Cryptographic Credentials with Privacy-Preserving Biometric Bindings

David Bissessar

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements for the degree of

Master of Computer Science

Under the auspices of the Ottawa-Carleton Institute for Computer Science

University of Ottawa
Ottawa, Ontario, Canada

Abstract

Cryptographic credentials allow user authorizations to be granted and verified. and have such applications as e-Passports, e-Commerce, and electronic cash. This thesis proposes a privacy protecting approach of binding biometrically derived keys to cryptographic credentials to prevent unauthorized lending. Our approach builds on the 2011 work of Adams, offering additional benefits of privacy protection of biometric information, generality on biometric modalities, and performance. Our protocol integrates into Brands' Digital Credential scheme, and the Anonymous Credentials scheme of Camenisch and Lysyanskaya. We describe a detailed integration with the Digital Credential Scheme and sketch the integration into the Anonymous Credentials scheme. Security proofs for non-transferability, correctness of ownership, and unlinkability are provided for the protocol's instantiation into Digital Credentials.

Our approach uses specialized biometric devices in both the issue and show protocols. These devices are configured with our proposed primitive, the fuzzy extractor indistinguishability adaptor which uses a traditional fuzzy extractor to create and regenerate cryptographic keys from biometric data and IND-CCA2 secure encryption protect the generated public data against multiplicity attacks. Pedersen commitments are used to hold the key at issue and show time, and A zero-knowledge proof of knowledge is used to ensure correspondence of key created at issue-time and regenerated at show-time. The above is done in a manner which preserves biometric privacy, as and delivers non-transferability of digital credentials.

The biometric itself is not stored or divulged to any of the parties involved in the protocol. Privacy protection in multiple enrollments scenarios is achieved by the fuzzy extractor indistinguishability adapter. The zero knowledge proof of knowledge is used in the showing protocol to prove knowledge of values without divulging them.

Dedication

In loving memory of

Daniel Bissessar

March 11, 2007 – January 1, 2012.

Danny you have taught me much, I will never be the same. We started this work together, and I carry it on now in your memory. This work started out as a statement in the spirit of creativity, invention, and doing cool stuff. It was meant to be an example that all things are possible. Now that you are gone, it becomes a symbol of survival, of perseverance, of acceptance; of life broken and reconstructed.

Danny, you are a beautiful and inspiring person. Dear Daniel, thank you for having touched our lives, for having transformed us, for having given me the knowledge of happiness, love and delight, for having brought back the child inside me, and for having given to that child, finally, a wonderful childhood.

You and I had the blessing of a beautiful relationship. We treasured and enjoyed our time together when we were together. I am so blessed to have known you and had the chance to have tasted life so completely with you. We lived each day to the fullest, for that I am forever thankful. Thank Heaven for the experiences and the memories. Thank you for being such a great partner, for the chances of working on all our projects together, for the chance of talking for hours about everything under the sun (as well as things above, beyond and including the sun). Thank you for having taught me the meaning of best friends, and for having been mine.

Thank you for the gift you have given us. Thank you for your deep love, your dedication. You are our champion. We were transformed by you during your life, and our transformation continues also after your passing away. Thank you for the happiness we have shared.

God Bless You My Little Friend

Acknowledgment

I would like to express my gratitude to my supervisor, Dr. Carlisle Adams, for both the professional and the human relationship developed during our work together. I am privileged to have worked with Dr. Adams. Each step of the way, the work has been a pleasure, an adventure.

Dr. Grace Wu, Dr. Dmitry Gorodnichy, Dr. John Oommen, Father Maurice Gagnon, Father Freely Muthukudarachchi, and Sister Rose Mary O'Toole -- Thank you for valued friendship, care, and consultation at crucial times.

To Cindy Zhang, thank you for the chance of sharing this journey with you, of learning together, growing together, leaning on each other, and picking each other up. As you know, more vividly than many, there may be points where life takes your health, your hope, your reason for existence; points at which it seems impossible that happiness will ever return or that there may ever be a reason to live again. God and Life have a manner which is not predictable. Dear Cindy, my blessing in all that is to come for you. Hoping the flower of wonder and delight may bloom again in your life.

To my parents, Winston and Laurette Bissessar: thank you for your love, faith, courage, wisdom, nurturing and support. Thank you for providing the living example of commitment to family, and perseverance through the challenges of life. How could I have done any of it without you?

Friends and beloved, thank you. Amazing people surrounding me. Thank you for your humour, your wisdom and your support throughout: past, present and future. Thank you as well for the dynamics of that support: sometimes in the background… gentle and subtle, sometimes right upfront…. strong and confident. God bless you all.

To Samantha, Alexander and Daniel Bissessar - God bless you. It forever astounds me, the amount of happiness, healing and inspiration you give me. It's always present, even in the smallest things.

Anything is possible… Keep going…..! Keeeeeeeeeep going!

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

This thesis presents a privacy-preserving method to use biometrics to prevent the lending of cryptographic credentials. The current cryptographic credentials schemes suffer the weakness that a credential can be lent between users. This allows unauthorized access to privilege or service. This problem is referred to as the transferability problem. Non-transferability in cryptographic credentials typically is achieved through enhancements through disincentives or by using biometrics. The use of biometrics immediately brings about concerns regarding privacy. Biometric-based solutions necessarily introduce a scanner to capture the biometric. To date little research has focused on biometric binding which prevents lending, yet preserves privacy.

1.1. Motivation

This research is pertinent from the perspective of the individual, online service providers, government agencies, and infrastructure providers.

From the perspective of the service provider, in the online economy, when a subscription or privilege is issued to an individual, it is important to have control over the transferability of that privilege. Many authentication methods suffer from the problem of transferability: the combination of user id and password, for example, can easily be transferred and used by multiple individuals. Various cryptographic credentials also suffer from the weakness that they can be copied and transferred to unauthorized users. To protect from this, a technology is needed which ensures that a privilege given to one individual is used only by that individual, or, that the person claiming a right, is indeed the person to whom the right was granted.

From the individual's perspective, the use of online systems has become an integral part of everyday life. Technology preventing transfer of digital credentials also prevents theft of credentials, protecting the user's credentials from unauthorized

use by other persons. The benefit is evident where there are constraints on the number of times a privilege is used (such as digital cash or medical prescriptions), or where an accountability is attached to the use of a credential (such as access control, or a driver's license). The approaches described in this thesis give the user confidence that issued credentials may not be used by unauthorized users.

From Government's perspective there is an interesting interplay of goals. On one hand, agencies must offer certification documents and provide services in a manner that is reliable enough for law to be enforced and public security to be maintained. On the other hand, systems must be constructed according to respecting privacy policies and legislation.

Infrastructure providers have interest of innovating products and technologies in this area. For example, Microsoft's U-Prove implements the digital credential scheme discussed in this thesis, and IBM's Idemix implements anonymous credentials [CL01][CL03]. Both products are available for open source download.

Binding cryptographic credentials to the owning individual through biometrics provides the security and accountability required by government agency and service provider organization: doing so in a manner which preserves privacy of identity and biometric meets the goals of individual users, and privacy protection agencies.

1.2. General Introduction to Credential Systems, and Biometric Authentication

1.2.1 Scenarios and Applications

A number of scenarios are commonly provided which motivate the applications and issues driving cryptographic credentials. The Prime Life project, part of the European Commission's 7th Framework Project, features a web resource and numerous associated documents which provide an introduction to scenarios and motivating privacy concerns driving cryptographic credentials [PL11][CP07]. Some applications include anonymous login, government identity cards, anonymous age verification, anonymous opinion polls, and anonymous digital cash.

1.2.2 General Concepts Definitions

1.2.2.1 Main Entities Roles and Actions

In its simplest form, a cryptographic credential system involves users and service providers, interacting to issue and show credentials. This section gives an overview of these entities and operations.

1) **Credential.** A credential is an object granted to a user which represents a certified attribute or privilege which can be later claimed for access or usage privilege. A credential is granted by an issuing organization to an individual user, and shown at a later date to a verifying organization, which may grant access or special permissions based on the specifics of the credential. A credential may have usage policies such as validity duration, or number of allowed uses, as will be discussed within this section.
2) **User.** The user participates in issue and show transactions with service providing organizations. The user holds personal attributes, credentials and associated data, playing the role of credential applicant, and credential shower in the system use cases. Within the scope of this work, the user is assumed to have a biometric measurement, and is assumed to be malicious in the sense that he seeks to collude with other entities in such a way to lend credentials and maximize privileges in the system, through any means possible. Although generally referred to as an individual user informal coalitions or organizations may play the part of the user– any entity which can be granted credentials and may later receive privileges based on ownership of credential can play the role of the user.
3) **Issuer.** The issuer is generally an organization, to which the individual applies for a credential. As part of issuing a credential, the issuer may verify the validity of the user's personal attributes, verify any required historic records, and cryptographically sign a credential to grant it to the credential applicant. In some proposals, the issuer is also able to revoke a previously issued credential.

   In this thesis, the issuer interfaces with a biometric device which records an enrollment biometric sample of the user, and converts it to a cryptographic commitment. The issuer then binds the commitment into the credential for signing

and issuing. This value assists in the process of authenticating the credential applicant during the show protocol.

4) **Verifier.** The verifier is the organization with whom the user participates in the showing protocol. The verifier makes sure that the presented credential is properly signed and grants the claimed privilege. In the protocols in this thesis, the verifier has access to a biometric device, which assists in the process of authenticating the user presenting the credential. *Issuers* and *Verifiers* may be generically referred to as *Service Providers*. While it is convenient to think of verifiers as organizations, the role of verifier can be taken by a person: In a military setting, for example, when two persons could interact to verify rank and seniority, while both parties are individuals, one user takes on the role of the verifier in the protocol.
5) **Issuance of Credential**. The issuance of a credential occurs between an organization and an individual. The credential represents the Issuer's certification that an individual possesses a particular quality or set of qualities. Having been issued a credential, a user may show it at a later date to a different organization (the *"Verifier"*).
6) **Showing of a Credential.** An individual possessing a particular credential shows it to a verifying organization, to claim a privilege. The verifying organization checks authenticity of the credential, and validity of the claimed attribute or privilege.

1.2.2.2 Possible additional entities and features

The basic model we have presented for a credential system allows the user to obtain a credential from an issuer, and show it at a later time to a different verifying organization. Additional functionality has been proposed in the literature and can be achieved in a credential system. Not all possible features are present (or desirable) within any one credential system. A general overview of some additional features possible is presented here.

1) Pseudonym. A pseudonym is a moniker between an individual user and an organization. It hides the identity of the user, yet allows an ongoing relationship

between the individual and the service provider. A user may hold many pseudonyms.

2) **Central Authority.** Some systems include a participant playing the role of a central authority, or a trusted central party. This actor is not necessarily desirable, and not always necessary, the trusted central party may go under various names and may perform various duties. For example Chaum includes a third party for pseudonym verification and issuance, and credential issuance [Cha85], [CE86]. Brands' also uses this party as a credential issuer and optional enforcement of credential use constraints [Bra00]. The central authority has been used to verify pseudonyms, to issue signature keys, and as a watchdog ensuring proper protocol behavior holding the ability to revoke anonymity.
3) **System Setup and Modification**. At system initialization, a number of global parameters and possibly entities may need to be configured. The system presented in this thesis, for example, includes the configuration of group parameters (2.2.2.2.1) onto biometric devices (3.2) in order to create Pederson commitments (2.2.4.3).
4) **Entity Setup and Modification**. Distinct from values which must be set at system initialization, it may also be necessary to initialize certain values at the time a new entity is added to a system. This can include, for example, an individual user selecting a secret key, an issuing organization publishing a public key, or as in the protocol proposed in this thesis, the provisioning of parameters onto a biometric sensing device.
5) **Selection of Pseudonym**. If a system includes pseudonyms, these must be selected, verified as valid, and accepted for use in a relationship between an individual and an organization. Selection of pseudonyms may involve a trusted third party (for example, the certificate authority [CL01], or the signing authority [CH85] [CE86]); or it may occur between the two entities with whom the pseudonym will be used, or the selection of a pseudonym may rest entirely with the user. Selection of pseudonyms is not present or needed in all systems.
6) **Transferral of Credential between Pseudonyms**. In a system where pseudonyms are used, it may be necessary for a user to transfer credentials between

pseudonyms, or to prove claims which span relationships with different agencies. For example, the user in question holds multiple credentials: a valid vehicle registration issued by the Ministry of Transport; valid coverage for liability issued by an Insurance provider, and a clean driving record (or no negative credentials) issued by applicable Law Enforcement Agencies.

At the time of this writing, it was a limitation of IDEMIX that propositions proving credentials issued by more than one credential issuer were not supported.

7) **Revocation of Credential.** Some systems allow a credential to be revoked. This may be important in the case of a credential which indicates a privilege granted due to compliance to a set of behaviours or regulations, such as a driver's license, or a fully paid account balance. In recognition of such needs, practical systems such as IDEMIX build these mechanisms for this.
8) **Fixed-Duration Credential.** In some systems it may be possible set a date after which the credential expires. A subscription based system may, for example, be interested in issuing a fixed duration credential which allows a user access to resources for a limited time.
9) **Frequency-Constrained Credential**. It may be desirable for the issuing agency to specify constraints on the number of times a credential can be shown. This may be useful, for example, in a credential representing a medical prescription having limited repeats.
10) **Delegation of Credentials**. In some systems, it may be desirable to allow controlled delegation of credential. A credential may, for example, be granted to a manager, and delegated to an employee (perhaps for a fixed duration, or for revocation at a later date). The delegation chain can be verified as valid, without divulging identity.
11) **Revocation of Anonymity**. Systems have been proposed which allow means by which an entity's anonymity may be revoked as a result of her own actions, or by another entity under some conditions

While the list presented here is not exhaustive, it gives an appreciation of the functionality beyond the basic issue and show functionality which may be present, in various combinations, in credential systems.

1.2.3 General Architecture

This section presents a general architecture for a credential system. This is done to illustrate the general participants and functionalities which may be involved in a credential system. While the architecture described here does not represent any one specific proposal from the scientific literature, it intends to capture the essential entities and functionalities involved in credential systems to set the stage for the literature survey and technical discussions to follow in this thesis.

Our generalized credential architecture presents a decentralized system in which entities of various types interact as peers to perform legitimate system activities or to mount attacks aimed at compromising system security. We model the system as a fully-connected graph with nodes representing the system entities and the edges between these nodes representing the interactions between them.

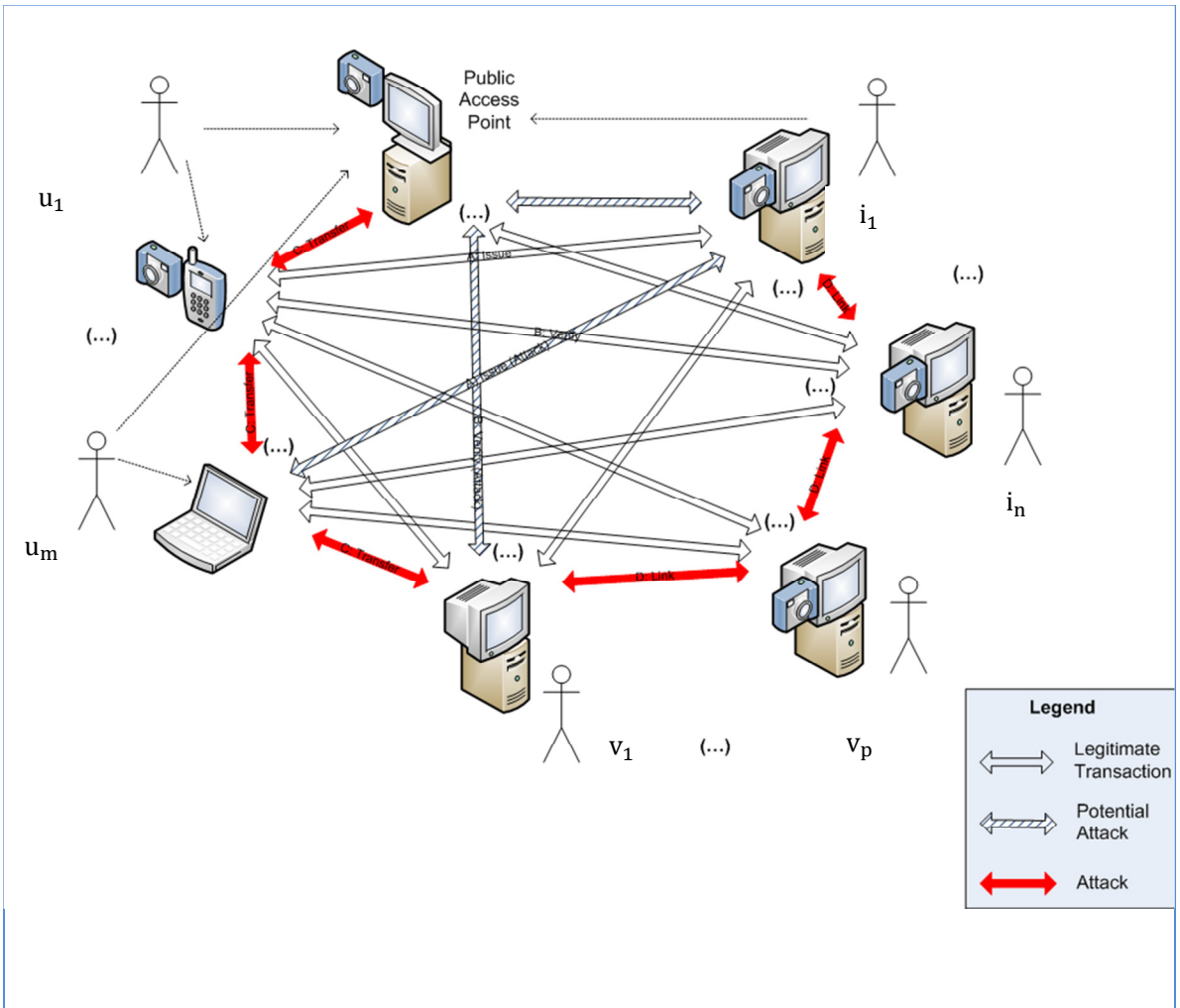

Figure 1. General Architecture for Credential System

Figure 1 presents the credential system as a fully connected graph: a network of interacting peers. The general interests and behavior of these entities will be described in the following section.

Nodes on the graph are given a type according to the type of system entity represented: users $\boldsymbol{U} = \{U_1, \dots, U_m\}$, issuers $\boldsymbol{I} = \{I_1, \dots, I_n\}$, and verifiers $\boldsymbol{V} = \{V_1, \dots, V_p\}$. A node's type dictates the protocols it can perform and the private data which is visible to it.

1.2.3.1 Characteristics of interactions

Edges in the graph represent the interactions between pairs of system entities in particular roles. An edge can be represented as a 2-tuple (e_i, e_r), where $e_i, e_r \in$

$\{\boldsymbol{U}, \boldsymbol{I}, \boldsymbol{V}\}$ represent the initiating and responding entities in a protocol. Since there are a set number of roles, each with clear behaviour, we can infer certain characteristics about the general nature of an interaction based on the roles involved. Table 1 presents this characterisation. In general we see that a) interactions initiating from $\boldsymbol{U}$ to $\boldsymbol{I}$ or $\boldsymbol{V}$ are typically legitimate, but also potentially malicious; b) interactions between $\boldsymbol{I}$ or $\boldsymbol{V}$ or between users $\boldsymbol{U}$ are generally collusive aimed at compromising system security.

Interaction		Characteristics
User to Issuer	$(u, i) \in U \times I$	Issue or Attack
User to Verifier	$(\mathrm{u}, \mathrm{v}) \in \mathrm{U} \times \mathrm{V}$	Show or Attack
User to User	$(a, b) \in U \times U \mid a \neq b$	Attack
Service Provider to Service Provider	$(s_i, s_j) \in S \times S \mid s_i \neq s_j$	Attack

Table 1 Categorization of Interactions

1.2.3.1.1. Credential "Issue" Protocol

The issue protocol is a legitimate system interaction in which a user *u* contacts an issuer i, to obtain a credential. We denote the interaction $(\mathrm{u}, \mathrm{i}) \in \mathrm{U} \times \mathrm{I}$. In this protocol, i issues a credential to *u* which *u* can show at a later time to a verifier *v* to claim a privilege.

In general terms, the credential is a data structure which contains the issuing organization's signature on a set of attributes about the individual in question.

In the proposal of this thesis, we augment Brand's issue protocol with a biometric sensor which creates a cryptographic commitment on a biometrically derived key.

An issue protocol between *(u, i)* may be legitimate, but may also be part of an attack. In the attack model presented in this thesis, the attacker is given the resources to corrupt a set of users and use knowledge gained during the issue protocol as part of an attack including the transfer of credentials.

The possibility that the interaction between (u, i) may be legitimate or part of an attack is highlighted in Figure 1 by showing various instances of Issue protocol in different manners: interaction (u_1, i_1) is shown as a legitimate transaction whereas interaction $(\mathrm{u_m}, \mathrm{i_1})$ is shown as malicious.

1.2.3.1.2. Credential Verification Protocol

User and verifier $(\mathrm{u}, \mathrm{v}) \in \mathrm{U} \times \mathrm{V}$ interact using the show protocol, a legitimate protocol in which u presents credential to v as part of a process where v ascertains the validity of the credential and determines the system entitlements of u.

This thesis augments the show protocol of the underlying credential system with a mechanism to verify that the biometrically-derived information obtained at show-time corresponds to information sealed in the credential during the issue protocol (§3.4.3).

As was the case in the issue protocol, it is also possible for the show protocol to be used by an attacker. For example, an attacker may corrupt a collection of service providers and mount a transaction unlinkability attack in attempt to correlate the transactions of an honest user. This will be further discussed in Chapter 4.

To highlight the possibility that *(u,v)* may be legitimate or part of an attack, Figure 1 shows instances of the show protocol in different manners. The interaction (u_1, v_n) is shown as a legitimate transaction, but the interaction (u_m, v_1) is shown as part of an attack.

1.2.3.1.1. Direct User Interaction

It is possible that distinct users $(\mathrm{a}, \mathrm{b}) \in \mathrm{U} \times \mathrm{U} \mid \mathrm{a} \neq \mathrm{b}$, may interact, however, no specific protocol is defined between them. The direct interaction between users generally represents malicious behavior. Possible examples of attacks would in-

clude attempts to gain access to unauthorized privilege by a forging the signature of an issuer, or simply the posting of credential information to public web-site for re-use by unauthorized individuals.

It should also be noted that while our general model does not define as valid any interaction between users, other protocols may include legal user interactions. An example of such a protocol between users is the delegation of credentials (see Section 1.2.2.2, item 10) "Delegation of Credential").

1.2.3.1.1. Interaction between Service Providers

As in the case of the direct interaction of users, when issuers and verifiers interact it typically represents either a collusion between users or an outright attack (for example a transaction linking attack as discussed in Section 4.1.2 "Unlinkability of Transactions").

