*. Faculty of Mathematics and Computer Science, TU/e. 2004-16

S.M. Orzan. *On Distributed Verification and Verified Distribution*. Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2004-17

M.M. Schrage. *Proxima - A Presentation-oriented Editor for Structured Documents*. Faculty of Mathematics and Computer Science, UU. 2004-18

E. Eskenazi and A. Fyukov. *Quantitative Prediction of Quality Attributes for Component-Based Software Architectures*. Faculty of Mathematics and Computer Science, TU/e. 2004-19

P.J.L. Cuijpers. *Hybrid Process Algebra*. Faculty of Mathematics and Computer Science, TU/e. 2004-20

N.J.M. van den Nieuwelaar. *Supervisory Machine Control by Predictive-Reactive Scheduling*. Faculty of Mechanical Engineering, TU/e. 2004-21

E. Ábrahám. *An Assertional Proof System for Multithreaded Java - Theory and Tool Support-* . Faculty of Mathematics and Natural Sciences, UL. 2005-01

R. Ruimerman. *Modeling and Remodeling in Bone Tissue*. Faculty of Biomedical Engineering, TU/e. 2005-02

C.N. Chong. *Experiments in Rights Control - Expression and Enforcement*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2005-03

H. Gao. *Design and Verification of Lock-free Parallel Algorithms*. Faculty of Mathematics and Computing Sciences, RUG. 2005-04

H.M.A. van Beek. *Specification and Analysis of Internet Applications*. Faculty of Mathematics and Computer Science, TU/e. 2005-05

M.T. Ionita. *Scenario-Based System Architecting - A Systematic Approach to Developing Future-Proof System Architectures*. Faculty of Mathematics and Computing Sciences, TU/e. 2005-06

G. Lenzini. *Integration of Analysis Techniques in Security and Fault-Tolerance*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2005-07

I. Kurtev. *Adaptability of Model Transformations*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2005-08

T. Wolle. *Computational Aspects of Treewidth - Lower Bounds and Network Reliability*. Faculty of Science, UU. 2005-09

O. Tveretina. *Decision Procedures for Equality Logic with Uninterpreted Functions*. Faculty of Mathematics and Computer Science, TU/e. 2005-10

A.M.L. Liekens. *Evolution of Finite Populations in Dynamic Environments*. Faculty of Biomedical Engineering, TU/e. 2005-11

J. Eggermont. *Data Mining using Genetic Programming: Classification and Symbolic Regression*. Faculty of Mathematics and Natural Sciences, UL. 2005-12

B.J. Heeren. *Top Quality Type Error Messages*. Faculty of Science, UU. 2005-13

G.F. Frehse. *Compositional Verification of Hybrid Systems using Simulation Relations*. Faculty of Science, Mathematics and Computer Science, RU. 2005-14

M.R. Mousavi. *Structuring Structural Operational Semantics*. Faculty of Mathematics and Computer Science, TU/e. 2005-15

A. Sokolova. *Coalgebraic Analysis of Probabilistic Systems*. Faculty of Mathematics and Computer Science, TU/e. 2005-16

T. Gelsema. *Effective Models for the Structure of pi-Calculus Processes with Replication*. Faculty of Mathematics and Natural Sciences, UL. 2005-17

P. Zoeteweij. *Composing Constraint Solvers*. Faculty of Natural Sciences,

Mathematics, and Computer Science, UvA. 2005-18

J.J. Vinju. *Analysis and Transformation of Source Code by Parsing and Rewriting*. Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2005-19

M.Valero Espada. *Modal Abstraction and Replication of Processes with Data*. Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2005-20

A. Dijkstra. *Stepping through Haskell*. Faculty of Science, UU. 2005-21

Y.W. Law. *Key management and link-layer security of wireless sensor networks: energy-efficient attack and defense*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2005-22

E. Dolstra. *The Purely Functional Software Deployment Model*. Faculty of Science, UU. 2006-01

R.J. Corin. *Analysis Models for Security Protocols*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2006-02

P.R.A. Verbaan. *The Computational Complexity of Evolving Systems*. Faculty of Science, UU. 2006-03

K.L. Man and R.R.H. Schiffelers. *Formal Specification and Analysis of Hybrid Systems*. Faculty of Mathematics and Computer Science and Faculty of Mechanical Engineering, TU/e. 2006-04

M. Kyas. *Verifying OCL Specifications of UML Models: Tool Support and Compositionality*. Faculty of Mathematics and Natural Sciences, UL. 2006-05

M. Hendriks. *Model Checking Timed Automata - Techniques and Applications*. Faculty of Science, Mathematics and Computer Science, RU. 2006-06

J. Ketema. *Böhm-Like Trees for Rewriting*. Faculty of Sciences, VUA. 2006-07

C.-B. Breunesse. *On JML: topics in tool-assisted verification of JML programs*. Faculty of Science, Mathematics and Computer Science, RU. 2006-08

B. Markvoort. *Towards Hybrid Molecular Simulations*. Faculty of Biomedical Engineering, TU/e. 2006-09

S.G.R. Nijssen. *Mining Structured Data*. Faculty of Mathematics and Natural Sciences, UL. 2006-10

G. Russello. *Separation and Adaptation of Concerns in a Shared Data Space*. Faculty of Mathematics and Computer Science, TU/e. 2006-11

L. Cheung. *Reconciling Nondeterministic and Probabilistic Choices*. Faculty of Science, Mathematics and Computer Science, RU. 2006-12

B. Badban. *Verification techniques for Extensions of Equality Logic*. Faculty of Sciences, Division of Mathe-

matics and Computer Science, VUA. 2006-13

A.J. Mooij. *Constructive formal methods and protocol standardization*. Faculty of Mathematics and Computer Science, TU/e. 2006-14

T. Krilavicius. *Hybrid Techniques for Hybrid Systems*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2006-15

M.E. Warnier. *Language Based Security for Java and JML*. Faculty of Science, Mathematics and Computer Science, RU. 2006-16

V. Sundramoorthy. *At Home In Service Discovery*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2006-17

B. Gebremichael. *Expressivity of Timed Automata Models*. Faculty of Science, Mathematics and Computer Science, RU. 2006-18

L.C.M. van Gool. *Formalising Interface Specifications*. Faculty of Mathematics and Computer Science, TU/e. 2006-19

C.J.F. Cremers. *Scyther - Semantics and Verification of Security Protocols*. Faculty of Mathematics and Computer Science, TU/e. 2006-20

J.V. Guillen Scholten. *Mobile Channels for Exogenous Coordination of Distributed Systems: Semantics, Implementation and Composition*. Faculty of Mathematics and Natural Sciences, UL. 2006-21

H.A. de Jong. *Flexible Heterogeneous Software Systems*. Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2007-01

N.K. Kavaldjiev. *A run-time reconfigurable Network-on-Chip for streaming DSP applications*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2007-02

M. van Veelen. *Considerations on Modeling for Early Detection of Abnormalities in Locally Autonomous Distributed Systems*. Faculty of Mathematics and Computing Sciences, RUG. 2007-03

T.D. Vu. *Semantics and Applications of Process and Program Algebra*. Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2007-04

L. Brandán Briones. *Theories for Model-based Testing: Real-time and Coverage*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2007-05

I. Loeb. *Natural Deduction: Sharing by Presentation*. Faculty of Science, Mathematics and Computer Science, RU. 2007-06

M.W.A. Streppel. *Multifunctional Geometric Data Structures*. Faculty of Mathematics and Computer Science, TU/e. 2007-07

N. Trčka. *Silent Steps in Transition Systems and Markov Chains*. Faculty of Mathematics and Computer Science, TU/e. 2007-08

R. Brinkman. *Searching in encrypted data*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2007-09

A. van Weelden. *Putting types to good use*. Faculty of Science, Mathematics and Computer Science, RU. 2007-10

J.A.R. Noppen. *Imperfect Information in Software Development Processes*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2007-11

R. Boumen. *Integration and Test plans for Complex Manufacturing Systems*. Faculty of Mechanical Engineering, TU/e. 2007-12

A.J. Wijs. *What to do Next?: Analysing and Optimising System Behaviour in Time*. Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2007-13

C.F.J. Lange. *Assessing and Improving the Quality of Modeling: A Series of Empirical Studies about the UML*. Faculty of Mathematics and Computer Science, TU/e. 2007-14

T. van der Storm. *Component-based Configuration, Integration and Delivery*. Faculty of Natural Sciences, Mathematics, and Computer Science,UvA. 2007-15

B.S. Graaf. *Model-Driven Evolution of Software Architectures*. Faculty of Electrical Engineering, Mathematics, and Computer Science, TUD. 2007-16

A.H.J. Mathijssen. *Logical Calculi for Reasoning with Binding*. Faculty of Mathematics and Computer Science, TU/e. 2007-17