In summary, as shown in Table 1, legitimate transactions include issue between (u, i) or show between (u, v). The issue and show protocols can also be used in attacks. Interactions between users, or between service providers, are typically malicious. These observations influence the security model we elaborate in Chapter 4.

1.3. Thesis Goals

This thesis has the goal to find a practical approach to solve the problem of transferability of cryptographic credentials. We seek to create a system which:

- Integrates with digital credentials [Bra00] and anonymous credentials [CL01];
- Supports multiple biometric modalities;
- Protects the privacy/confidentiality of biometric data;
- Provides provable security;
- Provides computation and communication efficiency

1.4. Thesis Contributions

This thesis makes the following contributions:

1) **Protocol Extension: Non-transferability for Cryptographic Credentials**

Our proposed protocol integrates biometric devices which are equipped with fuzzy extractors that generate cryptographic keys in the issue and show protocols of existing cryptographic credential schemes to deliver the non-transferability of credentials. We present the integration of our proposed protocol extension on one cryptographic credential protocol: the digital credential scheme of Stefan Brands [Bra00] and sketch applicability to the anonymous credentials protocol of Camenisch and Lysyanskaya. We prove the security of our proposed algorithm and provide a comparative presentation to the non-transferability protocol extension presented by Adams [Ada11]

2) Cryptographic Primitive: Fuzzy Extractor Indistinguishability Adaptor

We propose a primitive called the fuzzy extractor indistinguishability adaptor which supplements traditional fuzzy adaptor constructs to make the public data resistant to multiple use attacks

3) Pedersen Commitments to Hide Biometrically derived keys

This thesis proposes the use of Pedersen commitments in conjunction with a biometrically derived key as a means of identity verification. Due to its property of being perfectly hiding, the Pedersen commitment is secure for public viewing, storage and wire transfer. Due to its binding property, the Pedersen commitment provides assurance that the key which it hides remains unchanged. Combined with a proof of knowledge, Pedersen commitments on biometrically derived keys provide an effective way of identity verification.

4) Privacy preserving biometrics

In previous work, Adams presented a protocol in which the biometric itself was stored on the user's device [Ada11]. In the event that the device was obtained by an attacker, the biometric itself could be obtained, compromising privacy and security. This thesis uses fuzzy extractors to protect the privacy of the user's biometric.

5) Modalities supporting distance metrics other than the Hamming Distance

- In previous work, Adams presented a protocol in which lent itself to biometrics supporting the hamming distance metric [Ada11]. This thesis uses fuzzy extractors to implement biometric similarity calculations, and thus supports biometric

modalities where similarity is measured by hamming distance, edit distance, and set distance.

1.5. Thesis Outline

This thesis is structured as follows:

- Chapter 1 provides an introduction to credential systems, a walkthrough of common terminology and functionality and a general architecture for a peer-to-peer credential system.
- Chapter 2 provides the required technical background for the thesis, including:
    - An introduction to select concepts from the domains of biometrics, number theory, information theory;
    - An overview of the specific algorithmic components used to build the proposed protocol (digital credentials; fuzzy extractors; Pedersen commitments ; proofs of knowledge; IND CCA 2 block encryption) , as well as the 2011 protocol of Adams;
    - Literature reviews of a) the field of cryptographic credentials with focus on non-transferability and b) the field of secure sketches and fuzzy extractors as privacy enhancing technologies for biometrics.
- Chapter 3 describes the algorithmic contributions of this thesis. This includes a discussion of the fuzzy adaptor indistinguishability adaptor, the functioning of the biometric device used in both the issue and show protocols, a demonstration of the proposed protocol in the context of Brands' digital credential scheme and a sketch of the protocol integration with Camenisch Lysyanskaya's anonymous credentials.
- Chapter 4 analyzes the security of the protocol for correctness of ownership, non-transferability, and unlinkability of transactions.
- Chapter 5 compares the proposed protocol with that of Adams [Ada11] with respect to a number of parameters, including communication cost, computation cost, generality with respect to biometric modalities, complexity of architecture, imposed responsibility on sensor, and imposed complexity on user.

- Finally, Chapter 6 summarizes the main findings of the thesis and provides some directions for future research.

Chapter 2. Background

2.1. Introduction

In this section, we present the literature review of the area of cryptographic credentials. We also present literature review in the area of privacy enhancing techniques for biometric data. To do so, we first gather some key theoretical and cryptographic constructs used throughout. This discussion serves to set the context of the present work and provide the reader with an overview of what has been done.

2.2. Technical Concepts

2.2.1 Biometrics

This thesis uses biometric techniques to implement the non-transferability of cryptographic credentials. This section introduces some foundational items required for this thesis including biometric modalities, biometrics and noise, true reject rate and false accept rate in biometric authentication. To reflect common usage the information here largely draws from the definitions found at http://www.biometrics.gov which is the main publically available internet source concerning the activity in biometrics of the US Federal Government [BIOM].

Biometrics. "A general term used alternatively to describe a characteristic or a process. As a characteristic: a measurable biological (anatomical and physiological) and behavioural characteristic that can be used for automatic recognition. As a process: automated methods of recognizing an individual based on measurable biological (anatomical and physiological) and behavioral characteristics". [BIOM]

Biometric Modality. "A type or class of biometric system: for example face recognition, fingerprint recognition, iris recognition" [BIOM]. Separate from the type of system in question, the term biometric modality is generally used interchangeably

with the term “biometric trait”, to describe the particular biological/behavioural characteristic used by the system as the basis of subject comparison or identification. Examples of biometric modalities include fingerprint, face, iris, gait, voice, keyboard dynamics, palm vascular pattern, ocular region, etc. Some systems may use a number of separate modalities in conjunction to make a holistic decision about a subject of interest. Such systems are termed multi-modal systems.

In this thesis we do not focus on a particular biometric modality, but rather assume that the system deployer has selected a modality for which underlying distance metrics and receiver operating characteristics have been tuned and balanced to meet operational needs.

Authentication: In biometrics “authentication” is sometimes used as a synonym for verification, a function in which the biometric system attempts to confirm an individuals claimed identity by comparing a submitted sample to one or more previously enrolled templates. This highlights the standard two phases of a biometric authentication system: “enrollment” where a reference biometric of a subject is recorded and stored, and “verification” where a fresh template is gathered from a human subject and compared to the enrollment template to make a match decision regarding the templates, thus verifying the identity of the subject.

The approach we propose in this thesis for non-transferable digital credentials has similarities to the biometric verification paradigm: “enrollment” occurs during the issue protocol, and “verification” occurs during the show protocol.

Biometric Identification. A biometric identification system, contrasted to a biometric authentication system, performs a one-to-many match of a subject against a gallery of users to find the most similar enrolled subject to the challenge, or the best match.

Match. An algorithmic decision that, based on their high level of similarity, a biometric sample and a stored template come from the same individual. This generally involves measuring similarity using a distance metric, and comparing to a distance threshold.

Biometric Sample. “Information or computer data obtained from a biometric sensor device. Examples are images of a face or fingerprint” [BIOM]. Inspection of

the biometric sample reveals identifying characteristics about the owner of the biometric, including identity in some cases. Possession of the biometric raw data by the system owner allows a certain ability to migrate between various vendor systems, and template representations (see below). Possession by an attacker of the biometric raw data may allow playback attacks and other security breaches.

Biometric Template. The biometric template is a representation of the biometric sample optimized for system processing tasks such as algorithmic comparison, cross-system data interchange, or storage. It should be noted that the role of biometric template is not privacy protection: while it is not as easily interpreted as the raw biometric sample, by examining the biometric template, it is possible for the observer to deduce identifying features about the owner of the biometric.

Additionally, the loss of template data can be as damaging as loss of biometric raw data, as templates may be used in various system attacks, and may also in some cases be used to reverse engineer aspects the biometric raw data.

Distance. The distance between two biometric templates is an algorithmically produced value which indicates the degree to which these templates differ.

False Accept Rate. The false accept rate is the percentage of occurrence in which a biometric verification task incorrectly reports a positive match for a subject where actual identity does not correspond to claimed identity.

False Reject Rate. The false reject rate is the percentage of occurrence in which a biometric verification task incorrectly reports a negative match for a subject whose actual identity corresponds to the claimed identity.

Assumption 2.1: Trusted device which detects liveness

We assume that the integrity of biometric device cannot be compromised by an attacker, and that it may be trusted by all parties in system to adhere to the specified protocol. We further assume that the biometric device detects liveness of subjects submitting biometric impressions: that playback of data is not possible.

Assumption 2.2: Soundness of Biometric Modality

We assume soundness of the selected biometric modality; that distance metrics are selected such that the probability of true accepts is overwhelming and the

probability of false accept is negligible. If the biometric of the enrolled individual is sufficiently close to the biometric presented at verification time, the identity of individuals is deemed to correspond. Given a population of n users $\mathcal{U} = \{U_1, \ldots, U_n\}$ where $id(U_i)$ represents the identity of a given user, and $b(U_i)$ represents the biometric template of that user $\forall U_i, U_j \in \mathcal{U}$, $dist\left(b(U_i), b(U_j)\right) \leq t \Leftrightarrow id(U_i) = id(U_j)$.

2.2.2 Mathematics and Number Theory

This section introduces some of the foundations from mathematics and number theory upon which the primitives and protocols in this thesis are built. Detailed presentations of the material are available in [MvOV97],[Gol00][Gol04] and [Sho05].

2.2.2.1 Probability distributions

The following definitions are adapted from [Dam99].

Problem feasibility and Polynomial Time Turing Machines

Problem feasibility is expressed relative to standard theoretical complexity measurements. Computations which can be performed in time polynomial in k are considered feasible, whereas problems of super-polynomial complexity are deemed infeasible. The familiar model of the Turing machine is used to represent algorithms. The Turing machine may either be probabilistic or deterministic. The attackers in the security proofs, for example, can be represented as probabilistic polynomial-time Turing machines (PPTs). In this thesis, the interacting entities Users, Issuers, and Verifiers, their protocols and the attackers are viewed as PPTs.

Probabilities can be measured and described between limits of highly unlikely or overwhelming probability. Given security parameter k, a function $f(k)$ is **negligible** if it is smaller than the reciprocal of any polynomial $p(k)$: $f(k)$ is negligible in in k if there exists an integer constant c such that, $f(k) < p(k)^{-c}$ for sufficiently large k. Similarly, a function f(k) is **non-negligible** in k if there exists an integer constant c such that $f(k) > p(k)^{-c}$ for sufficiently large k. A probability 1 - $\epsilon(l)$ is called **overwhelming** if $\epsilon(l)$ is negligible. Events which occur with negligi-

ble probability remain negligible even if the event is repeated polynomially many times.

Statistical Difference. The Statistical Difference between two probability distributions P, and Q is defined as $SD(P, Q) = \sum_y |P(y) - Q(y)|$ where P(y) and Q(y) are the probabilities assigned to *y* by P and Q.

Distinguishability. Distinguishability is defined in terms of probabilistic polynomial-time Turing machines U and V. U_x denotes the probability distribution of U's output given input x.

- U, V are **perfectly indistinguishable**, written $U \sim^p V$, if $U_x = V_x$ for every x.
- U, V are **statistically indistinguishable**, written $U \sim^s V$, if $SD(U_x, V_x)$ is negligible in the length of the string x.
- U, V are **computationally indistinguishable**, written $U \sim^c V$ if a third PPT D (the distinguisher) cannot tell the difference between their outputs. Here, D is responsible for receiving an input *x* or *y* from an unidentified source, then guessing if the input came from U or V. A trend of successful guesses would suggest an advantage in distinguishing U from V. This is represented formulaically by saying that $|P_{U,D}(x) - P_{V,D}(x)|$ is negligible in the length of x where $P_{U,D}(x)$ and $P_{V,D}(x)$ are the probabilities that D outputs the correct guess (U or V as appropriate) when input comes from either of these respectively.

An intractable problem is one that cannot be solved in polynomial time with non-negligible probability of success.

2.2.2.2 Groups

A *group* G consists of a non-empty set of items S = {a,b,c,…} and a binary operation * such that the elements of the group under the group operation exhibit closure and associativity and the group contains both an identity and inverse elements.

Closure on the group states that $\forall a, b \in G, a * b \in G$. Associativity states that $\forall a, b, c \in G, a * (b * c) = (a * b) * c$. The following holds for the Group's *identity* element *e*, $\forall a \in G, a * e = a$. Each item in G has an *inverse* a^{-1} also in

G such that when the group operation is applied between the element and its inverse the identity is obtained: $\forall\, a \in G\, \exists\, a^{-1} \in S\, s.t.\ a * a^{-1} = e$. A group which, beyond closure, associativity, inverse and identity, also exhibits commutativity is called an Abelian group.

A group is *cyclic* if $\exists\, g \in G\ s.t.\ \forall\, b \in G\,, \exists\, x\, s.t.\ g^x = b$. In this case g is called a generator of G, and G is the group generated by g, written as $\langle g \rangle = G$.

The *order of group* G is the number of elements it contains, written $|G|$. If $|G|$ is an Integer, G is a finite group. The order of an element a within G is the smallest $n \in I$ s.t. $a^n = 1$. The order of any element in a finite group divides the order of the group.

2.2.2.2.1. Prime Order Sub-Group

The group G_q underlies many of the cryptographic protocols in this thesis. G_q is a subgroup of Z_p^* of order q where p, q are prime, and where Z_p^* contains the positive integers less than p.

Some useful properties of G_q include that 1) it is easy to test if $a \in Z_p^*$ is also in G_q because $\forall\ a \in Z_p^*,\ a^q \equiv 1\ mod\ p \leftrightarrow a \in G_q$ and 2) It is easy to find a generator for G_q because $\forall\, b \in G_q\, s.t.\, b \neq 1,\ \langle b \rangle = G_q$.

The group G_q is used in the Pedersen Commitments, and Digital Credentials described in this thesis.

2.2.2.2.2. Multiplicative Group

Another group commonly used in cryptographic applications is Z_n^*, the multiplicative group of integers modulo n. This group is also known as the group of residues mod n, where $Z_n^* = \{\, x \in Z_n\ where\ \gcd(x, n) = 1\,\}$. A particular form of the general group Z_n^* where $n = pq$ and p, q are prime has been commonly used in cryptographic protocols and algorithms, notably the RSA cryptosystem.

While the multiplicative group Z_n^* is not directly used in this thesis, it appears in some systems described in the literature review. It should be noted that though we use the group G_q subgroup of Z_p^*, throughout this thesis, digital credentials and our extension of the algorithm can also function in the multiplicative group Z_n^*.

2.2.2.3 Discrete Log

The discrete log problem can be stated as the problem of solving for x in $y = g^x \pmod q$ thus $x = log_g y \ (mod\ q)$ where x and y are values in group G_q where G_q and generator g are as described in Section 2.2.2.2.1 above.

The discrete log is believed to be intractable under certain constructions of G_q (such as choosing p, q primes $st.\ q|p-1$ with security parameters |p|=1024, and |q|= 160).

Assumption 2.3: Under appropriate construction, the DL is intractable

Under particular constructions for G_q, finding the discrete log of a number y is believed to be intractable. One such construction is the subgroup construction, G_q subgroup of Z_p^* as described above.

2.2.2.4 Discrete Log Representation

The discrete log representation of a number h in finite field Z_p is a tuple $\{x_1, \dots, x_l\}$ which when applied as exponents to a set of bases $\{g_1, \dots, g_l\}$ yields that number. We say that tuple $\{x_1, \dots, x_l\}$ is the discrete log representation of h with respect to bases $\{g_1, \dots, g_l\}$ mod q which in this thesis we may abbreviate to $\{x_1, \dots, x_l\}$ = DLREP(h, $\{g_1, \dots, g_l\}$). The (mod q) may be omitted in the notation, but should be clear from context.

Assumption 2.4: Under appropriate construction, the DLRep is intractable

It is computationally hard to find the tuple $\{x_1, \dots, x_l\}$ given p, h, and the bases $\{g_1, \dots, g_l\}$ [Sho05][Bra00]. The problem remains hard even if part of the representational tuple is known, but only one exponent remains secret, in which case, the problem has the hardness of the discrete log problem.

2.2.3 Information Theoretic Background

This section introduces some technical concepts from information theory. Detailed presentations of the material are available in [DRS04], [Huf03] and [Bla83].

2.2.3.1 Metric Space

A metric space *M* is a collection of elements equipped with a distance function, *dis(…)*: $M \times M \rightarrow Z+$. A number of distance metrics are available including the hamming distance, set distance, and edit distance.

Applied to the field of biometrics, the metric space *M* could represent the set of possible biometric templates for a population of users for a given modality (iris codes, for example) thus $M = \{b_1, \ldots . b_k\}$, and $d = dis(b_i, b_j)$ for some $b_i , b_j \in B$ would be the distance as derived by the corresponding distance metric (hamming distance, for example).

2.2.3.2 Hamming Distance

The *hamming distance* $\mathcal{H}D(b, b')$ is a distance metric used on bit strings which returns the number of bit positions in which the strings differ. The *hamming weight* of two vectors is a vector consisting of the component-wise distances of the vectors. For binary strings this corresponds to the exclusive-or of the strings: $\mathcal{H}W(b, b') = b \oplus b'$. The hamming distance can be expressed as the sum of the components in the hamming weight $\mathcal{H}D(b, b'): w = \mathcal{H}W(b, b');\ return\ \sum_{i=0}^{|b|} w_i$.

2.2.3.3 Error Correcting Codes

Error correcting codes where invented in the context of communications to recover messages in which bits may have been corrupted during wire transfer. In general, an error correcting code is a pair of algorithms, encode and decode $< Enc(\ldots),\ Dec(\ldots) >$, where $c = Enc(m)$ maps a message m of length l_1 to a codeword *c* of length l_2 in preparation for transmission over a noisy channel; and $m' = Dec(s)$ accepts a received string *c* and attempts to restore *m*. The received string *s* may be a distortion of the sent codeword *c* or may be an entirely unrelated string: if it is within the code's error correcting distance t, $Dec(\ldots)$ will successfully recreate *m*.

Error correcting codes have found applications in biometrics as a method of factoring out the intra-subject variations (the "noise") that appears between samplings of a person's biometrics. (See 2.2.4.4, Secure Sketches and Fuzzy Extractors)

2.2.3.4 *Predictability, Min-Entropy and Conditional Min-Entropy*

The measures of *predictability* and *min-entropy* from information theory are used to discuss the security of secure sketches and fuzzy extractors, which we will use for the biometric binding in our non-transferability scheme.

Given a random variable A with possible values $\{1, \dots, n\}$, its *predictability* is the probability of its most likely value. A related measure is the *Shannon Entropy* of a random variable, which measures its randomness, its unpredictability. These values are inversely proportional. The Min-Entropy of a random variable is its worst case entropy – its maximum predictability.

	Predictability	**Min-Entropy**
Single Variable	$\mathrm{P}(\mathrm{A}) = \max_{\mathrm{a}}(\Pr[\mathrm{A}=\mathrm{a}])$	$\mathrm{H}_\infty(\mathrm{A}) = -\log(P(A)) = -\log(\max_{\mathrm{a}}(\Pr[\mathrm{A}=\mathrm{a}]))$
Conditional	$\mathrm{P}(\mathrm{A}\|\mathrm{B}=\mathrm{b}]) = max_a(\Pr[A=a\|\,\mathrm{B}=\mathrm{b}])$	$\mathrm{H}_\infty(\mathrm{A}\|\mathrm{B}=\mathrm{b}]) = -\log(\mathrm{P}(\mathrm{A}\|\mathrm{B}=\mathrm{b})) = -\log(max_a(\Pr[A=a\|\,\mathrm{B}=\mathrm{b}]))$
Average Conditional	$\mathrm{P}(\mathrm{A}\|\mathrm{E}_{\mathrm{B}=\mathrm{b}}]) = \mathrm{E}_{\mathrm{B}=\mathrm{b}}\, max_a(\Pr[A=a\|\,\mathrm{B}=\mathrm{b}]) = \mathrm{E}_{\mathrm{B}=\mathrm{b}}\, 2^{-\mathrm{H}_\infty(\mathrm{A}\|\mathrm{B}=\mathrm{b}])}$	$\widetilde{H}_\infty(\mathrm{A}\|\mathrm{B}) = \widetilde{H}_\infty(\mathrm{A}\|\mathrm{E}_{\mathrm{B}=\mathrm{b}}]) = -\log(\mathrm{P}(\mathrm{A}\|\mathrm{E}_{\mathrm{B}=\mathrm{b}}])) = -\log(\mathrm{E}_{\mathrm{B}=\mathrm{b}}\, max_a(\Pr[A=a\|\,\mathrm{B}=\mathrm{b}])) = -\log(\mathrm{E}_{\mathrm{B}=\mathrm{b}}\, 2^{-\mathrm{H}_\infty(\mathrm{A}\|\mathrm{B}=\mathrm{b}])})$

Table 2. Predictability and Min-Entropy

Table 2 presents the relationships between the notions of predictability and min-entropy in three cases: 1) there is one (independent) random variable, 2) there are

two random variables and we are given the value of the second variable, and 3) there are two random variables, but we are only given the distribution of the second variable. The second two definitions extend the notion of min-entropy to the situation where there are two variables: what is the min-entropy of A when the value of B is known? What is the min-entropy of A when only the distribution of B is known?

In the field of biometrics, these concepts come into play when analysing privacy enhancing technologies to determine the attacker's chance of deriving the biometric when examining a protected version of the template. The secure sketches and fuzzy extractors presented in Section 2.2.4.4 use these concepts to discuss the security of biometric and cryptographic key given generated public data.

2.2.4 Cryptographic Primitives

This section will present cryptographic background relevant to this thesis's proposed protocol. Detailed presentations of the material are available in [MvOV97],[Gol00][Gol04]

2.2.4.1 Hash Functions

A *hash function* is a function accepting a "message" of arbitrary length as input and producing an output "digest" of a fixed size. The output of the function is always the same given the same input. A *perfect hash function* is a collision free function H that maps a set S onto a set of integers Z. Algorithms for collision free hashing are presented in [Gol04]. A *cryptographic hash function* is a hash function that produces a fixed-size bit string from an input block of data, such that a small change to the input argument will, with high probability, dramatically change the hash value. Some properties of an ideal cryptographic hash function are that 1) it is easy to compute the hash value for any given message, but infeasible to generate a message that has a given hash, and 2) it is infeasible to modify a message without changing the hash, or 3) to find two different messages with the same hash. A general use of a cryptographic hash function is in producing a fixed size message digest of an input string that can serve as an identifier for the message.

A message authentication code (MAC) is a special type of hash function which consumes two arguments, a message and a secret key, and produces a fixed length digest, which should be infeasible to reproduce without knowing the key.

The protocols in this thesis use hash functions in the issue protocol and verification relation of digital credentials [Bra00]. MACs are used by the IND-CCA2 encryption algorithm [Gold04] which we use to construct the Fuzzy Extractor Indistinguishability Adaptor.

2.2.4.2 Pseudo-Random Functions

Pseudo-random functions (PRF) are a family of functions $\{f_s: \{0,1\}^d \rightarrow \{0,1\}_{s\in\{0,1\}^d}\}$ such that given the seed *s*, the function f_s can be evaluated in polynomial time, but without the seed, it is infeasible to distinguish an oracle on f_s from a truly random function. The seed can act as a key between users: sharing the key, the users share access to the same PRF, otherwise the output seems random.

PRFs were introduced in [GGM87] which showed how a PRF $f_k(x)$ could be created using a cryptographically strong pseudo-random bit generator (CSB). A CSB is a function $r = G(s)$ which accepts an input seed s of length l returning random bit string r of length $2l$. The construction of [GGM87] PRF $f_k(x)$ defines $G_0(s)$ and $G_1(s)$ to be substring functions returning the respectively the left half and right half of $G(s)$. The subscript notation then extends to bit strings where $G_b(s)$ seeds G on s then works through b from left to right, bit-by-bit, invoking G as appropriate on the left or right substring of the previously returned random bit string. The GGM construction of PRF implements $f_k(x)$ by seeding G on k, then walking the bits of x to extract appropriate substrings to re-feed into G. We have in general that $f_k(x) = G_x(k)$. Other constructions of PRF exist, including the Naor-Reingold construction offered in 1997 which uses an array of integers, and successive multiplications followed by an exponentiation of a generator.

This thesis uses Pseudo-Random Functions (PRF) in its encryption approach to provide indistinguishability on the public data stored to assist in the regeneration of the biometrically derived cryptographic key. In this thesis the specific construction used is left open, though we perform our cost analysis using the GGM construction.

2.2.4.3 Pedersen Commitment

The Pedersen commitment [Ped92][Dam99] allows a sender to create a publically storable commitment on a value which irrefutably binds to the value, and also perfectly hides the value from being derived.

The Pedersen commitment scheme has two protocols $C_s = \text{Commit(s, r)} = g^s h^r \ (mod\ p)$ and $(\text{s, r}) = \text{Open}\ (C_s)$ where the secret s is a value from Z_q and random value *r* is uniformly drawn from Z_q. The specification of mod p for Pedersen commitments will be omitted throughout this thesis but should be clear from context.

The Pedersen commitment scheme uses G_q as described in 2.2.2.2.1 above, where p and q are large primes for which $q \mid p-1$ and g and h are generators of G_q for which $\log_g h$ as well as $\log_h g$ are not publicly known

In the protocol presented in this thesis, Pedersen commitments are used to commit to the biometrically derived cryptographic key: the "hiding" property preserves privacy of the key, and the "binding" property ensures security.

Proposition 2.1: Pedersen Commitments are Perfectly Hiding

For any $s \in Z_q$ and for $\text{r} \underset{R}{\leftarrow} Z_q$, $C_s = \text{Commit(s, r)} = g^s h^r \ (mod\ p)$ is uniformly distributed in G_q. No adversarial receiver learns any information about the committed value by looking at the commitment. Or, given a commitment C_s and a set of possible secrets $S = \{s_i\}$ every value s_i is equally likely to be the value committed to by C_s.

Proposition 2.2: Pedersen Commitments are Computationally Binding

Having committed to a value, no adversarial sender can successfully reveal a different value than the one which was committed to. If the sender can find different values x and x' both of which open commitment $C = g^x h^r$, this implies the sender can solve discrete log problem, which is assumed intractable (Assumption 2.).

Proposition 2.3: Pedersen Commitments are Indistinguishable

Given a collection of commitments, it is not possible to determine with any significant advantage whether or not they are commitments on the same secret.

This follows naturally from Proposition 2.1: "Pedersen commitments are perfectly hiding". To illustrate this, consider instead that the opposite of Proposition 2.3 is true: namely, that an adversary has a non-negligible advantage in determining whether two commitments are on the same secret value $|\ Pr[\ sameSecret\ (c1, c2)] - 0.5\ | > e$, where e is negligible and $sameSecret(c1, c2) <-> c1 = commit(s, \_);\ c2 = commit(s, \_)$. An attack on PC1, the perfectly hiding nature of commitments, can be constructed using this advantage: given candidate commitment C_s and collection of plaintexts P find the plaintext committed to by C_s as follows: $S = \{p\ s.t.\ sameSecret(C_s,\ C_p)\ where\ C_p = Commit(p\ , r\ )\ \forall\ p\ \in\ P,\ r \underset{R}{\leftarrow} Z_q\}$. The attacker could thus find the plaintext(s) for which $sameSecret(c1, c2)$ held true. The attacker would thus have found the plaintext within the collection P with non-negligible advantage. This contradicts Proposition 2.1, Pedersen Commitments are perfectly hiding, thus implying Proposition 2.3: that Pedersen Commitments are Indistinguishable.

2.2.4.4 Secure Sketches and Fuzzy Extractors

Secure sketches (SS) and fuzzy extractors (FE) are cryptographic primitives which work with noisy secrets to allow the generation of helper data suitable for public storage, and the generation of a key usable for cryptographic purposes. The term "noisy secret" comes from the fact that the secret will generally, not be the same from presentation to presentation but rather may vary slightly within a prescribed noise threshold as is exactly the case with the biometric samples of a user. Error correcting codes are typically used within these primitives to provide the tolerance to intra-subject variation. Secure sketches and fuzzy extractors are well suited for bridging between the requirements of cryptographic applications, where repeatable keys are needed, and realities of biometric systems, where variances are present between biometric samplings for an individual.

Secure Sketch

The secure sketch is a cryptographic primitive with two methods, $P = SS::sketch(b)$ and $b = SS::rec(b', P)$. The $P = SS::sketch(b)$ method accepts a noisy secret (a biometric in our case) and generates helper data P which can

safely be public: the entropy of b remains high even when P is known. The rec function, $b = SS::rec(b', P)$ recovers the initial secret b given public data P and another secret b' which is within an acceptable distance threshold t of the original.

Fuzzy Extractors

The fuzzy extractor has two methods, $< P, R >= FE::gen(b)$ and $R = FE::rep(b', P)$. The $< P, R >= FE::gen(b)$ method accepts a noisy secret and generates a tuple containing helper data P and extracted randomness $\mathrm{R} \in \{0,1\}^l$ which is ε-near randomly distributed. The $R = FE::rep(b', P)$ recovers the random string R given public data P and another secret b' which is within an acceptable distance threshold t of the original.

We make the following assumptions for this thesis:

Assumption 2.5: Soundness of fuzzy extractor configuration

We assume soundness of configured parameters and the behavior of the fuzzy extractor construction for the chosen modality, yielding overwhelming probability of true accepts and negligible probability of false rejects.

Assumption 2.6: Recovery of R through Fuzzy Extractor implies identity

Given a fuzzy extractor tuple <P,R> created with $\mathrm{FE} :: \mathrm{Gen}(\ldots)$ using biometric $\mathrm{b}(\mathrm{U_I})$, if another biometric $\mathrm{b}(\mathrm{U_j})$, combined with the same public data P can regenerate the original R through $\mathrm{FE} :: \mathrm{Rep}(\ldots)$, then the two biometrics must belong to the same individual, and vice versa: $\forall\, \mathrm{U_I}, \mathrm{U_j} \in \mathcal{U}$, given $\langle P, R \rangle = FE :: Gen\big(b(U_i)\big)$ and $R' = FE :: Rep\big(b(U_j), P\big)$ then $R == R' \Leftrightarrow b(U_i) = b(U_j) \Leftrightarrow id(U_i) = id(U_j)$

The following propositions are from[DRS04]:

Proposition 2.4: Fuzzy Extractor correctness

The *correctness* property of fuzzy extractors guarantees that if $dis(b, b') \leq$ t and R, P were generated by $\langle R, P \rangle \leftarrow Gen(b)$, then $Rep(b', P) = R$.

Proposition 2.5: Fuzzy Extractor Security

The security property guarantees that for any distribution B on metric space M with min-entropy m, the string R is ε-near to uniform even to those who observe P.

Proposition 2.6: Fuzzy Extractor generation of identical R is negligible

Since R is ε-near to random distribution, if ε is sufficiently small, the probability that $FE: gen(\dots)$ generates the same R on successive calls is negligible. This is true whether the successive calls are made by the same user, or by different users.

Proposition 2.7: The probability of U_j recovering the key of U_i is negligible

Given two users U_i and U_j the probability that the key generated for one user is recovered by the other user is negligible: given $b_i = b(U_i)$, $b_j = b(U_j)$ and $dis(b_i, b_j) > t$ then if $<R_{U_i}, P_{U_i}> = \text{FE:Gen}(b(U_i))$, $<R_{U_j}, P_{U_j}> = \text{FE:Gen}(b(U_j))$ and key recovery occurs $R' = \text{FE:Rec}(b(U_j), P_{U_j})$, then with overwhelming probability $R' \neq R_{U_i}$.

Constructions of Secure Sketch and Fuzzy Extractors

Various constructions of SS and FE exist. In [DRS04], Dodis et. al. demonstrate constructions for secure sketches and fuzzy extractors on metric spaces using hamming distance, set difference, and edit distance. We discuss two techniques here, the code-offset construction for secure sketches, and the construction of fuzzy extractors using secure sketches. These constructions are used for the comparative analysis of Chapter 5.

One common construction for the secure sketch is referred to as the code-offset construction. [DRS04] points out that this construction is equivalent to the fuzzy commitment construction of Juels and Wattenberg [JW99]

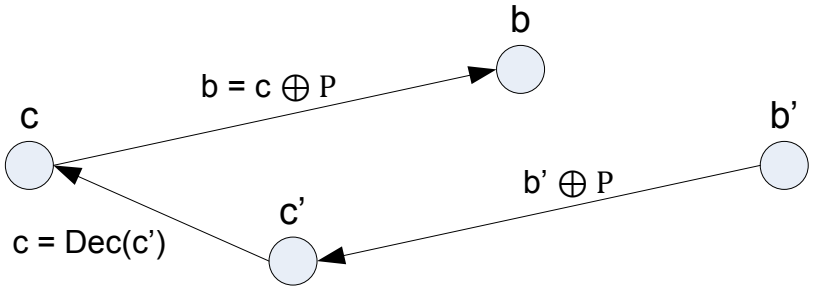

Figure 2. $SS::rec(b', P)$ for code-offset construction

The code-offset construction for secure sketches from [Dod04] provides a simple example, which is easily visualized in 2-dimensions. This construction uses an error correcting code C containing a set of codewords $K = \{c\}$ and an error correcting function $Dec(x)$ which maps an arbitrary point x to the closest codeword c in K. In $P = SS::gen(b)$, the secret b is encoded by selecting a random codeword c, calculating the offset distance P as the exclusive OR $\oplus$ between b and c, and returning this offset P. As illustrated in Figure 2, to recover the initial secret $SS::rec(b', P)$ accepts variation on the secret b', and the offset distance P calculated by $SS::gen(...)$ and proceeds to recover b from b' within acceptable distance threshold, by first performing $b' \oplus P$, to obtain c', then recovering c by using the error correcting function decode(c') to find the closest codeword, and finally, recovering the original secret b by shifting the recovered codeword c by the offset P, $b = c \oplus P$.

[DRS04] shows that a fuzzy extractor can be made using a secure sketch and a strong randomness extractor Ext(). Figure 3 presents the pseudo code.

$Gen(b)$:
    $set\ P\ =\ SS :: sketch(b), R\ =\ Ext(b)$
    $return < R, P >$

$Rep(b', P)$:
  $b\ =\ SS :: Rec(b', P)$
  $output\ R\ =\ Ext(b)$

Figure 3. Construction of fuzzy extractor using secure sketch

As shown in Figure 3, the $FE::Gen(b)$ accepts secret b and first uses it as argument to the $SS::sketch$ method to produce the public data P. The strong extractor is then used to extract randomness R. $FE::Gen(b)$ returns the tuple $< R, P >$. To reproduce the key R using noisy secret b', $FE::\mathrm{Rep}(b', P)$ first recovers the original secret b using the supplied arguments to invoke $SS::rec(b', P)$. Having successfully recovered the secret, the random key R is reproduced by invoking the strong extractor on the recovered secret $R\ =\ Ext(b)$. The recovered key is returned to the user. This will only work if b and b' are within distance the threshold t: if they are not, b will not be recovered, and the strong extractor will return a value not equal to R.

2.2.4.5 IND-CCA2 Secure Symmetric Key Block Encryption Scheme

Ciphertext Indistinguishability

"Indistinguishability" describes a property of a cryptosystem in which an attacker is unable to distinguish between ciphertexts given the messages which they encrypt.

Three incrementally strong variants on the property of indistinguishability are presented, which are usually described in terms of an attack game in which the adversary is given the opportunity to experiment with the encryption and decryption algorithms to learn information that could be useful at the challenge. At a particular point in the game, the challenge occurs. The adversary chooses 2 plaintexts, submits them to the challenger who selects one at random, and encrypts it. The ciphertext is

returned to the adversary, who must decide to which of the two plaintexts the ciphertext corresponds. The levels of security vary depending how and when the attacker can use the encryption and decryption services, as follows:

Indistinguishability under Chosen Plaintext Attacks (IND-CPA): In IND-CPA, the attacker may only experiment with the encryption algorithm, encrypting different plaintexts, with goal of learning enough information to have an advantage in guessing which of two plaintexts a given challenge ciphertext encrypts.

Indistinguishability under Chosen Ciphertext Attacks (IND-CCA): In IND-CCA the attacker may experiment with both encryption and decryption until the challenge, when ciphertext is received, after which the decryption service may no longer be used. Due to the fact that the decryption service may only be used before the challenge and guess, IND-CCA is also referred to as an a-priori chosen ciphertext attack.

Indistinguishability under Adaptive Chosen ciphertext Attacks (IND-CCA2): In IND-CCA2 the attacker is again given access to both encryption and decryption services. In contrast to IND-CCA, however, the attacker may continue to experiment with the both these services after the challenge ciphertext is received. The only restriction is that the decryption service may not be used to decrypt the challenge ciphertext. An IND-CCA2 attack is also referred to as a-posteriori chosen ciphertext attack.

IND-CCA2 Symmetric Key Block Cipher

The protocol in this thesis uses an IND-CCA2 encryption algorithm to protect the public data generated during the issue protocol and stored on the user's computer. While any IND-CCA2 encryption scheme may be used, our proposed protocol is presented with the Private Key Block Cipher secure against a-posteriori CCA ("IND-CCA2 Block Cipher"), as presented in [Gold02] "Construction 5.4.19", and as shown in Figure 4, below.

$\boldsymbol{E}\big(\boldsymbol{x}, \boldsymbol{k} = (\boldsymbol{k_1}.\boldsymbol{k_2})\big)$:

$r \underset{R}{\leftarrow} \{0,1\}^K$

$p \leftarrow f_{k_1}(r)$

$y \leftarrow p \oplus x$

$t \leftarrow F_{k_2}(r,\ y)$

$c \leftarrow \langle r\ ,\ y\rangle$

$return\ \langle \langle r\ ,\ y\rangle, t\rangle$

$\boldsymbol{D}\big(\boldsymbol{c} = \langle \boldsymbol{r}\ ,\ \boldsymbol{y}\rangle, \boldsymbol{t},\ \boldsymbol{k} = (\boldsymbol{k_1}.\boldsymbol{k_2})\big)$:

$if\ \big(f_{k_2}(r,\ y) == t\big)$

$p \leftarrow f_{k_1}(r)$

$return\ x \leftarrow p \oplus y$

$else$

$return\ \perp$

Figure 4. IND-CCA2 Block Cipher

This algorithm requires a compound key $k = (k_1, k_2)$, a random number selection process, and a pseudo-random function family from which the encryption pad PRFs and the MAC PRF, f_{k_1} and f_{k_2} respectively, can be obtained using the elements of the compound key.

Encryption returns a tuple $\langle c = \langle r\ ,\ y\rangle, t\rangle$ consisting of a ciphertext tuple $c = \langle r\ ,\ y\rangle$ and accompanying message authentication code (MAC) t. Ciphertext tuple c contains a random value r, and the ciphertext y which is the plaintext x XOR'ed with encryption pad p. The encryption pad p is obtained by passing random value r through f_{k_1}. The MAC t is the computation of f_{k_2} of the ciphertext tuple values.

The decryption algorithm receives a cipher text tuple, its corresponding MAC, and the symmetric key. Decryption first verifies that cipher text tuple and MAC are consistent with each other by retrieving f_{k_2}, deriving a new MAC from the ciphertext tuple, and comparing it to the MAC received as argument. If these are equal, the de-

cryption recreates the encryption pad p using r and f_{k_1} and returns the XOR of p and ciphertext y. If however, the correct relationship does not hold between ciphertext and MAC the algorithm returns null $\perp$.

As will be discussed in Section 3.2, in the proposed algorithm, $E(\ldots)$, $D(\ldots)$, and $k = (k_1.k_2)$ are configured onto the biometric devices.

Proposition 2.8: "IND-CCA2 Block Cipher" provides IND-CCA2 indistinguishability

The "IND-CCA2 Block Cipher" shown in Figure 4 provides indistinguishability of ciphertexts under adaptive ciphertext attacks. Proof available in [Gold04].

2.2.5 Cryptographic Protocols

2.2.5.1 Proofs of Knowledge

A *Proof of Knowledge* (PoK) is an interactive protocol in which a prover $\mathcal{P}$ convinces a verifier $\mathcal{V}$ of possession of particular knowledge, without divulging that knowledge. In general a proof of knowledge has the properties of *completeness* and *validity*. The property of *completeness* states that if $\mathcal{P}$ holds the required knowledge, then $\mathcal{P}$ will succeed in convincing $\mathcal{V}$ of that fact. The property of *validity* states that if $\mathcal{V}$ accepts the proof, then $\mathcal{P}$ really knows the required knowledge. An additional property can be added: *zero-knowledge,* which states that during the protocol, $\mathcal{V}$ learns nothing beyond the fact that $\mathcal{P}$ holds the required knowledge. A PoK holding these three properties is called a *Zero Knowledge Proof of Knowledge* (ZKPoK).

The *Σ-Protocol* is a three move interactive protocol form used to prove knowledge. The general Σ-Protocol protocol between $\mathcal{P}$ and $\mathcal{V}$ proceeds as follows:

1) $\mathcal{P}$ calculates a commitment, sends to V;
2) $\mathcal{V}$ generates random challenge, sends to $\mathcal{P}$,
3) $\mathcal{P}$ computes response value with information from commitment, and knowledge to be proved, sending this response to $\mathcal{V}$.

$\mathcal{V}$ accepts if verification relation holds.

The Σ-Protocol can be made zero-knowledge using Pedersen commitments. The Σ-Protocol can be made non-interactive using the Fiat-Shamir heuristic in which

a) the interaction with $\mathcal{V}$ in step 2) above may be removed, and replaced with a call to a cryptographic hash function [FS86].

The *Schnorr protocol*, a proof of knowledge for the discrete logarithm, provides an excellent example of a sigma protocol. Given group G_q of order q with generator g and $y = g^x$, where g and y are public values, $\mathcal{P}$ proves knowledge of *x* to $\mathcal{V}$ by engaging in the following protocol:

1) $\mathcal{P}$ selects random value r , sends commitment t = g^r to $\mathcal{V}$,
2) $\mathcal{V}$ then selects a random challenge value *c* and sends it to $\mathcal{P}$.
3) $\mathcal{P}$ creates response s using challenge *c* and the secret: $s = r + cx$

   $\mathcal{V}$ accepts if $g^s = ty^c$

This thesis uses the proof of knowledge in the following section to establish correspondence between issue-time and show-time information. As well, the protocol presented in this thesis offers improvements on [Ada11] by doing away with a series of Schnorr-like proofs to demonstrate that the issue and show time commitments are within required hamming distance of each other.

2.2.5.2 *Proof of Knowledge of DL-Rep with Pedersen Commitments*

Adams [Ada11] presents a zero knowledge proof of knowledge which, given public values h', Pedersen commitment C_{pub} and associated group parameters, provides the ability to prove knowledge of the discrete log representation of h' consisting of l+1 terms, where the l^{th} term C_{priv} is a Pedersen commitment on the same secret as C_{pub}. This protocol, which we refer to as $PoK: DLRepWithPC$ throughout this thesis, is presented in Figure 5.

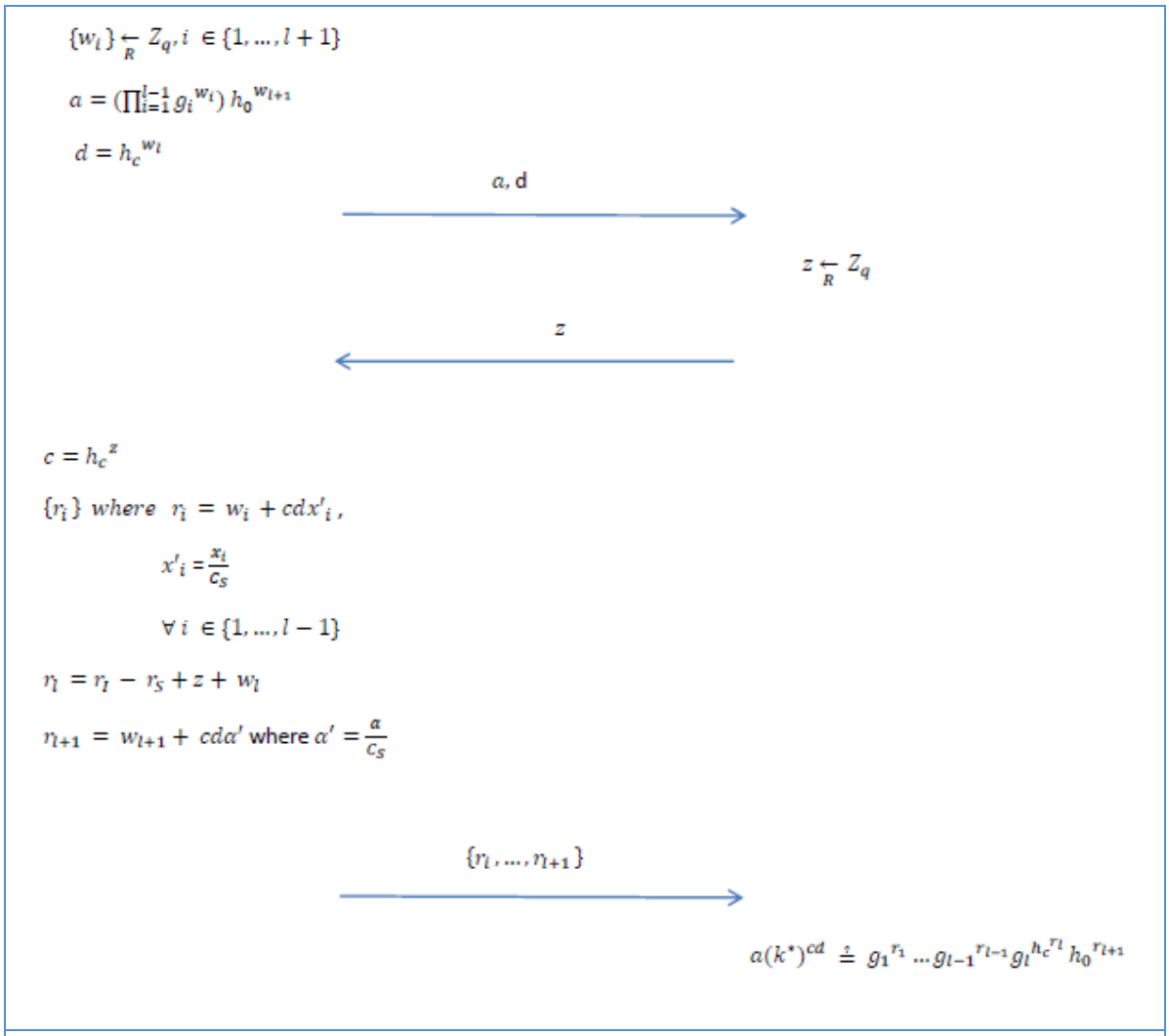

Figure 5. Proof of Knowledge $DLRepWithPC$

In the following, $\{g_1, \ldots, g_l\}$ are generators of G_q and $\{g_C, h_C\}$ are generators G_p; $X = \{x_1 \ldots x_{l+1}\}$, $C_{pub} = g_C^R h_C^{r_S}$ *and* $x_l = C_{priv} = g_C^R h_C^{r_I} \in G_q$, and $R, r_S, r_I \in G_q$.