D. Jarnikov. *QoS framework for Video Streaming in Home Networks*. Faculty of Mathematics and Computer Science, TU/e. 2007-18

M. A. Abam. *New Data Structures and Algorithms for Mobile Data*. Faculty of Mathematics and Computer Science, TU/e. 2007-19

W. Pieters. *La Volonté Machinale: Understanding the Electronic Voting Controversy*. Faculty of Science, Mathematics and Computer Science, RU. 2008-01

A.L. de Groot. *Practical Automaton Proofs in PVS*. Faculty of Science, Mathematics and Computer Science, RU. 2008-02

M. Bruntink. *Renovation of Idiomatic Crosscutting Concerns in Embedded Systems*. Faculty of Electrical Engineering, Mathematics, and Computer Science, TUD. 2008-03

A.M. Marin. *An Integrated System to Manage Crosscutting Concerns in Source Code*. Faculty of Electrical Engineering, Mathematics, and Computer Science, TUD. 2008-04

N.C.W.M. Braspenning. *Model-based Integration and Testing of High-tech Multi-disciplinary Systems*. Faculty of Mechanical Engineering, TU/e. 2008-05

M. Bravenboer. *Exercises in Free Syntax: Syntax Definition, Parsing,*

and Assimilation of Language Conglomerates. Faculty of Science, UU. 2008-06

M. Torabi Dashti. *Keeping Fairness Alive: Design and Formal Verification of Optimistic Fair Exchange Protocols*. Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2008-07

I.S.M. de Jong. *Integration and Test Strategies for Complex Manufacturing Machines*. Faculty of Mechanical Engineering, TU/e. 2008-08

I. Hasuo. *Tracing Anonymity with Coalgebras*. Faculty of Science, Mathematics and Computer Science, RU. 2008-09

L.G.W.A. Cleophas. *Tree Algorithms: Two Taxonomies and a Toolkit*. Faculty of Mathematics and Computer Science, TU/e. 2008-10

I.S. Zapreev. *Model Checking Markov Chains: Techniques and Tools*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2008-11

M. Farshi. *A Theoretical and Experimental Study of Geometric Networks*. Faculty of Mathematics and Computer Science, TU/e. 2008-12

G. Gulesir. *Evolvable Behavior Specifications Using Context-Sensitive Wildcards*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2008-13

F.D. Garcia. *Formal and Computational Cryptography: Protocols, Hashes and Commitments*. Faculty of Science, Mathematics and Computer Science, RU. 2008-14

P. E. A. Dürr. *Resource-based Verification for Robust Composition of Aspects*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2008-15

E.M. Bortnik. *Formal Methods in Support of SMC Design*. Faculty of Mechanical Engineering, TU/e. 2008-16

R.H. Mak. *Design and Performance Analysis of Data-Independent Stream Processing Systems*. Faculty of Mathematics and Computer Science, TU/e. 2008-17

M. van der Horst. *Scalable Block Processing Algorithms*. Faculty of Mathematics and Computer Science, TU/e. 2008-18

C.M. Gray. *Algorithms for Fat Objects: Decompositions and Applications*. Faculty of Mathematics and Computer Science, TU/e. 2008-19

J.R. Calamé. *Testing Reactive Systems with Data - Enumerative Methods and Constraint Solving*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2008-20

E. Mumford. *Drawing Graphs for Cartographic Applications*. Faculty of Mathematics and Computer Science, TU/e. 2008-21

E.H. de Graaf. *Mining Semi-structured Data, Theoretical and Experimental Aspects of Pattern Evalua-*

tion. Faculty of Mathematics and Natural Sciences, UL. 2008-22

R. Brijder. *Models of Natural Computation: Gene Assembly and Membrane Systems*. Faculty of Mathematics and Natural Sciences, UL. 2008-23

A. Koprowski. *Termination of Rewriting and Its Certification*. Faculty of Mathematics and Computer Science, TU/e. 2008-24

U. Khadim. *Process Algebras for Hybrid Systems: Comparison and Development*. Faculty of Mathematics and Computer Science, TU/e. 2008-25

J. Markovski. *Real and Stochastic Time in Process Algebras for Performance Evaluation*. Faculty of Mathematics and Computer Science, TU/e. 2008-26

H. Kastenberg. *Graph-Based Software Specification and Verification*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2008-27

I.R. Buhan. *Cryptographic Keys from Noisy Data Theory and Applications*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2008-28