The following information is made public:

$\{g_1, \ldots, g_l\}$ generators of G_q,

$\{g_C, h_C\}$ are generators G_p

The Prover retains the following private information:

DL Rep: $X = \{ x_1 \dots x_{l+1} \}$

Secret R and random values r_I and r_S

The protocol proceeds as follows:

1) $\mathcal{P}$ selects $t + 1$ random values $\{w_i\}$ calculates
   $a = (\prod_{i=1}^{l-1} g_i{}^{w_i})\, h_0{}^{w_{l+1}}$ and $d = h_c{}^{w_l}$ and sends these to $\mathcal{V}$.
2) The $\mathcal{V}$ then selects a random challenge $z \underset{R}{\leftarrow} Z_q$ and sends it to $\mathcal{P}$
   ($\mathcal{V}$ and $\mathcal{P}$ can now construct $= h_c{}^{z}$)
3) $\mathcal{P}$ constructs responses $R = \{r_i\}$ such that:

$$r_i = \begin{cases} w_i + cdx'_i & if\ i \in \{1, l-1\} \\ r_I - r_S + z + w_l & if\ i = l \\ w_i + cd\alpha'_i & if\ i = l+1 \end{cases}$$

$$where\ \ i \in [\,1\,, l + i\,]$$
$$and\ \ x'_i = \frac{x_i}{c_S} \quad \forall\, i \in \{1, \dots, l-1\}$$
$$and\ \ \alpha' = \frac{\alpha}{c_S}$$

$\mathcal{P}$ sends responses $\{r_1, \dots, r_{l+1}\}$ to $\mathcal{V}$.

4) $\mathcal{V}$ accepts if the verification relation
   $a(k)^{cd} \stackrel{?}{=} g_1{}^{r_1} \dots g_{l-1}{}^{r_{l-1}} g_l{}^{h_c{}^{r_l}}\, h_0{}^{r_{l+1}}$ holds.

While PoK: DLRepWithPC is of general applicability, it is presented in [Ada11] as part of a protocol for non-transferability of digital credentials. We retain the variable names and indexing used in the original presentation. We parameterize the PoK's invocation as $\text{DLRepWithPC}_{\mathcal{P},\mathcal{V}}(X, C_{priv}, R, r_I, r_S)$, where $\mathcal{P}, \mathcal{V}$ denote the Prover and Verifier respectively, and the parameters correspond to the private information required of the Prover.

Proposition 2.9: DLRepWithPC accepts if commitments are on the same secret

Assuming $\mathcal{P}$ knows the discrete log representation of h' consisting of the attributes $\{ x_0 \ \dots \ x_{l+1} \}$, random value r, and secret R the protocol DL-Rep with PC will pass if $x_l = C_{priv}$ is on the same secret R as C_{pub} and fail otherwise. While the detailed proof is available in [Ada11], as a sketch, we point out the following:

1) By applying the exponent $\frac{1}{C_{pub}}$ to h' one can derive $k' = (h')^{\frac{1}{C_{pub}}}$;
2) The l^{th} element in the DLREP of k' is $\frac{C_{pub}}{C_{priv}}$; and,
3) When C_{pub} and C_{priv} are on the same value, then $\frac{C_{pub}}{C_{priv}} = \frac{g_C{}^R h_C{}^{r_I}}{g_C{}^{R'} h_C{}^{r_S}} = \frac{\mathrm{h_C}^{\mathrm{r_I}}}{\mathrm{h_C}^{\mathrm{r_S}}} = \mathrm{h_C}^{\sigma}$ where $\sigma = \mathrm{r_I} - \mathrm{r_S}$.

2.2.5.3 Brands' Digital Credential Algorithms

The protocol proposed in this thesis is illustrated using Brands' digital credential scheme [Bra00]. This section discusses the required setup and the issue and show protocols of digital credentials.

Common public parameters

The public parameters to Brands' digital credential scheme are the parameters describing the group G_q:

1) q, prime order of G_q (Where G_q may be constructed as in Section 2.2.2.2.1, Prime Order Sub-Group);
2) g_0 a random generator of G_q.
3) Cryptographic hash function $H(\ldots)$ as discussed in Section 2.2.4.1

Issuer Key Generation

To generate the public and private keys, the Issuer *I* does the following:

1) Select x_0 at random from G_q
2) Select l random numbers from Z_q: $\{y_1 \ldots y_l\}$.
3) Collect the results of raising the public generator g_0 to each of these y_i, thus: $\{g_i\}\ where\ g_i = {g_0}^{y_i}\ \forall\ i\ \in [1, l]$
4) Generate $h_0 = {g_0}^{x_0}$
5) Set Issuer public key to: $\langle h_0,\ \{g_1 \ldots g_l\}\rangle$
6) Set Issuer private key to $\langle x_0, \{y_1 \ldots y_l\}\rangle$.

The Issue Protocol

The issue protocol proceeds between user *U* and issuer *I,* initiated by U. The issue protocol has access to public parameters g_0 and *H(…)* as well as issuer public key $\langle h_0,\ \{g_1 \ldots g_l\}\rangle$. The user begins the protocol with knowledge of l attributes $X = \{x_i\}$ where $x_i\ \in Z_q,\ i \in [1, l]$, and finishes the protocol also holding a credential h' and issuer (blinded) signature $({r_0}',\ {c_0}')$.

In [Bra00], Brands presents a number of variations on the issue protocols including constructions based on the DLREP and RSAREP functions as well as security enhancements to provide immunizations against parallel execution attacks. For the sake of simplicity, the proposed protocols in this thesis are demonstrated using the

"DLREP-based Scheme 1" variation shown in Figure 6. The other protocol variations, however, can similarly be used.

0) *U* initiates protocol:

i. *U* sends attributes $\boldsymbol{X} = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_l}\}$ to *I*

ii. *I* verifies attributes as needed through external process

1) *I* provides randomness:

i. *I* generates random number w_0,

ii. I sends $\boldsymbol{a_0} = \boldsymbol{g_0}^{\boldsymbol{w_0}}$ to U

2) U begins blinded signature process:

i. U selects random $\propto_1, \propto_2, \propto_3 \in Z_q$

ii. U calculates $h' = hg_0{}^{\propto_1}$, where $h = \prod_{i=1}^{l} g_i{}^{x_i}$

iii. U calculates $c_0' = H(h',\ g_0{}^{\propto_2}(h_0 h)^{\propto_3}\ a_0)$

iv. U calculates $c_0 = c_0' + \propto_3 \ mod\ q$

v. U sends $\boldsymbol{c_0}$ to I

3) I sends blinded signature to U:

i. I calculates $r_0 = c_0\big(x_0 + \sum_{i=1}^{l} x_i y_i\big) + w_0\ mod\ q$

ii. I sends $\boldsymbol{r_0}$ to U

4) U verifies and removes blind:

i. U accepts iff $g^0(h_0 h)^{-c_0} = a_0$

ii. U calculates $r_0' = r_0 + \propto_2 + c_0' \propto_1 \ mod\ q$

iii. *U* sets digital credential to $\boldsymbol{h'}$

iv. *U* sets issuer signature to $<\boldsymbol{c_0'}, \boldsymbol{r_0'}>$

Figure 6. Digital Credential Issue Protocol (DLRep-based Scheme 1)

Together, the credential and signature $<h', (r_0',\ c_0')>$, form user's credential public key and the values $X = \{x_1, \ldots, x_l\}$ and $\propto_1$ are the private key, where some of the values of X may be divulged, but $\propto_1$ must be kept secret. The issue protocol is constructed such that when it is complete a verification relation can be evaluated

which proves the integrity and authenticity of the credential signature pair. The verification relation differs depending on which variation of the issue protocol is used. For DLREP-based Scheme 1, the verification relation is $c_0' \stackrel{?}{=} H\left(h',\ g_0^{r_0'}(h_0\, h')^{-c_0'}\right)$.

The Show Protocol

The show protocol allows *U* to convince verifier *V* of possession of credential h' certified by *I* and of a credential statement *F* on these attributes. As described below, certain items within the show protocol depend on the specific F to be demonstrated. A sample instance of the show protocol (based on Figure 3.1 in [Bra00]) using on statement F = "x_1= y" is presented in Figure 7 below:

0) User initiates protocol
   i. I sends $\langle h', c_0', r_0' \rangle$ to V
   ii. V checks verification relation $c_0' = H\left(h',\ g_0^{r_0'}(h_0\, h')^{-c_0'}\right)$
1) Verifier sends formula description and nonce
   i. V calculates m = nonce‖…
   ii. F = "x_1 = y"
2) User calculates a, c and $\{r_i\}$
   i. U calculates a, where:

a = $g_2^{w_2}$ …$g_l^{w_l} h_0^{w_{l+1}}$, and,

$\{w_2, \ldots, w_{l+1}\} \underset{R}{\leftarrow} \mathbb{Z}_q$

   ii. U calculates response values $\{r_i\}$ i ∈ [2, l+1] where

$$r_i = \begin{cases} c\, x_i + w_i \bmod q & if\ i \in [2, l] \\ c \propto_1 + w_{l+1} \bmod q & if\ i = l+1 \end{cases}$$

   iii. User sends <a, c, $\{r_i\} > to\ V$
3) V verifies F:
   i. The verifier accepts if the formula verification relation is true:

$$h'^c a \stackrel{?}{=} g_1^{yc}\, g_2^{r_2} \ldots g_l^{r_l} \alpha_1^{r_{l+1}}$$

Figure 7. DLREP-based Scheme 1 show protocol for $x_1 = y$

Some notes on Figure 7:

1) **Specific inclusion of credential verification relation**

   The credential verification relation in Step 0-ii) may be checked at any time to verify authenticity and integrity of credential and signature. In the show protocols of [Bra00], the credential verification relation is omitted: it is implied, rather than explicitly shown, reducing required number of rounds in the protocol. We specifically include it here, as a preliminary step to highlight its presence as it is a fundamental part of the protocol which plays an important role in the security proofs.

2) **Credential Statement Grammar**

   The credential statement *F* is a boolean expression on the user's attributes which must conform to a particular grammar. Statement *F* may be a disjunction of sub-statements, in which each sub-statement is a conjunction of linear expressions on attributes in X which contains at most one negation. Example 3.5.5 from [Bra00] provides a sample a formula illustrating the allowed grammar for *F*:

   $((x_1 + 2x_2 + 5x_3 = 13)$ AND $(x_1 - 4x_3 = 5))$ OR

   (NOT $(x_1 + 3x_2 + 5x_3 = 7)$ AND $(3x_1 + 10x_2 + 18x_3 = 23))$

3) **Protocol specifics which depend on credential statement *F***

   Figure 7 illustrates the show protocol for a particular F = "$x_1 = y$" for some $y \in Z_q$. The specific values used in the proof will vary based on the formula to be proved. This is true for the value used for the coefficients of the exponents in Step 2-i); for the set of response values in computed Step 2-ii); and, for the exponents used in the formula verification relation of Step 3-i).

General Properties of Digital Credentials

We make the following assumption about the issue process:

Assumption 2.7: Verified Identity during Credential Issuance

We assume the individual's identity is reliably verified at the time of credential issue. This may include a check on the Civil Registry if needed. This assumption ensures that the credential when issued does not go to an imposter.

Some useful properties of the Digital Credential protocol from [Bra00] are provided below:

Proposition 2.10: Signatures on credentials are unforgeable

Under appropriate construction from [Bra00] the issued signatures $<c_0', r_0'>$ are provably unforgeable. (See [Bra00], Proposition 2.5.5, 4.3.2, 4.3.10)

Proposition 2.11: Signatures on credentials provide tamper resistance to X and h'

Under appropriate construction from [Bra00] the issued signatures $<c_0', r_0'>$ and appropriate verification relation prevent any changes to the values $\{x_1, \ldots, x_l\}$. (See [Bra00], Proposition 3.3.8, 4.3.2, 4.3.10)

Proposition 2.12: Blinded signature prevents Issuer from knowing final $<h', c_0', r_0'>$

Going through the protocol, the Issuer has no way of deriving the final credential and signature $<h', c_0', r_0'>$ based on the Issuer view of the protocol. (See [Bra00], Proposition 4.3.4)

Proposition 2.13: Verifier and Issuer cannot collude to link transactions

Given blinding process and random selection of $\propto_1, \propto_2, \propto_3$, it is not possible for issuer and verifier to collude to link c_0 from the issue protocol, with $<h', c_0', r_0'>$ from the show protocol.

Proposition 2.14: An attacker cannot derive $\propto_1$ even knowing X

While the attributes *X* are not necessarily private, the random value $\propto_1$ must be kept private. Even if an attacker knows all *X*, the attacker still can-

not derive $\propto_1$. This is due to the hardness assumption of the discrete log problem. (See [Bra00], Proposition 4.3.14)

2.2.5.4 Adams' Protocol for Non Transferability

Adams presents an extension to digital credentials to provide non-transferability [Ada11]. The protocol adds global parameters g_C and h_C, generators of Z_q and biometric devices to the issue and show protocols. Devices accept a biometric sample b and random value $r \underset{R}{\leftarrow} Z_q$ from the user, and produce Pedersen commitment $C = {g_C}^b \, {h_C}^r$ and returning <*b, C*> to the user, and C to the service provider.

During the Issue protocol, the user interacts with the biometric device, submitting issue-time biometric sample b_I, and random value r_I. The device creates commitment C_I, which is placed into the user's l^{th} attribute. After which, the underlying issue protocol [Bra00] is used to create the credential and obtain the issuer's signature.

During the show protocol, once again the user interacts with the device, submitting a biometric sample b_s and fresh randomness $r_s \underset{R}{\leftarrow} Z_q$ and obtaining show-time commitment C_s which is also given to the verifier. The user creates a third commitment C_f using the issue-time biometric and fresh randomness. The user proves to the verifier that C_s and C_f are on the same biometric using the proof of knowledge described in Section 2.2.5.2. After this, the user and verifier engage in up to $|b_I| - t$ Schnorr-like proofs of knowledge using commitments on the bits of b_I to verify that the biometric underlying C_s and b_I are within hamming distance threshold t of each other.

2.3. Credential Systems and Biometric Privacy Schemes

This thesis draws on 2 main areas: cryptographic credentials, and fuzzy extractors. This section presents a literature review of both these areas.

2.3.1 Cryptographic Credential Systems

This section provides a high-level overview of the main alternative cryptographic credential schemes that have been presented over the years, as well as the variations of features and functionality that have been elaborated in the literature. Throughout the discussion, we will highlight aspects relating to non-transferability as relevant.

In 1985, Chaum presents a landmark paper which identifies the privacy concerns resulting from the ability of service providers to aggregate the electronic records of individuals into dossiers which would form a profile on the individual [Cha85]. The paper presents a credential system based on the discrete logarithm problem, and blind signatures. In the algorithm, the user agrees on a pseudonym with a trusted party and interacts with organizations using a particular pseudonym. The pseudonyms are unlinkable between organizations, and use blind signatures to enable the showing of a credential issued by one organization to another. Chaum presents an intuitive analogy to a paper based systems including carbon paper, and an envelope with an acetate window. To receive a credential from an organization, the slip of paper is placed in a carbon-lined envelope making her pseudonym with the issuing organization visible through the envelope's acetate window. The issuing organization grants a credential by applying a repeating stamp to the outside of the envelope. The mark transfers to the paper inside, placing the credential beside the hidden pseudonyms. To show the credential to another organization, the individual moves the piece of paper within the envelope so that the appropriate pseudonym and the adjacent credential are visible.

In this scheme, Chaum uses a trusted organization to ensure that messages are properly constructed. While [Cha85] introduces the concepts and provides semi-formal sketches of the algorithms and proofs, the concepts are formalized, extended and refined in a follow-up paper the following year [CE86].

In the follow-up paper, Chaum and Evertse elaborate on the credential system introduced in [Cha85] generalizing and offering a multi-party, multi-credential system. The third party signing authority is preserved.

In this proposal, an individual has many pseudonyms, one per organization; a Signing Authority is used; and credentials each have a credential public key c and private key c^{-1}, such that $cc^{-1} \equiv 1 \, mod \, \Phi(N)$. [CE86] expands on the role of the Signing Authority adding pseudonym validators, and generalizes from [Cha85] to present a system where the user holds multiple credentials.

The main limitations of [CE86] are that it relies on trusted party Z in both pseudonym creation and the showing protocol; that validators are necessary; that the cut-and-choose protocol can be costly; and that an "idealized model" which assumes idealized RSA and hash function is used for proofs. Credentials are not bound to the individual and are transferable by lending the private key information.

Damgard [Dam88] proves the feasibility of constructing a credential system using weaker assumptions than [CE86]. The security of the proposed system rests on the problem of factoring Blum integers, which enables the claw-free functions. A Blum integer is an integer $n = pq$ where $n \equiv 3 \, mod \, 4$ and p and q are prime. A claw free permutation is a pair of permutations f_0 and f_1 such that it is hard to find a collision$(\mathrm{a}, \mathrm{b}, \mathrm{c})$ such that $\mathrm{f}_0(\mathrm{a}) = \mathrm{f}_1(\mathrm{b}) = \mathrm{c}$. In [Dam88] the trusted authority is only required at pseudonym registration time, but need not be online when issuing and verification transactions occur. [Dam88] moves the responsibility for credential signing to the issuer, whereas in [CE86] this responsibility lay with the Signing Authority. The author acknowledges that the proposed system is of predominantly theoretical interest with a goal of proof of feasibility rather than efficiency. As well, credential sharing is possible through a copy of the credential.

In 1995, Chen [Che95] proposes a credential system based on subgroup G_q, rather than Z_n^* on which [Cha85], [CE86] were based. [Che95] proposes a credential system in which the trusted center is not required during credential issuance, but only required in the initial assignment and validation of pseudonyms. This reduced involvement of the trusted authority in and of itself presents an important improvement. Furthermore, the trusted center in [Che95] does not require cut-and-choose to validate pseudonyms as did [CE86].The proposed system has the further feature that pseudonyms can be used as signing keys. As we will see later, the use of pseudonym as signing/encryption keys is a feature that Lysyanskaya et. al. [LRSW99] would later

identify as a feature which is nice-to-have, but not essential. Under Chen's approach, transferability remains possible by the copying of private information. Furthermore, a malicious trusted center can transfer credentials between users.

The period from 1996-1999 produced foundational work which would contribute to coming works in the period from 1999-2003. Some of this work includes Chaum and Pedersen's Wallet with Observer architecture [CP92], Jan Camenisch's work in Zero Knowledge Proofs, Anna Lysyanskaya's work in signatures and Stefan Brands' work in anonymous payments. The primitives and concepts introduced in these papers would provide input to the credential systems to follow.

The period from 1999 to 2001 saw important activity in the area of credential systems, with complementing and contrasting work occurring. Notably the non-transferable anonymous credential scheme put forward by Canetti et. al.; Camenisch and Lysyanskaya's Anonymous Credentials, Stefan Brands' Digital Credentials, and Eric Verheuil's Chameleon Certificates.

In 1998-1999, Ran Canetti et. al. propose a non-transferable anonymous credential system which introduces the concept of "controlled anonymity" and prevents transfer of credential through an "all-or-nothing" approach.. In Canetti's approach, the user is issued a master key, which the user would never voluntarily give away (the examples include the key being a token permitting access to safety deposit box, house key, etc.). The key is then used in a manner whereby if a user were to transfer a credential, the key itself would also be transferred. The approach to non-transferability is a disincentive-based approach. The approach outlined by Canetti was patented in 2007 under US Patent number 7222362 [Can07].

In 1999 Lysyanskaya et al. [LRSW99] present an anonymous credential system which discourages sharing in a manner similar to Canetti's - by linking credentials to a user's master secret such that if the credential is shared, the secret is revealed. First [LRSW99] specifies an abstract definition of 4 algorithms upon which a credential system can be built, namely: a) generation of master secret key pair, and credential key pairs; b) establishment of pseudonyms between user CA, and issuing/verifying organizations; c) credential assignment (issuing); and d) credential transfer ("showing"). Furthermore, [LRSW99] puts forward a set of requirements

against which any candidate system should be assessed: (a) each authenticated pseudonym belongs to a unique user; b) the user's master secret cannot be deduced from her public key; c) credential sharing implies master secret sharing; d) unlinkability of pseudonyms; e) unforgeability of credentials; and f) possibility to use pseudonym ("nym") to sign and encrypt (nice to have). Following this, 2 constructions are proposed which adhere to the protocol and meet the requirements: a theoretical construction based on [Dam88] which illustrates construction of a credential system on any one way function; and a practical construction based on [Che95]. The practical scheme uses proofs of knowledge of equality of discrete logs, a hash function based transform to convert the interactive proof to a non-interactive proof, and a blinding procedure for the non-interactive proof to produce a transcript T which is independent of the prover's view of the conversation.

The user's master secret is selected to be a valuable piece of information, external to the credential system. It is also the discrete log of his master secret. Pseudonym issuance creates a pseudonym tuple based on the user's master secret, and uses a Proof of Knowledge to ensure the user knows his master secret. Authentication of pseudonym also uses a Proof of Knowledge to verify knowledge of the master secret. To issue a credential the issuer raises to user's public key to powers of its own credential secret key and then produces two blinded transcripts, proving the relationship between the issued credential and the credential secret key. These transcripts become part of the credential. When shown the credential, a verifying organization will check correctness of the transcripts to ensure credential is from the issuer and then engages in a zero knowledge proof of knowledge to ensure appropriate relationship between nym and master secret.

The main limitations of [LRSW99] are that the scheme is not practical, being based on one-way functions, and general zero-knowledge proofs. In terms of sharing, [LRSW99] discourages sharing, by linking sharing the credential to sharing a valuable external secret, such as a bank account or credit card number.

Stephan Brands **[Bra00]** presents a system in which credential issuer participates during the issue protocol to sign the digital certificate, and optionally as a part of showing to enforce usage constraints on certificates. The scheme, dubbed "Digital

Credentials" provides the ability to prove arbitrary boolean statements on the attributes contained in the certificate using proofs of knowledge. [Bra00] uses a signature created on a proof of knowledge based on the Fiat-Shamir heuristic [FS86] to sign the verifier's nonce, which provides protection against replay attacks.

The main limitation of [Bra00], is that the credentials are single show: the digital credential contains a digital credential public key and the Issuer signature which are presented during the showing transaction and can be used to link transactions. Thus, digital credentials are single show credentials. As presented in [Bra00], Brands' scheme is not a pseudonymous credential system as outlined in [Cha85]. No pseudonym is issued and the notion of transferring a credential from one pseudonym to another is not present. The concept of aggregating credentials from different issuers for asynchronous presentation individually or in combination to various organizations is not the focus. Digital credentials are akin to digital currency in which multiple attributes can be associated, and selectively proved. Transfer of credential is possible through copying. Brands provides the approach of including the biometric in the credential and also making the key a valuable piece of information.

In 2001, Camenisch and Lysyanskaya [CL01] present "Anonymous Credentials", which uses zero-knowledge proofs to deliver multi-show credentials. As per the model presented in [Cha85] the individual is associated with multiple pseudonyms, each of which represents a relationship with an organization. The individual can collect credentials issued by organizations, and prove they hold these credentials to other organizations. The system uses RSA groups and zero knowledge proofs of knowledge discrete logarithms, of discrete representations, of equality of discrete logarithms and representations on different bases, and of a discrete log within a particular range.

As additional features, Camenisch and Lysyanskaya also propose all-or-nothing non transferability which uses a publically accessible billboard and circular encryption. All-or-nothing non transferability has the property that if a user shares one credential, this allows all his pseudonyms and credentials to be used. While providing a disincentive to sharing, this does not prevent sharing among colluding users between which secrets are immaterial. They also provide for one-show

credentials with an offline double spending test. While the authors propose mechanisms to discourage sharing, nothing prevents Anonymous Credentials to be transferred between colluding malicious users.

In 2001, Verheuil [Ver01] presents an approach which complements Brands' digital credentials, allowing them to be multiple-show. The scheme introduces the concept of chameleon certificates, where the issuing organization provides a "*master chameleon certificate*" to a user, who uses a "*refresh*" algorithm to create a "*slave chameleon certificate*" for showing when required. The scheme works on the RSA Assumption, and does not include protection against lending beyond that proposed by Brands [Bra00].

In 2005, Yang, Bao and Deng follow up on Verheuil's earlier chameleon certificate approach, demonstrate an attack in which a colluding CA and SP are able to trace a user's transactions.

2.3.2 Survey of Privacy Enhancing Technologies for Biometrics

Privacy concerns accompany the use of biometrics due to the extent to which they identify the individual in an irrevocable manner. The algorithm presented in this thesis draws on work conducted in privacy enhancing technologies and applications to Biometric data. Specifically we draw on particular architectures, threat models and on the fuzzy extractor primitive.

Davida et. al. propose an architecture for an offline database and a card-based biometric matching system which used encryption and matching to protect biometric templates [DFM98]. In Davida's approach, as in the approach outlined in this thesis, the sensors hold private key data and thus are a point of maintenance for system update. Davida's approach is listed under Patent US7711152.

In 1999 Soutar et. al. [Sou99] propose an image-based approach using a majority vote based algorithm in which a number of images are captured, a majority vote sample is constructed at enrolment time and a Fourier transform calculated and a private key is generated. At verification time, samples of the verification biometric are taken, the Fourier transform is recalculated, and compared to the stored prototypes. The security of this technique was not proven. Due to the fact that match results were

leaked by the algorithm, the approach was vulnerable to hill-climbing attacks as shown by Adler in [Adl04] [Adl05].

In 1999, Juels and Wattenberg present a primitive called the fuzzy commitment, which offers the ability to create a commitment on a secret *w* which can be unlocked with a witness *w'* that is sufficiently close to the original secret relative to a vector distance metric [JW99]. This is in contrast to traditional cryptographic commitments which require an exact match on the witness to unlock. This is applicable to biometrics because for a given person there exists small variations between biometric samplings. In the fuzzy commitment scheme, noisy opening string *w* is used to form a commitment, where a codeword *c* is selected at random from codebook *C*, calculating the offset *d* from the secret to the codeword, and returning the commitment as the hashed random codeword and the offset $< h(c), d >$. The commitment is opened by applying error correction to a noisy secret and adding the offset to the corrected value to obtain c'. Authentication succeeds if the original hash value *h(c)* equals the hash of the corrected value *h(c').* The fuzzy commitment is based on vector difference which, in the case of a binary string, is the hamming distance. Alignment of biometric templates is necessary in this approach.

In 2002, Juels and Sudan introduce the fuzzy vault, a primitive akin to the fuzzy commitment, based on a set difference metric rather than a vector difference [JS02]. The fuzzy vault collects features from the secret, represents them as (x, y) values, and derives a polynomial which fits this original set. So called "chaff points", not lying on the polynomial, are then added to the set. The resulting set then contains a collection of valid points, and chaff points which hide the secret. At verification time, to extract the secret, a candidate biometric is sampled, and its characteristic points are created. The verification set of points is compared to the original set. If there is enough commonality between these, the original polynomial can be recreated, and the secret extracted. Polynomial reconstruction is impossible if the enrolment and verification biometric are so dissimilar that they do not have sufficient points in common. This approach based on set difference does not assume or require alignment of biometric templates.

Linnartz and Tuyls present a different approach to the problem of template protection and secret regeneration [LT03]. Given an assumed noise free biometric template X at enrollment-time, they encode a secret S to generate so called helper data W. The algorithm assumes that each dimension of the template is quantized at q resolution levels. Finding the helper data corresponds to determining whether a 1 or a 0 should be added to each quantized dimension. At verification time, a noisy version of the biometric Y is obtained and used in conjunction with W to obtain a message similar to S. Template alignment is assumed in this technique.

Dodis et. al [DRS04] generalize the fuzzy commitments and fuzzy vaults put forward in by Juels [JW99] [JS02], presenting two corresponding primitives, the Fuzzy Extractor (*FE*) and the Secure Sketch (*SS*). The Secure Sketch consists of a pair of functions $\langle sketch, Rec \rangle$. The $sketch(\dots)$ function accepts a noisy secret and generates public data which is safe for storage. The $Rec(\dots)$ function allows recovery of the initial secret given the public data and a candidate secret sufficiently close to the original secret. The Fuzzy Extractor also consists of a pair of algorithms $\langle Gen, Rep \rangle$. The $Gen(\dots)$ function accepts a secret and creates a tuple $< P, R >$, where R is a random string, suitable for cryptographic purposes, and P consists of public data safe for storage. The $Rep(\dots)$ function reproduces the random string R using the public data P and a string w' sufficiently close to the original w. Dodis et. al. demonstrate FE and SS on three distance metrics (hamming distance, set distance, and edit distance), and show that the fuzzy commitment [JW99] and the fuzzy vault [JS02] are instantiations of secure sketches on the Hamming distance the Set difference metrics respectively.

In 2004, Xavier Boyen [Boy04] points out vulnerabilities in SS and FE under scenarios in which multiple calls to the $\mathrm{SS::sketch}(\dots)$ function are allowed. Under these conditions, in certain cases, an attacker can collect multiple samples of public data produced under separate enrollments of the same individual and exploit leaked information to distinguish between the users who produced the public data. Boyen proposes "adaptive chosen perturbation attacks" under which an attacker can repeatedly query an oracle service and selectively modify his query string in the process. Two variations on this model are proposed: the "insider" attack, in which the attacker

is allowed oracle access to both the extraction and regeneration methods, and the "outsider" attack in which only access to the extraction method is allowed.

Hao, Anderson and Daugman [HAD06] published a privacy protecting approach for iris templates in which the iriscode is not stored, but rather an error correcting string from which the iriscode cannot be constructed. Hao et. al. present a two-tier construction using Hadamard codes and Reed-Solomon error correcting codes, achieving empirical results with a 99.5% success rate. The authors present a two-factor (iris and token) and three factor (iris, password and token) authentication approach.

Bringer et al. [Bri07] further studied the approach presented by Hao Anderson and Daugman, incorporating some variations including an iterative decoding algorithm. They present empirical results of 5.25% false recognition rate. The authors also present a theoretical maximum of 2.49% based on particular code characteristics.

Other applications of fuzzy extractors and fuzzy vaults have been applied to various biometrics modalities, including fingerprints [CKL03] [NNJ08].

Multimodal biometrics have also been tackled with fuzzy vaults. Nagar, Nandakumar and Jain [NNJ12] present an approach to combine fingerprint, iris and face templates for template protection in the context of person identification. Wu et. al. [Wu11] use the fuzzy vault for fusion of facial feature templates under which a key is split into overlapped sub-keys, and each sub-key is then used to generate helper data from two fuzzy vaults, one on a facial feature template produced using a multi-block binary pattern, and the other using a template produced using principal component analysis [Wu11].

In 2005 Dodis and Smith revisit the secure sketch and show that the code-offset construction can be made indistinguishable by randomizing the choice of error-correcting code rather than using a fixed code.

In 2007 Sheirer and Boult describe a number of attacks on biometric encryption and fuzzy vaults. They present the attack via record multiplicity (ARM), the surreptitious key-inversion (SKI) attack, and the blended substitution attacks [SB07]. Using fuzzy vaults to illustrate, the ARM attack consists of two fuzzy vaults encoding the same secret, which are used to match commonality and extract biometric and key.

In the SKI attack, the underlying secret is obtained through dishonest acquisition (rather than by compromising the vault) and is then used to filter chaff points from an existing fuzzy vault, generate the polynomial and extracting the underlying biometric. Under the blended substitution attack the attacker overlays some of the chaff points in an honest user's fuzzy vault with the encoding of his own biometric. The vault's integrity is thus compromised, containing the data belonging to the honest user and to the attacker. The paper also illustrates attacks against the Biometric Encryption approach of Soutar et. al, recalling the attack by Adler and presenting their own ARM and SKI attacks. Schreirer and Boult present general requirements for biometric encryption and privacy enhancing technologies which include guards against the types of attacks put forward in their paper.

In 2009, Simoens Tuyls and Preneel [SKTP09] further examine attacks on Secure Sketches in multiple use scenarios introducing security definitions for indistinguishability and reversibility. Simoens presents attackers which are weaker than those of Boyen's attackers and demonstrates the vulnerabilities of certain SS/FE constructions in multiple use situations. Simoens et. al. put forward 3 attack games, the indistinguishability game, the N-indistinguishability game and the irreversibility game. Simoens also presents an attack on 2 constructions of secure sketch: the code offset and bit permutation. The attack against the code offset construction (which takes advantage of linearity) provides a significant advantage to an attacker targeting indistinguishability. Simoens illustrates sample calculations in which an attacker on a code of length 100 with a distance of 7 would win the indistinguishability game 4 out of 4 times.

2.4. Related Non-Transferability Approaches

As an introduction to non-transferability to cryptographic credentials we categorize and discuss the approaches here.

2.4.1 Approaches based on a Third Party

In these approaches, a third party is used to provide functionality to ensure appropriate user behavior. The PKI-Assured non-transferability of Camenisch and Lysyanskaya [CL01] is an example of such an approach. The downfall of such an approach is that the third party becomes quite powerful and a corrupt third party can cause damage to privacy and can also transfer credentials in some cases (cf. [CRRS99]).

2.4.2 Approaches based on Disincentives

Disincentive-based approaches discourage sharing rather than preventing it. This is typically done by embedding a valuable piece of information within the credential. In [CL01], for example, the idea proposed is to base the private key on a secret having intrinsic value outside the scheme, such as a bank account number, for example. The premise is that the data is so valuable that the credential holder is reluctant to share it. Here, transfer can be achieved if the user has no qualms about sharing the secret: this may be the case in close circles such as family, friends, or particular organizations.

2.4.3 Biometric based approaches

1) Directly Embedded Biometric Data

In Brands' Digital Credential scheme one manner in which non-transferability is addressed is by embedding biometric data. While this approach functions to deliver non-transferability, since the biometric is stored in the credential, there is some threat to biometric security: in Brands, showing the wrong attribute exposes the biometric.

2) Tamper-proof Chip

The Tamper-proof chip approach, also referred to as the Wallet-with-Observer architecture, was proposed by Chaum and Pedersen [CP92]. This architecture features two main components, a computer controlled by the user, and a tamper resistant chip, which is under the organization's control. These components collaborate to meet the system requirement of sending user data to a service providing organization in a manner which respects the user's privacy and meets the organization's need for data

correctness. In the protocol, these components provide a mutual check-and-balance mechanism where each ensures that the interests of their respective owners are maintained.

Under this model, the Observer cannot communicate with the outside world: all communications must go through the computer, which has the ability to block any message. Under the possibly unrealistic assumption that the computer is under the user's control, the tamper proof chip cannot send out any data not approved by the user. The protocol ensures that an organization will not accept any message which is not signed by the tamper proof chip through a blind signature scheme. This ensures that the organization's need for correct information is met.

The model proposed by Chaum and Pedersen presents the requirement that in-bound and out-bound messages should not leak any extra information which could be used by the organization or the tamper resistant chip to compromise the user's privacy. This extra information is called subliminal information.

Bleumer adds biometric authentication to the wallet-with-observer architecture[Bleu98]. In this proposal, the user device is equipped with a biometric reader and a tamper resistant chip. At enrolment time, when the tamper resistant chip is issued by the organization to a user, it initialized with the biometric information of the authorized user. In subsequent transactions, a fresh biometric is taken and compared to the enrolment biometric in a manner which ensures privacy based on the wallet-with-observer paradigm. Given that the chip is initialized with the user's biometric, that it is tamper proof, and that acceptable false acceptance rates exist on the biometric, the chip becomes bound to the individual and credentials become non-transferable. Bleumer uses restrictive blind signatures, restrictive cascade signatures and divertible proofs to implement the biometrically enhanced wallet-with-observer.

Impagliazzo [IM03] builds on the wallet-with-observer with biometric authentication model of Bleumer [Bleu98] and the definition of subliminal-free protocol of Burmeister [Bur99], proposing a stronger notion of a subliminal free protocol which includes specific requirements when dishonest parties are detected.

The weaknesses of tamper-proof chip based methods include that they rest on the assumption of tamper-resistance, which can be hard to guarantee in practice. As well, the approach imposes a cost on the issuing organization proportional to the number of users in the system: a chip must be created for each issue protocol. Depending on the model as well, an environment with multiple issuers may impose a significant inconvenience to the user where various chips from different credential issuers and the credentials they contain must be maintained.

3) Privacy Preserving Biometric Approaches

Adams technique for non-transferability [Ada11] does not directly embedded the biometric in the credential, rather, a commitment on the credential is embedded and then proofs of knowledge are used to verify the user at show-time. The protocol does, however, call for the storage of the biometric on the user's computer, compromise of which exposes the biometric.

In 2009, Blanton and Hudelson [BH09] propose an extension to anonymous credentials [CL01], [CL02] which uses fuzzy extractors, verifiable random functions and zero knowledge proofs to provide non transferability enforced using biometrically derived data.

The approach presented in this thesis can be categorized as this type. Similarly to Blanton and Hudelson, we use fuzzy extractors to derive keys. The technique of Blanton and Hudelson applies to anonymous credentials and is not verified on digital credentials. Our approach is demonstrated on digital credentials and is believed to apply to anonymous credentials. Our approach is also arguably more simple, requiring only fuzzy extractors and IND-CCA encryption complemented by a specialized biometric device and a Zero-Knowledge Proof of Knowledge.

2.5. Chapter Summary

This chapter has provided an overview of the required background for this thesis, presenting foundational material in biometrics, mathematics, number theory, information theory, and cryptography and presenting a literature review focusing on influential work that has been done in cryptographic credentials and biometric privacy enhancing technologies with special focus on fuzzy extractors and secure sketches.

The next chapter presents our protocol extension for non-transferability of cryptographic credentials

Chapter 3. Non-Transferability extension for Cryptographic Credentials

The following chapter presents our protocol extension for non-transferability of cryptographic credentials. In our technique, the issue and show protocol of the underlying digital credential protocol are supplemented with biometric devices that are configured with a new cryptographic primitive we propose, the fuzzy extractor indistinguishability adapter. The biometric devices generate a biometrically derived key which is sealed in a Pedersen Commitment, bound to the digital credential at issue time and verified through regeneration during the showing protocol.

This chapter presents the components of our proposal: first the indistinguishability adapter, then the biometric device, then the proposed non-transferability protocol in terms of required setup, issue and show of digital credentials, finally a sketch of the integration of the proposed protocol into the anonymous credentials scheme of Camenisch and Lysyanskaya is presented.

3.1. Fuzzy Extractor Indistinguishability Adaptor

The fuzzy extractor indistinguishability adaptor $\mathrm{FE}_{\mathrm{Ind}}$ is designed as a wrapper that can be used with any existing traditional fuzzy extractor constructions $\mathrm{FE}_{\mathrm{Trad}}$, providing resistance against multiple-use attacks [Boy04] [STP09] [BA11] (see Section 2.3.2). We use IND-CCA2 secure encryption (see Section 2.2.4.5) to make the public data indistinguishable, thus resistant to these multiple-use attacks. Figure 8 presents the Indistinguishability Adapter gen and rep methods.

$$FE_{Ind} :: gen\big(b_I, k = (k_1.k_2)\big):$$
$$< P, R_I > = \; FE_{Trad} :: gen(b_I)$$
$$\uparrow \langle P_e, R_I \rangle = \; \langle E(P,k), R_I \rangle$$

$$FE_{Ind} :: rep\big(b_S,\; P_e,\; k = (k_1.k_2)\big):$$

$\uparrow R_S = FE_{Trad} :: rep(b_S, D(P_e, k))$.

Figure 8. Fuzzy Extractor Indistinguishability Adapter

As presented in Figure 8 the indistinguishability adapter has $\langle gen, rep \rangle$ methods as do traditional fuzzy extractors. The indistinguishability adapter, however, requires an additional parameter: the symmetric encryption key k. The key which is used depends on which IND-CCA2 cryptosystem is used. In our case, we use the IND-CCA2 Secure Symmetric Key Block Encryption Scheme, discussed in Section 2.2.4.5; we therefore use compound key $k = (k_1, k_2)$.

For generation of private data and key, the indistinguishability adaptor's gen method $FE_{Ind} :: gen(b_I, k)$ receives biometric b_I and symmetric key k, invokes $FE_{Trad} :: gen(b)$ to obtain the tuple $\langle P, R \rangle$. The public data P is then encrypted using the symmetric key , $P_e = E(P, k)$; the resulting P_e is combined with the biometrically derived cryptographic key R and returned as tuple $\langle P_e, R \rangle$.

To reproduce cryptographic key R, the indistinguishability adapter's $rep(\dots)$ method $FE_{Ind} :: rep(b_S, P_e, k)$ receives the challenge biometric b_S, the encrypted public data P_e, and the symmetric key k. The public data is decrypted, $P = D(P_e, k)$ and the decrypted public data P is then used with the challenge biometric b_S to call the traditional fuzzy extractor's $rep(\dots)$ method to recreate the cryptographic key $R = FE_{Trad}::\text{rep}(b_S, P)$.

The cryptographic key R is reproduced if a) P was as created by $gen(\dots)$, b) the correct key k is used, and c) $dis(b_I, b_S) \leq t$)

3.1.1.1 Contrast with [BA11]

The Indistinguishability adapter we present is related to a construct presented by Blanton and Aliasgari in 2011 [BA11]. Though some similarities are present, our construct has some important differences which will now be discussed.

Both constructs can be instantiated on any existing Fuzzy Extractor construction to add the characteristic of indistinguishability to the return tuple. This pluggable approach allows the application designer to select the underlying Fuzzy Extractor construction based on the biometric modality and error tolerance requirements of the

particular deployment. Furthermore, both constructions use PRF's and symmetric key semantics as a means to provide indistinguishability between data generated by repeated calls to $FE::Gen(\dots)$.

To deliver indistinguishability of elements within the fuzzy extractor return tuple, [BA11] passes both P, and R through a transform of their own design. In contrast to [BA11], our focus is strictly on P, on which we apply IND-CCA2 encryption to produce P_e which is returned to the user. We do not provide added encryption on R, leaving to the ε-closeness property delivered by the underlying Fuzzy Extractor which delivers a measured degree of closeness to the uniform random distribution. IND-CCA2 encryption can be added for *R* depending on the needs of the application at hand. We focus on *P* because this is the element which has been identified in the literature as susceptible to multiplicity attacks [Boy04][SKTP09]

Our construct is based on Goldreich's symmetric key encryption scheme [Gol04]. In doing so, we benefit from the security proof and review which has occurred on that work. In contrast, [BA11] present their own primitive whose security proof was omitted in the published paper due to space limitations.

The primitive in [BA11] attempts to optimize storage cost. Pairwise independent hash functions are used to compress the biometric prior to its use as input to the pseudo-random function on the basis of space optimization. While the claim is made that entropy is not affected, this is not proved in the paper.

In contrast, we do not include such space optimization in order to present the construct in its simplest form. If needed, compression (using hashing or potentially other means) may be added to our approach.

The primitive offered in [BA11] is constructed on the secure sketch of the underlying system. Their fuzzy extractor construct is then composed in terms of the secure sketch. As a result of this approach, an $\mathrm{FE}_{\mathrm{Ind}}$ can be constructed purely based on a $\mathrm{SS}_{\mathrm{Trad}}$; however, the underlying system's fuzzy extractor $\mathrm{FE}_{\mathrm{Trad}}$ is not used. Due to this approach, [BA11], must devise and deliver a mechanism to generate and process P. For this they use a second PRF. Instead, we have chosen to implement $\mathrm{FE}_{\mathrm{Ind}}$ using the fuzzy extractor of the underlying system which has immediate benefits of simplicity and economy. It may also have a security benefit because the level

of randomness on R is a responsibility and quality of the underlying FE, which may have proven security qualities.

3.2. Device "Gen" and "Rep" Algorithms

We add a biometric device to the Issue and Show protocols of the underlying credential system. The devices are responsible for gathering the biometric sample from the user, performing biometric key generation and key reproduction, and adding required indistinguishability to the public data. Devices behave differently during the Issue and Show protocols: during Issue, the device D_I is configured to use the fuzzy extractor generate (GEN) method (Figure 9); during Show, the device D_S is configured to use the fuzzy extractor "reproduce" (REP) method (Figure 10). All devices are provisioned with symmetric encryption key k = (k_1, k_2). As will be seen below, the key is used by $D_I :: gen(b_I, r_I)$ and $D_S :: rep(b_S, P_e, r_S)$ in their invocations of the respective indistinguishability adapter methods.

3.2.1.1 Device GEN Algorithm

$$\boldsymbol{D_I :: gen(b_I, r_I):}$$

$$\langle P_e, R_I \rangle = FE_{Ind} :: gen(b_I, k)$$

$$CR_I = g_C{}^{R_I} h_C{}^{r_I}$$

$$return \langle P_e, R_I, CR_I \rangle \; to \; U$$

$$return \langle CR_I \rangle \; to \; I$$

Figure 9. Device GEN Algorithm

The $D_I :: gen(b_I, r_I)$ method accepts 2 arguments: the issue-time biometric b_I, and user-generated random value r_I. D_I first retrieves its symmetric key k and then invokes the indistinguishability adapter $FE_{Ind} :: gen(b_I, k)$ with the received biometric and the symmetric k to obtain the return tuple $\langle P_e, R_I \rangle$. D_I then creates Pedersen commitment $CR_I = g_C{}^{R_I} h_C{}^{r_I}$ using the biometrically derived key R_I and

the random value r_I. The tuple $\langle P_e, R_I, CR_I \rangle$ is returned to the user, and the commitment CR_I to the issuer.

3.2.1.2 Device REP Algorithm

$$\boldsymbol{D_S :: rep(b_S,\ P_e,\ r_S)}:$$

$$R_S = FE_{Ind} :: rep(b_S,\ P_e, k)$$

$$CR_S = {g_C}^{R_S} {h_C}^{r_S}$$

$$return\ \langle R_S,\ CR_S \rangle\ to\ U$$

$$return\ \langle\ CR_S \rangle\ to\ V$$

Figure 10. Device Rep Algorithm

The $D_S :: rep(b_S,\ P_e,\ r_S)$ method accepts the show-time biometric b_S, the encrypted fuzzy extractor public data P_e and user-generated random value r_S. D_S first retrieves symmetric key *k,* then invokes $FE_{Ind} :: rep(b_S,\ P_e, k)$ passing in the received biometric b_S, public data P_e and key k. $FE_{Ind} :: rep(b_S,\ P_e, k)$ returns a value R_S which may correspond to the original key R_I depending on whether a) P_e was as originally generated, b) the key k is correct and c) the issue and show time biometrics are within the required distance threshold.

D_S then creates Pedersen commitment $CR_S = {g_C}^{R_I} {h_C}^{r_I}$ using the biometrically derived key R_S and the random value r_S. D_S returns tuple $\langle R_S,\ CR_S \rangle$ to the user, and commitment $\mathrm{CR_S}$ to the verifier.

Table 3 summarizes the differences between device required inputs, behaviour and outputs in each of the issue and show protocols.

Protocol	Device method	Input from User	Invoked FE Behavior	Return data (User)	Return data (SP)
Issue	$D_I :: gen(b_I, r_I)$	$<b_I, r_I>$	$\mathrm{FE_{Ind}} :: \mathrm{gen(b_I, k)}$	$<P_e, \mathrm{R},\ CR_I>$	$<CR_I>$
Show	$D_S :: rep(\mathrm{b_S},\ P_e,\ r_S)$	$<b_S,\ P_e, r_S>$	$\mathrm{FE_{Ind}} :: \mathrm{rep(b_S}, P_e, k)$	$<R_S, CR_S>$	$<CR_S>$

Table 3. Biometric Device Arguments and Return Values

3.3. Complete view of device in Issue and Show modes

Figure 11 shows the device D_S configured in show-mode. The first thing to note is that the device stands between the user and the issuer. Internal to the device we see the fuzzy extractor indistinguishability adapter, which itself uses a traditional fuzzy extractor, the Pederson commitment algorithm, and INDCCA2 encryption. The user supplies biometric b_I and random r_I. The device retrieves onboard key k and uses it to invoke the $gen(...)$ method of the fuzzy extractor indistinguishability adapter FE_{Ind}, which uses FE_{Trad} to obtain $< R, P >$. P is encrypted using INDCCA2 encryption, commitment CR_I is formed on the key R. The tuple $<P_e, R_I, CR_I >$ is returned to the user, and $< CR_I >$ is returned to the issuer.

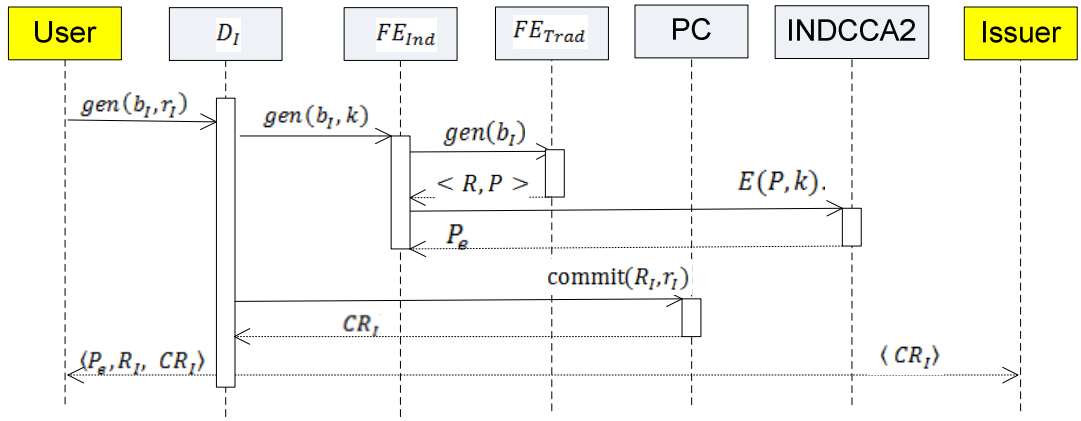

Figure 11. Device in Issue-Mode

Figure 12 shows the device D_S configured in show-mode. As in Figure 11, we see that the device stands between two entities, this time the user and the verifier. The device in show mode receives the show-time biometric b_S, the show time random r_S, and the encrypted public data P_e. As a first step FE_{Ind} is invoked with key k which is then used to decrypt public data P_e to retrieve P, which is provided with b_S as input to FE_{Trad} to recover the biometrically derived key R_S. Pederson com-

mitment CR_S is then formed on R_S and r_S and returned to each the user and the verifier.

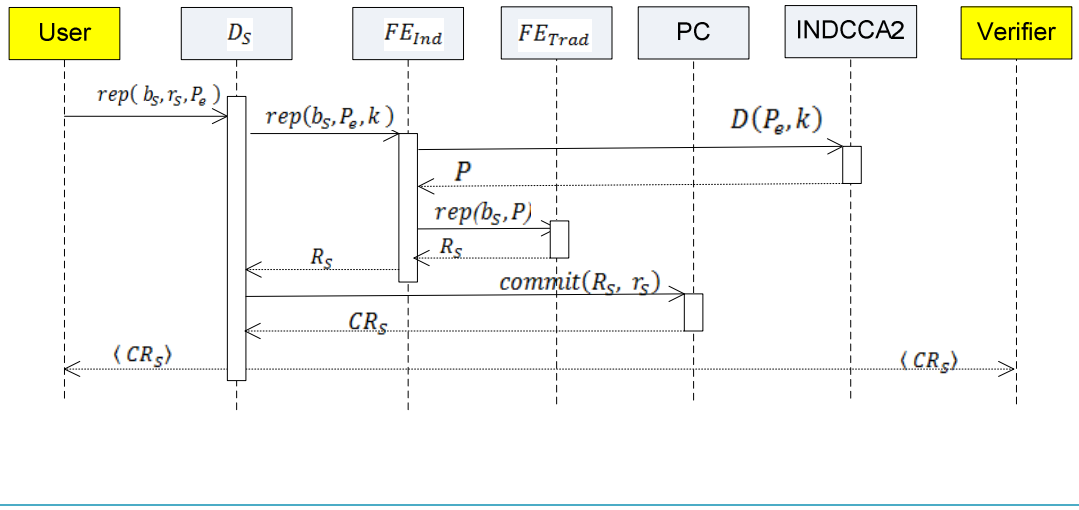

Figure 12. Device in Show-Mode

3.4. Application to Digital Credential Protocol

Figure 13 shows a high-level view of the proposed issue and show protocols.

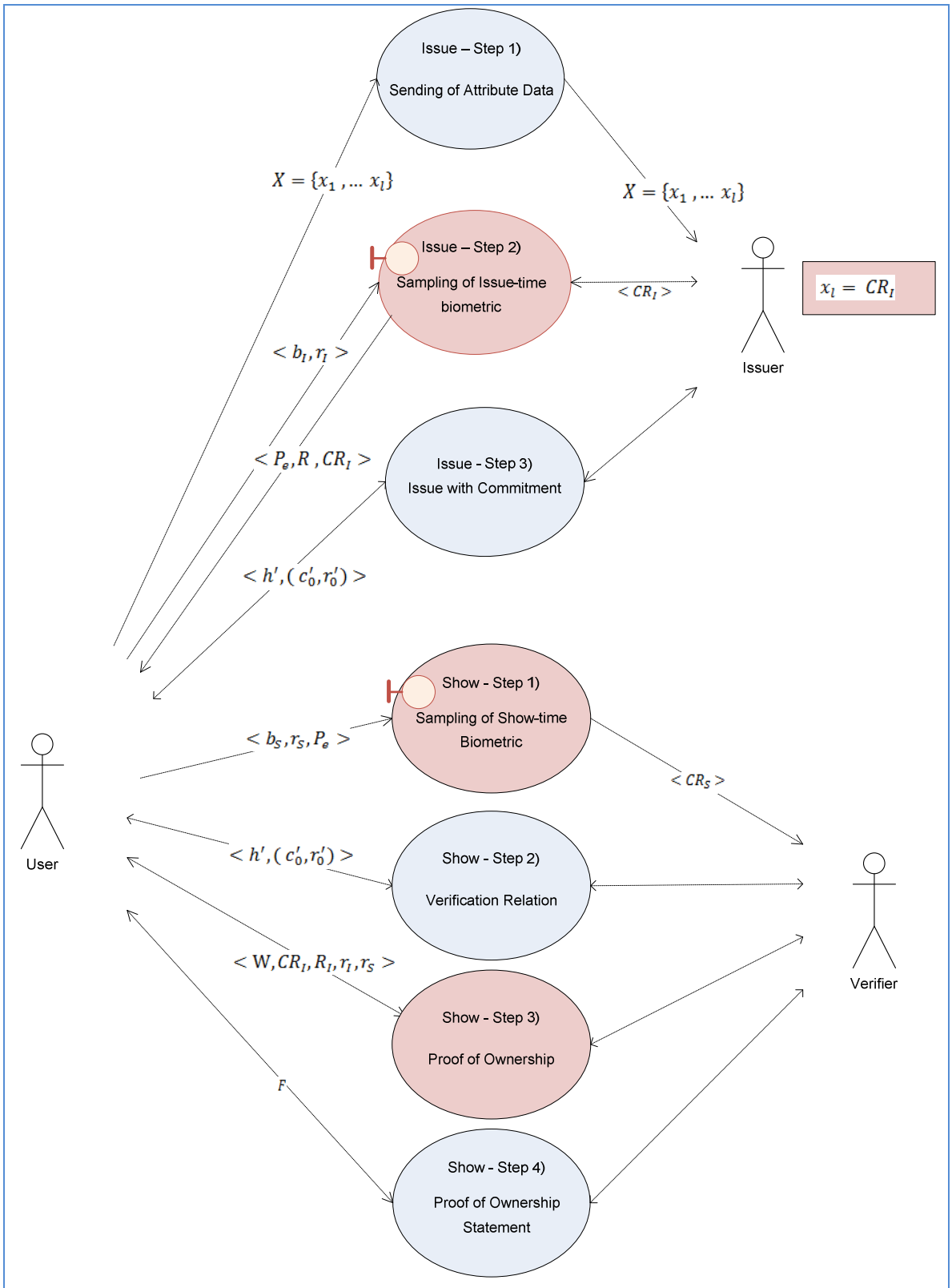

Figure 13. Digital Credential Issue and Show with Fuzzy Extractors

3.4.1 Setup

This section describes the items which the proposed protocol adds to the setup requirements of the underlying credential scheme.

3.4.1.1 The biometric devices

The devices $\{D_i\}$ are all provisioned with the fuzzy extractor indistinguishability adapter $FE_{Ind} :: Gen(...)$, $FE_{Ind} :: Rep(...)$; the fuzzy extractor methods $FE_{Trad} :: Gen(...)$, $FE_{Trad} :: Rep(...)$; the algorithm to compute Pedersen Commitments, $PC :: commit(R_I, r_I)$; the encryption methods $E(...), D(...)$ and the symmetric encryption key $k = \langle k_1, k_2 \rangle$. Each biometric device may be deployed in Issue mode or in Show mode. Devices in deployed in these modes are denoted D_I and D_S respectively.

3.4.1.2 Global Parameters

Group generators g_C, h_C, used for Pedersen commitments by the biometric devices are added to the global parameters of the underlying credential system.

3.4.2 Credential Issue

As in the original digital credential protocol, issue has the User *U* interacting with the Issuer *I* to obtain a digital credential and signature $\langle h', (c_0', r_0') \rangle$. The proposed protocol augments the underlying issue protocol with a) the interaction of *U* and *I* with issue-time biometric sensor D_I; b) the storage of supporting data $\langle P_{e,}, R_I, C_I, r_I \rangle$ on the *U*'s computer, and c) the explicit setting of an attribute by the issuer. The proposed protocol takes advantage of the restrictive blind signature and issuance process of the underlying credential system which results in *U* obtaining $\langle h', (c_0', r_0') \rangle$

3.4.2.1 Issue-Step 1) Sending of Attribute Data

U submits personal attributes *X* to issuer *I*. *I* may verify the validity of attributes as needed through a certification process external to this protocol.

3.4.2.2 Issue-Step 2) Sampling of Issue-time Biometric

U submits biometric $\mathrm{b_I}$ and random value $\mathrm{r_I} \xleftarrow{\mathrm{R}} \mathbb{Z}_{\mathrm{q}}$to $\mathrm{D_I}$. Since $\mathrm{D_I}$ is configured for issue-mode, $\mathrm{D}_I :: \mathrm{Gen}(b_I, r_I)$ is invoked. The device retrieves the symmetric key $k = \langle k_1, k_2 \rangle$ with which it has been configured, and invokes the indistinguishability adapter $FE_{Ind}::Gen(b_I, k)$. The indistinguishability adapter invokes the underlying fuzzy extractor $< P, R > = FE::gen(b_I)$, to generate the fuzzy extractor tuple containing biometrically derived cryptographic key R, and helper data P . The device uses IND-CCA2 encryption to encrypt P, producing $P_e = E(P, k)$. The device then creates cryptographic commitment $\mathrm{CR_I} = \mathrm{g_C}^{R_I}\mathrm{h_C}^{\mathrm{r_I}}$. D_I returns $\langle P_e,\ R_I,\ C_I \rangle$ to U, and $\langle \mathrm{C_I} \rangle$ to I. U securely stores $\langle P_e,\ R_I,\ C_I \rangle$ for later use during the show protocol. I uses $\mathrm{C_I}$ in *Issue-Step 3*, below.

3.4.2.1 Issue-Step 3) Credential Issue with Inclusion of Commitment

In *Issue-Step 3*, organization I sets attribute x_l to the commitment received from $\mathrm{D_I}$ in *Issue-Step 2;* thus $x_l = CR_I$. Once x_l has been set, the protocol proceeds as per [Bra00]: U and I collaboratively participate in the credential Issue and restrictive blind signature process to derive the credential $C = \langle h', (c_0', r_0') \rangle$. By the properties of the underlying digital credential protocol, the values of the attributes X are embedded in the credential which is sealed by the issuer's signature. In our proposed protocol, given that $x_l = CR_I$, U's biometrically derived key R_I is embedded in the digital credential since it is the secret bound to by the Pedersen commitment CR_I. At the end of *Issue-Step 3* as per the underlying digital credential protocol, U obtains credential and blinded signature pair $C = \langle h', (c_0', r_0') \rangle$.

3.4.3 Credential Show

As in the original digital credential show protocol, in the proposed protocol, the user U interacts with verifier V to demonstrate authenticity of the credential/signature pair $\mathrm{C} = \langle \mathrm{h}', (\mathrm{c}_0', \mathrm{r}_0') \rangle$ and to prove a statement F on the attributes X. The proposed protocol augments the underlying digital credential show protocol with a) interaction with biometric device D_S and b) a proof of knowledge to verify U's ownership of C. The proposed protocol takes advantage of a) the verification relation

of the underlying credential system to ensure that no tampering or forgery have occurred with credential and signature $\langle h', (c_0', r_0')\rangle$, and, b) the mechanism for constructing and proving *F*.

3.4.3.1 Show-Step 1) *Sampling of Show-time Biometric*

User and verifier interact with D_S to begin biometric sampling and key regeneration. Since the device is deployed in show-mode $D_S :: rep(b_S, r_S, P_e)$ is invoked with *U* supplying fresh biometric b_S, a new random value r_S and the public data P_e which was obtained in the Issue protocol. The $D_S :: rep(b_S, r_S, P_e)$ method retrieves the symmetric key $k = \langle k_1, k_2\rangle$ and invokes the indistinguishability adapter $FE_{Ind} :: rep(b_S, P_e, k)$ method which runs the decryption algorithm to obtain $P = D(P_e, k)$. $D_S :: rep(b_S, r_S, P_e)$ then attempts to regenerate the cryptographic key using $R_S = FE_{Trad} :: rep(b_S, P)$.The value R_S only corresponds to R_I if b_S is appropriately close to b_I with respect to the given distance metric and if P_e and k correspond to the appropriate issue-time values. After obtaining R_S, the device creates a cryptographic commitment $CR_S = PC :: commit(R_S, r_S) = {g_C}^{R_S} {h_C}^{r_S}$ which it returns to *I* and *V*. The show-time commitment $\mathrm{CR_S}$ will be compared to the issue-time commitment $\mathrm{CR_I}$ to ascertain credential ownership in *Show-Step 3.*

3.4.3.2 Show-Step 2) Verification Relation

The verification relation from the underlying digital credential protocol is used. In the case of Brands' DL-Rep based scheme 1, verification relation accepts if ${c_0}' = H\left(h',\ {g_0}^{r'_0}(h_0 h')^{-{c_0}'}\right)$ is true given $\langle h', (c_0', r_0')\rangle$. This test assures *V* that the credential has not been tampered with and that the signature has not been forged.

3.4.3.3 *Show-Step 3) Proof of Ownership*

In *Show - Step 3) U* proves ownership of credential $\langle h', (c_0', r_0')\rangle$ to *V*. To achieve this, *U* and *V* enter into the proof of knowledge of Section 2.2.5.2, $DLRepWithPC_{U,V}(W, CR_I, R_I, r_I, r_S)$ where $W = X||\alpha_1$, and as defined above, (CR_I, R_I, r_I) and (r_S) are commitment information from the issue and show biometric sampling steps respectively. The required public information for the proof of knowledge is set to h' and CR_S.

By Proposition 2.9, using $\text{DLRepWithPC}_{U,V}$, U successfully proves ownership of $\langle h', (c_0', r_0') \rangle$ since all required data values are known, and both the issue-time commitment CR_I and the show-time commitment CR_S are on the same secret R_I.

3.4.3.4 Show-Step 4) Proof of Credential Statement F

U uses the mechanism of the underlying credential system to prove credential statement F on attributes X/ x_l , as discussed in Section 2.2.5.3. Note that U must not divulge x_l to V. As will be discussed in Section 4.1.2, "Unlinkability of Transactions", revealing x_l during show would allow colluding I and V to link the transactions of U in the issue and show transaction databases.

3.5. Application to Anonymous Credential Protocol

In this section we sketch the approach which would be followed to integrate our protocol extensions into Anonymous credentials of Camenisch and Lysyanskaya. As shown in Figure 14 the basic anonymous credential protocol is and augmented quite similarly as was done for digital credentials, with an extra step, the randomization of credential and signature by the user.

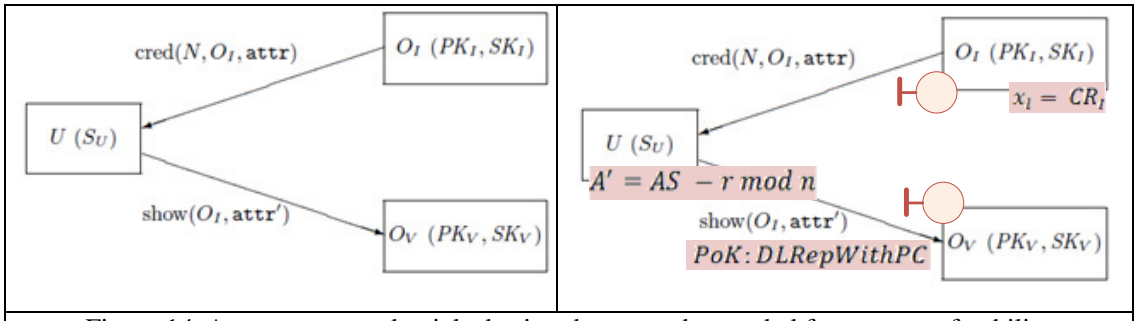

Figure 14. Anonymous credentials: basic scheme, and extended for non-transferability

At a high level the basic anonymous credential protocol proceeds as follows:

1) Issuing organization O_I create a pseudonym and issue credentials C for user U. Pseudonym N is created by O_I using the user's master secret key S_U.
2) Following pseudonym creation, O_I may issue credential C to U by signing a statement S on attr and N.

3) Credential C can be shown by U to a verifying organization O_V. To do so, U uses a zero-knowledge proof to convince O_V of possession signature of O_I on a statement S, of knowing the master secret key S_U related to N.

Since a zero-knowledge proof is used, U does not show the credential or the attributes to O_V. This allows multiple show, and unlinkability. Due to the fact that the credential may not be revealed, we must incorporate a slight variation when we integrate our protocol extension.

Using the notation from [CG08], AC is augmented with our protocol as follows:

During Setup:

As outlined in [CG08]:

1. Choose n such that $n = pq$, with p, q primes
2. Choose, uniformly at random, $\{R_0, \dots, R_{l-1}\}, S, Z \in QR_n$
3. Output the public key $(n, \{R_0, \dots, R_{l-1}\}, S, Z)$ and the secret key p.

Additionally:

4. Choose Pedersen commitment bases g_C and h_C as in previous section
5. Configure devices D_I and D_S for inclusion in the issue and show protocols respectively.

During Issue protocol:

1. The user provides issue-time biometric tuples are returned as normal. Including CR_i which is provided to the Issuer
2. Given messages $\{m_0, \dots, m_{l-1}\}$ set m_{l-1} to CR_i select a random prime number e and a random number v
3. Compute credential A and signature (e,A, v) as specified in

During show protocol:

1. As outlined in [CG08], user randomizes signature (A, e, v) to produce new valid signature (A_0, e, v_0) where $A_0 := AS - r \bmod n$, and $v_0 := v + er$ using random r
2. User reveal A' and uses the proof of knowledge to prove validity of randomized signature as outlined in [CG08].

3. As outlined in previous section, user provides biometric sample and helper data, key is regenerated and commitment CR_S provided to user and verifier
4. User and Verifier proceed with $PoK{:}\,DLRepWithPC$ with the verifier knowing A_0 and CR_S but neither A or CR_I being revealed
5. The underlying protocol proceeds as normal.

The forgery resistance, is provided by the properties of the signature process. While this sketch shows high level integration, detailed presentation, proofs of non-transferability correctness of ownership and unlinkability are not addressed in this thesis but left as future work.

3.6. Chapter Summary

This chapter presented the components of our protocol, and demonstrated the protocol integration with digital credentials. A sketch was also provided of how the protocol would be inserted into the anonymous credentials. Our approach features a specialized biometric device which is inserted into the issue and show protocols of the underlying credential system. In issue mode, the device invokes the generation method of a fuzzy extractor indistinguishability adaptor to produce Pedersen commitment on biometrically derived key and helper data. During the show protocol, a challenge biometric is supplied and the sensor attempts to regenerate the key, and produces another Pedersen Commitment. A proof of knowledge is then used to show that the commitments derived during show and issue are on the same key.

Chapter 4. Security Analysis

4.1. Introduction – Security for Digital Credentials

This chapter presents the proofs of security of our non-transferability extension to cryptographic credentials. The following properties are proved: 1) correctness of ownership, 2) indistinguishability and 3) non-transferability. While the proposed extensions should apply to the anonymous credential scheme of Camenisch and Lysyanskaya, the proofs in this chapter specifically focus on the instantiation of our protocol extensions on Brands' digital credentials as presented in Section 2.2.5.3, and Chapter 3.

4.1.1 Correctness of Ownership

This section proves correctness of ownership for the proposed protocol. In correctness of ownership, the show protocol should accept a credential that is presented by the user to which the credential was properly issued. This property features an honest user complying with the protocol, receiving a properly issued credential, and being assured that the credential will be accepted when it is presented at show-time.

Settings of the proof

The following features of the interaction qualify the user's interaction in the protocols for the proof of correctness of ownership. We refer to these as the "settings of the proof"; these include that:

a) The participating entities are honest

In this proof, the user *U,* issuer *I* and verifier *V*, are all honest, and complying with the protocol as specified.

b) Properly issued credential $\langle h', c_0', r_0' \rangle$

U has interacted with I in the issue protocol, submitting biometric sample b_I, and obtaining credential $\langle h', c_0', r_0' \rangle$ on attributes X which include $x_l = CR_I = {g_C}^{R_I} {h_C}^{r_I}$, where R_I is a cryptographic key derived from biometric b_I and r_I is a random value known only to *U*;

c) Valid information is provided to the show protocol

This follows from the fact that *U* conducts the show protocol honestly. However, we state explicitly here in the arguments supplied to show, biometric sample $\mathrm{b_S}$ belongs to *U*, credential $\langle \mathrm{h}', \mathrm{c}_0', \mathrm{r}_0' \rangle$ is as issued, with no attempt to tamper of forge and credential statement F is valid and true.

Proof

We are required to prove that a credential properly issued to a user will be accepted by a verifier at show-time:

$$\langle h', c_0', r_0' \rangle = Issue_{U,I}(b_I = b(U), X) \rightarrow$$
$$Show_{U,V}(b_S = b(U), \langle h', c_0', r_0' \rangle,\ F"true") == \text{accept}$$

Starting point of proof

Our proof begins after the successful completion of $\langle \mathrm{h}', \mathrm{c}_0', \mathrm{r}_0' \rangle = \mathrm{Issue_{U,I}}(\mathrm{b_I} = \mathrm{b(U)}, \mathrm{X})$. We show this implies that $\mathrm{Show_{U,V}}(\mathrm{b_S} = \mathrm{sample(U)}, \langle \mathrm{h}', \mathrm{c}_0', \mathrm{r}_0' \rangle,\ \text{"true"}) == \text{accept}$, by demonstrating that, within the settings of the proof, each step passes.

Show-Step 1. Biometric sampling always passes

This step always passes: it gathers and prepares information important for the remainder of the proof. U submits biometric sample b_S, $\mathrm{r_S}$ and $\mathrm{P_e}$ to biometric device $\mathrm{D_s}$ which decrypts $\mathrm{P_e}$, reconstructs R and creates commitment $\mathrm{CR_S}$ to be returned to U and *V*. We have that no biometric may have been substituted through a playback attack or other means: the biometric submitted for issue and show are acquired by the trusted biometric device.

At the end of *Show-Step 1,* the individual possesses the information obtained from Issue, namely, the credential $\langle h', c_0', r_0' \rangle$, attributes *X* where $x_l = CR_I$, and se-

curely stored private knowledge R_I, r_I, P_e to which *Show-Step 1*, has added R_S, CR_S, r_S. The verifier holds $\langle h', c_0', r_0' \rangle$ and CR_S.

Show-Step 2. Verification relation passes: credential and signature are valid

By our assumptions for this proof, *Show-Step 2* properly receives credential $\langle h', c_0', r_0' \rangle$ from *U*. Specifically, we know that the credential has been properly constructed, that it is properly signed by *I*, and that there has been no tampering or forgery. As per [Bra00] DLRep-based scheme 1, described in Section 2.4.3, credential $\langle h', c_0', r_0' \rangle$ will pass the verification relation $c_0' = H(h',\ {g_0}^{r'_0}(\ h_0\, h')^{-c'_0}\ )$ under the settings of the proof.

Show-Step 3. Proof of knowledge passes: $\mathbf{CR_I}$, $\mathbf{CR_S}$ are on the same R

As described in Section 2.2.3.5 and 3.1.4.3.2 the zero-knowledge proof of knowledge in *Show-Step 3* passes when *U* can prove knowledge of attributes *X* and private values R_I , r_I, and r_S, and when the issue-time commitment CR_I and the show-time commitment CR_S are on the same cryptographic key R.

By *Assumption 2.2: Biometric soundness*, since *U* engages in both the issue and show transactions, the biometrics supplied to these transactions will be within the prescribed threshold t, thus $b_\mathrm{I}, b_\mathrm{S} \in B_\mathrm{U} \ \rightarrow \mathrm{dis}(b_\mathrm{I}, b_\mathrm{S}) \leq t$. Furthermore given *Proposition 2.4: Fuzzy Extractor* correctness, we know that since $\mathrm{dis}(b_I, b_S) \leq t$, then $Show(\dots)$ will properly recover cryptographic key created by $Issue(\dots)$: thus $R_S = R_I$. This means that the Pedersen Commitments generated during issue and show will be on the same secret $R = R_S = R_I$. Recalling Proposition 2.9: **DLRepWithPC** accepts if commitments are on the same secret, the proof of knowledge of *Show-Step 3*, will therefore pass since the individual has required knowledge *X* where xl = CR_I, R_I, r_I , r_S and since the commitments $CR_S = {g_C}^{R_S} {h_C}^{r_S}$ and $CR_I = {g_C}^{R_I} {h_C}^{r_I}$ are on the same secret $R_I = R_S$.

Show-Step 4. Proof of Credential Statement passes: F is valid and true

The specifics of the credential statement to be proved are not relevant to the proof of correctness of ownership. By the "Settings of the Proof", above, the statement is assumed to be true, and thus Show-Step 4 passes trivially.

In conclusion, given that U obtains $\langle h', c_0', r_0' \rangle$ through interaction with I in "Issue", $\langle h', c_0', r_0' \rangle = Issue_{U,I}(b_I = b(U), X)$, if U then submits $\langle h', c_0', r_0' \rangle$ to V in $Show_{U,V}(b_S = b(U), \langle h', c_0', r_0' \rangle, T)$, the show protocol will pass. This proves correctness of ownership, whereby if a user engages in a show protocol, presenting a credential which was properly issued to her, the show protocol will be accepted appropriately. ■

4.1.2 Unlinkability of Transactions

This section proves the proposed protocol's property of unlinkability under which the privacy of users is protected against an attacker who seeks to correlate user transactions with the goal of creating user profiles, gathering usage data, or learning user identity information. Our proof proceeds in 2 parts: first we prove the unlinkability between issue and show transactions, and then, the unlinkability between show transactions. In both cases, we seek to prove that a polynomially-bounded attacker has a negligible chance of linking transactions by the same user.

Attack Game

The environment for the attack game includes two service providers I^* and V^* who engage in issue and show transactions with honest users, and store data tuples for each transaction into databases I_{DB}, and S_{DB} respectively. Our model features two polynomially-bounded attackers $\hat{\mathcal{A}}_1$ and $\hat{\mathcal{A}}_2$ who seek to find transactions performed by the same user in the databases collected by service providers. $\hat{\mathcal{A}}_1$, the attacker against the unlinkability of issue and show transactions, has access to both databases, and must link between them, whereas $\hat{\mathcal{A}}_2$ attacks unlinkability of show transactions and has access only to S_{DB}.

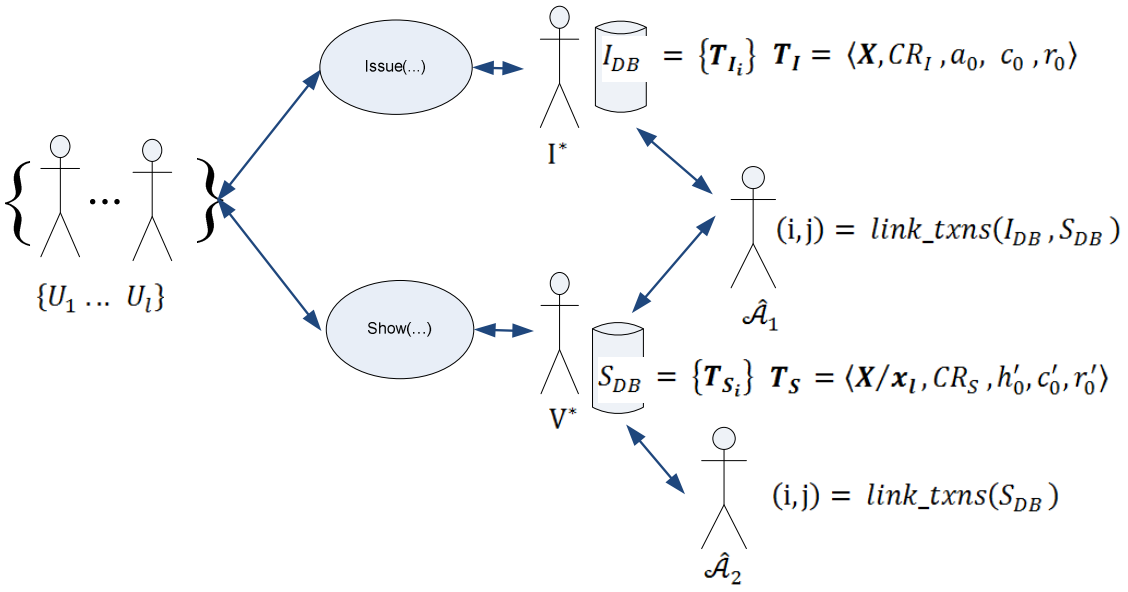

Figure 15. Unlinkability Attackers

The tuples of the databases contain the service provider views of protocol transcripts: only the data which the corrupt service providers can see. The issue tuple $T_I = \langle X_I = X, CR_I, a_0, c_0, r_0 \rangle$ contains the full set of user attributes X; the issue-time biometric derived key commitment CR_I (which is also stored in x_l); and intermediate values a_0, c_0, r_0 computed during the restricted blind signature process by I^* of the underlying credential protocol. The show tuple $T_S = \langle X_S \subseteq X/x_l, CR_S, h', c'_0, r'_0 \rangle$ contains a subset of the users attributes $X_S \subseteq X/x_l$ (by constraints of our extension to the show protocol, X_S excludes x_l), the show-time commitment CR_S, and h', c'_0, r'_0 the digital credential data.

In our model, the logic of the attacks is encapsulated in a procedure. For issue-show unlinkability, $\hat{\mathcal{A}}$ selects procedure (i,j) = $link\_txns(I_{DB}, S_{DB})$ which returns i and j, indices into I_{DB}, and S_{DB} respectively. $\hat{\mathcal{A}}$ has succeeded in linking issue and show transactions if rows i and j correspond to transactions by the same user.

For unlinkability of show transactions, $\hat{\mathcal{A}}$ selects procedure $(i, j) = \text{link\_txns}(S_{DB})$ which returns i and j where i≠j. Again, $\hat{\mathcal{A}}$ has succeeded in his attack if i and j correspond to transactions by the same user.

To determine if the transactions were performed by the same user a challenger having access to user data can verify that the biometrically derived keys of the users initiating the transactions are equal $R_i = R_j$

In each case, the protocol has been proven secure against the attack on unlinkability if the attacker has negligible chance of success

4.1.2.1 Unlinkability between issue and show transactions

We first address unlinkability of issue and show transactions (thus unlinkability of rows across I_{DB} and S_{DB}). As presented above, this data is represented by the tuples $T_I = \langle X_I = X, CR_I, a_0, c_0, r_0 \rangle$ and $T_S = \langle X_S \subseteq X/x_l, CR_S, h', c_0', r_0' \rangle$.

We can reduce the scope by excluding both linkability in terms of attribute subsets (X_S and X_I) and linkability based on signature intermediate data ($\langle a_0, c_0, r_0 \rangle$ and $\langle h', c_0', r_0' \rangle$)) since these have both been addressed by Brands [Bra00]. We first provide a rapid sketch of why these may be excluded and then focus on proving unlinkability in terms the commitments CR_I and CR_S.

The linkage of transactions by subsets of attributes from X_S and to the master set X_I is a real concern. During the issue protocol, the user's divulges her entire attributes set to the service provider $X_S = X$. During the show protocol a subset of attributes is revealed $X_S \subseteq X/x_l$. Depending on which subset of attributes is revealed during the show protocol and their values, it could be possible to link transactions. The choice of which attributes to reveal however belongs to the user. For the purposes of these proofs, we exclude linkage of transaction based on attribute subsets, and assume that the user exercises judgement during the show protocol, not to reveal attribute subsets which can be used by the attacker to guess identity with non-negligible advantage. Note that if a user chooses to be careless about revealing attributes that will identify her, then no technical protocol can prevent linkability, thus the burden of protection here lies with the user and not with the protocol.

Next we inherit security against transaction linkage using signature intermediate values from Brands. During the blinded signature process the Issuer sees intermediate data $\langle a_0,\ c_0\ ,\ r_0\ \rangle$; during show the digital credential and signature $\langle h', c_0', r_0' \rangle$ are presented for the verification relation. We refer to the proof of Brands' in [Bra00]; however, to sketch, the final signature values are dependent on unique random values known only to the user, and thus the intermediate data seen by the issuer cannot be linked to the signature.

Having excluded attribute subsets and intermediate signature data as potential data elements to link transactions, we can simplify the tuples T_I and T_S ruling out all attributes other than CR_I and CR_S. The databases may be reduced to two collections of commitments: CR_I and CR_S, and the attack function restated as $(\mathrm{i}, \mathrm{j}) = link\_txns(CR_I, CR_S)$.

To win the game, $\hat{\mathcal{A}}$ must either a) derive the secret R which underlies a commitment or b) distinguish between commitments based on the secret on which they are formed. By Proposition 2.1: "Pedersen Commitments are Perfectly Hiding", it is not possible for the attacker to derive the secret in polynomial time, or in fact, in any time.

By, Proposition 2.1: "Pedersen Commitments are Perfectly Hiding" and Proposition 2.3: "Pedersen Commitments are Indistinguishable", given a collection of Pedersen commitments, the Attacker has no advantage at determining whether or not they commit to the same or to different secrets. The attacker has no advantage in correlating any commitment in CR_I with any commitment in CR_S.

Given that unlinkability for attribute linkage and signature data has been addressed by [Bra00] and that $\hat{\mathcal{A}}$ can neither derive R nor correlate between commitments based on the key they conceal, we have that issue and show transactions are unlinkable■

4.1.2.2 Unlinkability between show transactions

The second aspect of unlinkability of transactions concerns linking separate show transactions by the same user. To begin, we recall that in Brands' digital credentials protocol, an issued credential can only be shown once. Presenting the same

credential in two distinct show transactions, allows trivial linkage based on the common signature data $\langle h', c_0', r_0' \rangle$. This limitation can be addressed within the context of Brands' digital credential scheme by issuing multiple credentials for the same (set of) privilege(s), and using each one time only, as required. For the purposes of this thesis, we assume that two transactions from the same user do not violate Brands' single-show restriction, and specifically, that separate digital credentials where issued with distinct signature data $\langle h', c_0', r_0' \rangle$.

Furthermore, as in the previous section, with respect to linkability based on common subsets of attributes, we assume that in the case where multiple transactions by the same user are present in the database, the user has not divulged common attributes which allow them to be linked.

Due to the restrictive blinding issue process, tuple $\langle h', \langle c_0', r_0' \rangle \rangle$ cannot be used to link transactions based on the underlying values. We assume that each time a user gets a credential issued, a new alpha value α_1 is generated. In this case, each credential generated for a user $h' = hh_0{}^{\alpha_1} \bmod q$ includes a random value α_1. The h' therefore appears random and can't be used for transaction linkage. Similarly $\langle c_0', r_0' \rangle$ are blinded by random values. As the random values blinding each of $\langle h', \langle c_0', r_0' \rangle \rangle$ are independent there is no statistical relation between these values within a row, or among rows.

Having established the assumption of single show which excludes linkage using signature data, and also excluded linkage by common attributes, the problem of linking separate show transactions can be reduced to the problem of finding a pair commitments on the same secret within a collection of commitments where the secrets are not known. Thus $(i, j) = link\_txns(CR_S)$ would return a pair of indices of two commitments on the same biometrically derived key R.

Recalling the discussion on Pedersen Commitments (Proposition 2.3: Pedersen Commitments are Indistinguishable) it is not possible to determine with any significant advantage which commitments within a collection are on the same plaintext. The situation with $(i, j) = link\_txns(CR_S)$ is identical: $\hat{\mathcal{A}}$ cannot distinguish with any advantage any commitments on the same biometrically derived key R within CR_S.

Therefore, it is impossible for $\hat{\mathcal{A}}$ to link show transactions based on observing the commitments alone. Further, having ruled out linkage by attribute subsets, and assumed single show transactions, we have that the proposed protocol is secure against show transaction linkage. ■

4.1.3 Non Transferability of Credentials

Non-transferability is the flip side of correctness of ownership: while correctness of ownership asserts that the protocol will accept if *U* was issued *C*, non-transferability asserts that the protocol will fail if *U* was not the party to which *C* was issued. The proof proceeds by analysing possible attacks on the protocol and showing that these are not possible. This proof uses the following notation:

Polynomially-bounded attacker	$\hat{\mathcal{A}}$
Honest Users	$\mathbb{U} = \{U_1 \dots U_l\}$
Corrupted Users	$\mathbb{U}^* = \{\mathrm{U}_1^* \dots \mathrm{U}_n^*\}\ where\ \mathbb{U}^* \subseteq \mathbb{U}$
Challenger:	$\langle Issue(\ ), Show(\ )\rangle$
Lending User	U_i^*
Borrowing User	U_j^*
Random selection from Honest Issuers	$I_{t_n} \xleftarrow{r} \{I_1 \dots I_j\}$
Random selection from Honest Verifiers	$V_c \xleftarrow{r} \{V_1 \dots V_k\}$
Local Data authentication helper data private data credential-signature data	$L = \{B, P, C\}\ where$: $B = \langle r_I, (P_e, R)\rangle$ $P = \{x_1 \dots x_l, \alpha_1\}$ where $x_l = CR_I$ $C = \langle h', (c_0', r_0')\rangle$
Augmented attribute set X^{Δ}.	$X^{\Delta} \subseteq X_{U_j^*} \cup \mathcal{B}$ where $\mathcal{B} = \{x \mid \exists x \in X_{U_i^*} \wedge x \notin X_{U_j^*}\}$
Challenge credential statement.	$\mathcal{F}^{\Delta}(X^{\Delta})$
Polynomially Bounded Attack Strategy	$\langle U_i^*, U_j^*, L^{\Delta}, X^{\Delta}, \mathcal{F}^{\Delta}\rangle = f(\mathbb{U}^*, \mathbb{L}^*)$ where L^{Δ} is an attacker change on local data *L*

4.1.3.1 Attack Strategies

4.1.3.1.1. Preliminary Discussion

Assumptions

No replay. Due to *Assumption 2.1: Trusted device which detects liveness* we have that only real biometrics enter the enrollment and verification steps; that no replay is possible.

Secure Communication. We assume secure communication, that the endpoints are as specified in the interaction, and that $\hat{\mathcal{A}}$ has no advantage through impersonation or wire taping.

Given the assumptions above, during the challenge, $\hat{\mathcal{A}}$ is bound to the sequence of steps in the show protocol. We restrict the attacks considered here to manipulations of the input data, their computational possibility, and their probability of resulting in a successful attack. Table 1 shows a number of attack strategies based on the variations of local data L = $\{B, P, C\}$. There may exist other possible manipulations of local data, however those listed here, illustrate the main variations and the resilience of the proposed protocol.

Attack	B^{Δ}	P^{Δ}	C^{Δ}	Description
1	$B^{\Delta} = B_{U_i^*}$	$P^{\Delta} = P_{U_i^*}$	$C^{\Delta}=C_{U_i^*}$	No Change to Local Data
2	$B^{\Delta} = B_{U_j^*}$	$P^{\Delta} = P_{U_i^*}$	$C^{\Delta}=C_{U_i^*}$	Borrower biometric data; Lender Commitment
3	$B^{\Delta} = B_{U_j^*}$	Set $x_l = CR_{I_j}$ s.t. $CR_{I_i} \neq CR_{I_j}$	$C^{\Delta}=C_{U_i^*}$	Set x_l without changing h'
4	$B^{\Delta} = B_{U_j^*}$	Set $x_l = CR_{I_j}$ s.t. $CR_{I_i} \neq CR_{I_j}$. Derive $\alpha^* \leftarrow \mathbb{Z}_q$ s.t. $h^{*\prime} = h'$	$C^{\Delta}=C_{U_i^*}$	Derive value for α^* such that new x_l works without changing h'
5	$B^{\Delta} = B_{U_j^*}$	Set $x_l = CR_{F_j}$ s.t. $CR_{I_i} = CR_{F_j}$	$C^{\Delta}=C_{U_i^*}$	Set $x_l = CR_{F_j}$ (No change required to h')
6		Pre-Issue: $Set\ x_k = CR_{F_j}$		Attempt to store commitment prior to signature

Table 1: Local Data variation for each attack strategy. The attack strategies in are shown here cross-referenced to each component of the Local Data L = $\{B, P, C\}$ $where$: $B = \langle r_I, (P_e, R)\rangle$ $P = \{x_1 \ldots x_l, \alpha_1\}$, $x_l = CR_I$ and $C = \langle h', (c_0', r_0')\rangle$

4.1.3.1.2. Data manipulation by the Attacker

$\hat{\mathcal{A}}$ must prepare for the challenge by selecting lending and borrowing users, creating borrowed permissions and credential proposition, and making optional changes to the local data, as may benefit the attack. Given *Assumption 2.2: Soundness of Biometric Modality*, we have that for any two distinct users i, j the distance between their biometrics exceeds the threshold: $dis(b(i), b(j)) > t$. For the purposes of our security proof, we assume that any two users U_i^* U_j^* may be selected by $\hat{\mathcal{A}}$ as candidate lender and borrower respectively in step 3 of the attack game.

4.1.3.1.3. Borrowed permissions and credential proposition

Modified attribute set. Under the non-transferability attack game, an augmented attribute set X^{Δ} must be produced which contains incremental privilege for the borrower. The augmented attribute set can be created by borrowing the lender's attribute set $X_{U_i^*}$ and then subsequently modifying it as needed to suit the needs of the attack (c.f. *Attack 3*, *Attack 4*), or using the borrower's attribute set $X_{U_j^*}$ as the base set and borrowing one or many attributes from the lender by changing the corresponding value in $X_{U_j^*}$. The second approach immediately changes a value within X^{Δ} the base attribute set, which changes the calculated value of the credential $h' = hh_0^{\alpha_1}$ and leaves the borrower with the problem of obtaining a new signature on this modified credential without which the verification relation in *Show-Step 2* will not pass. The security proof is indifferent to the manner in which X^{Δ} is created. From the perspective of $\hat{\mathcal{A}}$ borrowing $X_{U_i^*}$ as the base attribute set, this only defers the problem of invalid signatures until the modified local data L^{Δ} is prepared. The attacks presented in Table 1 are based on initially borrowing the entire attribute set $X_{U_i^*}$ from the lender.

Formula. Defining a valid formula on those attributes is trivial, if $\hat{\mathcal{A}}$ has successfully borrowed attribute set and avoided any signature impact. For the attacks in this section, we assume a valid formula is constructed by $\hat{\mathcal{A}}$ on the modified attribute set.

4.1.3.1.4. Local Data Change Attacks

Changes to Fuzzy Extractor Commitment Data. In Table 1, attempts are made to modify the key commitment tuple $B = \langle r_I, (P_e, R)\rangle$. Initially the lender's original values are used (c.f. Attack 1) ; subsequently, the borrower's values are substituted into the lender's tuple (c.f. Attack 2).

Changes to Credential and Signature. No attacks are shown in Table 1 which attempt to change the credential and signature tuple $C = \langle h', (c_0', r_0')\rangle$ for two reasons: 1) by "*Proposition 2.10: Signatures on credentials are unforgeable*" it is impossible for $\hat{\mathcal{A}}$ to forge a signature; and, 2) by *"Assumption 2.7: Verified Identity during Credential Issuance"*. The second option is not valid because since the only way $\hat{\mathcal{A}}$ can have the modified tuple X^{Δ} re-signed is by going through a legitimate issue process with I_j. If all steps of issues pass (including the out-of-band attribute verification process), the signed credential obtained by U_j^* would no longer qualify as a borrowed credential with an augmented attribute set: it is a valid credential, legitimately issued to, and owned by U_j^*.

Changes to Private Data. The private data $P = \{x_1 \ldots x_l, \alpha_1\}$ presents interesting possibilities for $\hat{\mathcal{A}}$ to make changes. In *Attack 3* x_l is changed in an attempt to insert the borrower's issue time commitment. In *Attack 4* $\hat{\mathcal{A}}$ attempts to change both x_l and α_1 to insert the borrower's issue time commitment in a manner that does not invalidate the signature.

Attack 1: No Change to Local Data

In this attack $\hat{\mathcal{A}}$ adopts the most naïve strategy: none of the local data is modified: $L^{\Delta} = L_{U_i^*}$.

Lemma 1: An attack which relies on regeneration of a user's keys by a different user has negligible probability of success

This attack is thwarted at show time, as the borrower must submit her own biometric to D_S. D_S attempts to use the borrower's biometric to regenerate a key equal to the lenders. Given the fuzzy extractor property Proposition 2.7 we have that if $dis\left(b(\mathrm{U}_i^*), b\left(\mathrm{U}_j^*\right)\right) > t$ and <R, P> = $FE{:}\,Gen(b(U_i^*))$ and R' =

FE:Rec($b(\mathrm{U}_j^*), P_{\mathrm{U}_j^*}$), then with overwhelming probability $R' \neq R$. This attack will fail with overwhelming probability, since by *"Proposition 2.9:* **DLRepWithPC** *accepts if commitments are on the same secret",* when $R_{I_i} \neq R_{\mathrm{S}_j}$ DLRepWithPC in *Show - Step 3)* will fail■

Attack 2: Borrower Helper Data; Lender Attributes

In this attack $\hat{\mathcal{A}}$ changes the authentication helper data component *B* of the local data to be that of the borrower's, while leaving the attribute holding the lender's issue time commitment unchanged, thus: $L^{\Delta} = \left(B_{U_j^*},\ P_{U_i^*},\ C_{U_i^*}\right)$.

Lemma 2: An attack which relies on FE generation of identical keys for different users has negligible probability of success.

Since the borrower's P_e where substituted into L^{Δ} through $P_{U_j^*}$ *Show-Step 1* successfully regenerates the borrower's biometrically derived key $R_{U_j^*}$. However since independent calls to $FE::Gen()$ are unlikely to result in identical keys (see: Proposition 2.6), the probability that $R_{U_j^*} = R_{U_i^*}$ is negligible. Ultimately, this attack will fail with overwhelming probability, since by Proposition 2.9, if $R_{I_i} \neq R_{\mathrm{S}_j}$ then DLRepWithPC in *Show - Step 3* fails■

Attack 3: Set commitment without changing credential

In this attack $\hat{\mathcal{A}}$ places a) a commitment on the borrower's biometric key data CR_{I_j} in x_l and b) sets the authentication helper data to be that of the borrower $B_{U_j^*}$, thus: $L^{\Delta} = \left(B_{U_j^*},\ P^{\Delta},\ C_{U_i^*}\right)$ where $x_l^{\Delta} = CR_{I_j}$ and $CR_{I_j} \neq CR_{I_i}$. This is an attempt to counter the failure encountered in Attack 2 where the PoK failed due to incompatible commitments.

Lemma 3: An attack which modifies x_l changing h', will fail in Show - Step 2.

Under Attack 3, the proof of knowledge in *Show-Step 3* would pass (since $\mathrm{CR}_{\mathrm{I_j}} = \mathrm{g_C}^{\mathrm{R_{I_j}}}\mathrm{h_C}^{\mathrm{r_I}}$ and $\mathrm{CR}_{\mathrm{S_j}} = \mathrm{g_C}^{\mathrm{R_{S_j}}}\mathrm{h_C}^{\mathrm{r_S}}$ are on the same key $\mathrm{R}_{\mathrm{I_j}} = \mathrm{R}_{\mathrm{S_j}}$). Attack 3

unconditionally fails, however, in step *Show-Step 2* because the signature $C_{U_i^*}$ has become invalidated. The calculated value of the credential h^{Δ} has changed due to the change in attributes where $x_l^{\Delta} \neq x_l$ and therefore $h^{\Delta} \neq h'$. The new value h^{Δ} no longer verifies with the signature data (c_0', r_0') which, by construction, will only pass with the original credential h' from $C_{U_i^*} = \langle h', (c_0', r_0') \rangle$ ■

Attack 4: Derive new alpha to change commitment without change to h'

Attack 4 begins like Attack 3, setting x_l^{Δ} to an issue-time commitment CR_{I_J} on the borrower's biometric key. $\hat{\mathcal{A}}$ seeks to overcome the weakness of Attack 3 by attempts to derive a new value for α_1^{Δ} which would allow h' to remain constant (despite the changed attribute x_l^{Δ}), and thus allow the verification relation to pass (where it failed in *Attack 3*) .

The local data constructed by $\hat{\mathcal{A}}$ is set as follows: $\mathrm{L}^{\Delta} = \left(\mathrm{B}_{\mathrm{U}_j^*},\ \mathrm{P}^{\Delta},\ \mathrm{C}_{\mathrm{U}_i^*}\right)$ where $\mathrm{B}_{\mathrm{U}_j^*}$ is the borrower's biometric helper data, $\mathrm{P}^{\Delta} = \left\{\mathrm{x}_1 \ldots \mathrm{x}_l^{\Delta}, \alpha_1^{\Delta}\right\}$ and $\mathrm{x}_l^{\Delta} = \mathrm{CR}_{\mathrm{I}_j}$ and $\mathrm{CR}_{\mathrm{I}_j} \neq \mathrm{CR}_{\mathrm{I}_i}$ .and $\alpha_1^{\Delta} \in \mathrm{Z}_q$ such that $\mathrm{h}' = \left(\mathrm{g}_1{}^{\mathrm{x}_1} \ldots \mathrm{g}_l{}^{\mathrm{x}_l^{\Delta}}\right) \mathrm{g}_o{}^{\alpha_l^{\Delta}}$

Lemma 4: If discrete logarithm is hard, Attack 4 is not feasible.

$\hat{\mathcal{A}}$ must find a value for $\propto_1^*$ such that $h' = h'^*$ where $h' = g_1^{x_1} g_2^{x_2} \ldots . g_l^{x_l} h_0^{\alpha_1}$ and $h'^* = g_1^{x_1} g_2^{x_2} \ldots . g_l^{x_l^*} h_0^{\alpha_1^*}$. Let $= g_1^{x_1} g_2^{x_2} \ldots . g_l^{x_l^*}$. We have then that $h'^* = y(h_0^{\alpha_1^*}) \rightarrow \alpha_1^* = DL(g_l{}^{(x_l - x_l^*)} h_0{}^{\alpha_1},\ h_0)$. Finding α_1^* depends on the ability of $\hat{\mathcal{A}}$ to solve the discrete log (even if $\hat{\mathcal{A}}$ learns the values of x_l and α_1 from the lender). ∴ If discrete logarithm is hard, Attack 4 is not feasible ■

Attack 5: Derive falsified commitment requiring no change in h'

In this attack, $\hat{\mathcal{A}}$ attempts to derive a manufactured commitment which is compatible with the verification time commitment and the originally signed credential in that a) it is on the borrower's biometrically derived key, and b) it has the same value as the original lender issue time commitment. Finding such a commitment would allow show step 2 and step 3 to both pass. Thus, the local data is set to

$L^{\Delta} = \left(B_{U_j^*},\ P^{\Delta},\ C_{U_i^*}\right)$ where, $\mathrm{P}^{\Delta} = \{x_1 \ldots x_l^{\Delta}, \alpha_1\}$ *and* $x_l^{\Delta} = CR_{F_j}$ where $CR_{F_j} = CR_{I_i}$ and $\frac{CR_{F_j}}{CR_{S_j}} = {h_C}^r$.

If $\hat{\mathcal{A}}$ can deduce *r*, this attack allows both the steps *Show-Step 2* and *Show-Step 3* to pass. The verification relation in *Show-Step 2* would pass since the signature data $C_{U_i^*} = \langle h', (c_0', r_0') \rangle$ remains consistent with the attribute data which would remain unchanged given that $x_l = CR_{F_j} = CR_{I_i}$. The proof of knowledge in step *Show - Step 3* would pass since a) $\frac{CR_{F_j}}{CR_{S_j}} = {h_C}^r$ and b) the borrower, U_j^* would know required values for the proof of knowledge: $< x_1 \ldots x_l^{\Delta}, \alpha_1,$ r>. *Attack 5* depends on $\hat{\mathcal{A}}$ solving for *r*, which in turn depends on $\hat{\mathcal{A}}'s$ ability to solve the discrete log as shown below.

Lemma 5: If discrete logarithm is hard, Attack 5 is not feasible.

$\hat{\mathcal{A}}$ must find a value CR_{F_j} where $CR_{F_j} = CR_{I_i}$ and $\frac{CR_{F_j}}{CR_{S_j}} = {h_C}^r$.

Let $CR_{F_j} = g_C^{R_{I_j}} h_C^{r_{F_j}}$ and $CR_{I_i} = g_C^{R_{I_i}} h_C^{r_{I_i}}$. The values of CR_{I_i} , r_{I_i} , R_{I_j} ,R_{I_i} are all known. Solving $CR_{F_j} = CR_{I_i}$ becomes solving $r_{F_j} = \mathrm{DL}(\mathrm{w},\ h_C)$) where $w = g_C^{R_F} * h_C^{r_{I_i}}$ and $R_F = R_{I_i} - R_{I_j}$.

Solving $r_{F_j} = \mathrm{DL}(\mathrm{w},\ h_C)$) depends on the ability of $\hat{\mathcal{A}}$ to solve the discrete log.

∴ If discrete logarithm is hard, *Attack 5* is not feasible ■

Attack 6: Attempt to store $\mathbf{CR_{I_j}}$ prior to signature

While generally blocked in advance by the issuer's out-of-band attribute verification process (see 3.4.2.1), this attack consists of an attempt to place data useful to the attacker in an unused attribute for later retrieval during pre-challenge manipulation. In this attack, $\hat{\mathcal{A}}$ attempts to derive advantage by setting an attribute value to a commitment on the borrower's biometrically derived key, hoping that it could be useful in passing the Show protocol's proof of knowledge and verification relations.

Thus, prior to the challenge, the local data is set to $L^{\Delta} = \left(B_{U_j^*},\ P\ ,\ C_{U_i^*}\right)$ where, $\mathrm{P}^{\Delta} = \{x_1 \ldots, x_k\ , x_l\ , \alpha_1\}, x_k\ =\ CR_{I_j}, and\ \ x_l\ =\ CR_{I_i}$.

Lemma 6: Attack 6 offers no advantage.

By definition, the index *l* of the attribute to be verified is pre-determined and known to the verifier. The attacker cannot derive advantage by letting the proof of knowledge proceed on base k rather than base l; this is not achievable by protocol definition. Furthermore, since each of the bases $g_0, \ldots, g_l$ are distinct, an attempt to move the borrower's commitment value from x_k to x_l would impact the credential value and invalidate the signature.

∴ *Attack 6* is not feasible and offers no advantage ■

4.1.3.2 Security Proof

Suppose the attacker $\hat{\mathcal{A}}$ may only use computation and data manipulation means to challenge the protocol. The attacker has 2 intervention opportunities to prepare data: 1) pre-issue manipulation: here arbitrary attribute data can be inserted into X, with the goal of augmenting A's advantage; 2) pre-challenge manipulation: here data preparation consists of modifying local data L^{Δ}. Neither of these modifications is mandatory. The attacker must however, create a modified attribute set and boolean proposition.

The attacker may choose to forgo data manipulations. In this case, he has negligible probability of winning the game. To show transfer of credential, $\hat{\mathcal{A}}$ must show augmented privileges. This is done by preparing a modified attribute set which contains different privilege than he had originally. In Brands' original digital credential scheme, this could be done by transferring the entire credential and using it. This simple copy-based transferal is analogous to the approaches described in *Attack 1* and *Attack 2* where the attacker chooses not to manipulate data in either the pre-issue or pre-challenge intervention. As described in *Lemma 1*, *Lemma 2* , these approaches both fail with overwhelming probability in *Show-Step 3*.

The attacker may choose to try pre-challenge data manipulations. In this case, the proof of knowledge and verification relation block successful attack. To pass Show-Step 3, $\hat{\mathcal{A}}$ must ensure that x_l in the borrowed attribute set $X_{U_i^*}$ contains a commitment on the key generated by the borrower's biometric. To do so, $\hat{\mathcal{A}}$ could attempt to alter the value of x_l (see Attack 3). Changing x_l in this manner, however, would invalidate the signature causing the verification relation in *Show-Step 2* to fail and abort the protocol. In order to pass *Show-Step 2* using this approach, the borrower would have to forge a signature. This is not possible based on Proposition 2.10 "Signatures on credentials are unforgeable".

To pass *Show - Step 2* as well as *Show - Step 3* $\hat{\mathcal{A}}$ must insert a compatible x_l without invalidating the signature. To do so, then either the old commitment value should not be changed, but new opening values which include the borrower's key should be found, (see *Attack 5*) or or a new value for α_1 must be calculated which allows h' to remain unchanged (see *Attack 4*). These approaches, however, both require the ability to solve the discrete log (see *Lemma 3*, *Lemma 4*).

The attacker may choose to try pre-issue data manipulations. In this case, the trusted sensor, and construction of DL-Rep bases stop successful attack. As we see in *Attack 6*, attempts to hide an issue time commitment for the borrower in the attribute data to be signed and later used in pre-challenge preparation provide no advantage due to properties of the credential generation formula, the proof of knowledge verification algorithm, and the signature verification process.

Assuming $\hat{\mathcal{A}}$ cannot solve the discrete log problem, the only feasible means of attack then become *Attack 1* and *Attack 2*, which have negligible probabilities of winning. ■

4.2. Chapter Summary

This chapter has presented the proofs of security of our non-transferability extension to cryptographic credentials for correctness of ownership, unlinkability and

non-transferability. Correctness of ownership is delivered due to the properties of the biometric modality. Unlinkability comes predominantly from the properties of Pedersen commitments. Non transferability comes from the verification relation, and the proof of knowledge in the show protocol in combination with unforgeability of signatures in digital credentials and the intractability of the discrete log problem.

Chapter 5. Comparison to Previous Work

The following section compares the proposed algorithm with that of Adams [Ada11]. We refer to the proposed protocol as Alg_1, to that proposed by Adams as Alg_0.

5.1. System Maintenance

To compare system maintenance, we consider values which must be configured when the system is first initialized or when individual users or sensors are added to the system. Such values may include, for example, group parameters, entity private and public keys or any values which must be configured into the biometric devices. The comparison of the number parameters and the frequency of change of these values can indicate relative difficulty of maintenance between systems.

In Alg_0 the global system parameters require the addition g_C and h_C. The sensor requires the ability to receive a random value from Alice, and the ability to compute a Pedersen commitment. Furthermore, in Alg_0 the hamming distance threshold t is maintained by the credential verifying organizations v. This allows different organizations to maintain different thresholds which can offer significant flexibility.

In Alg_1, g_C and h_C, the group generators for Pedersen commitments, must similarly be added to the global system parameters. Additional maintenance is imposed by Alg_1 due to the fuzzy extractor indistinguishability adaptor which is provisioned into each biometric device. These algorithms require two important values common to all devices: the distance threshold t and the symmetric encryption key. There is a choice of whether the values should be entered into each device or should be available online to the devices, however this is an important difference adversely affecting the maintainability of Alg_1.

Alg_1 adds complexity because it requires fuzzy extractor scheme and associated parameters to be on the sensor. As with the algorithm in [DFM98], Alg_1 sensor values must be consistent across all sensors.

Due to the simplicity of the device, Alg_0 may be preferable to Alg_1.in terms of system maintenance.

5.2. Imposed requirement on underlying Cryptographic Credential System

Both systems propose a technique which can be added onto an existing cryptographic credential system. Alg_0 adds 1) configuration parameters, 2) a biometric challenge in the issue time protocol 3) requirement to store data on the user's system 4) requires use of one user attribute to hold a commitment; 5) a biometric challenge in the show time protocol; 6) a proof of knowledge of biometric in the show protocol; 7) t interactive proofs of knowledge for bit-wise equality in the show protocol.

Alg_1 adds items 1 through 6, however removes item 7), the t proofs of knowledge for bit-wise equality.

Alg_1 may be preferable in terms of imposed requirements on the underlying protocol.

5.3. Generality in terms of Biometric Modalities and Distance Metrics

Both systems use biometrics to ensure non-transferability. Both impose some restrictions regarding biometric modality, metric spaces and distance functions to be used.

Alg_0 is specifically constructed for the hamming distance. As such, it is only applicable to such biometric modalities which use the hamming distance to best measure template similarity.

Alg_1 builds on fuzzy extractors, and as such is more general. A number of distance metrics have been explored for fuzzy extractors, and a number of construc-

tions have been proposed on various biometric modalities including iris, fingerprint and face.

Alg_1 may be preferable with respect to generality of biometrics.

5.4. Computation Cost

Analysis in terms of basic operations

For the analysis that follows, we express higher-level protocols in terms of the number of modular additions, multiplications and exponentiations of which they are composed.

Table 4, below, shows the complexity of modular arithmetic operations. The purpose here is to give a frame of reference for the relative complexity. Similarly, sample execution times are shown for different values of the security parameter k.

		number of bits (k = log n)		
Operation	**Complexity**	**k= 512**	**k= 1024**	**k= 2048**
Addition	O(*n*)	0.002 ms	0.002 ms	0.004 ms
Multiplication	O(*n* log *n*)	0.017 ms	0.059 ms	0.229 ms
Exponentiation	*O*(n^2)	7.0 ms	41.0 ms	312.0 ms

Table 4. Cost of Basic Operations

In what follows, we will leave the analysis in terms of these three operations and represent the result as a tuple $\kappa_{op}\langle a, m, e\rangle$ where op is an identifier for the operation, and a, m, e are the number operations: addition, multiplications and exponentiations respectively. For the purposes of this analysis, the costs of other operations (selecting a random value, for example) are ignored. . The algorithmic complexity, can be estimated from this tuple if needed. Forecasted running times can be obtained by taking the dot product with the observed milliseconds on the appropriate key size.

We illustrate estimating complexity and running time, using the example of the Pedersen commitment C=$g^a h^b (mod\ p)$. This operation would have 1 multiplication and 2 exponentiations, thus $\kappa_{PC}\langle 0,1,2\rangle$. The complexity of this operation is dominated by the cost of exponentiation having a cost of $O(n^2)$. The Pedersen

commitment includes a constant of 2, because two exponentiations are made. As is standard, this constant is not shown in Big O notation, but may (or may not) contribute a material cost an actual implementation with given system constraints. Continuing with the example, we can forecast estimated running time of the Pedersen commitment with a security parameter of 1024 as follows: $\kappa_{PC}\langle 0,1,2\rangle * \kappa_{1024}\langle 0.002,0.059,41.0\rangle = 82\,ms$. The calculation of running time below is omitted as it is highly dependent on hardware and software implementation, and subject to quickly becoming out of date.

Main differences between protocols

To determine the comparative computation cost between proposed protocols, we look at responsibilities of the user, the device, the issuer and the verifier for each of the setup issue and show phases (beyond what is required in the underlying credential scheme, itself).

		Alg0	Alg1
Setup	User	None	none
	Device	None	generate k1 generate k2
	Issuer	None	none
	Verifier	None	none
Issue	User	generate r_I	generate r_I
	Device	Extract b C_I= PC::Commit(b, r_I)	Extract b <R,P> = FE::Gen(b) P_e = E(P, k1, k2) CR_I= PC::Commit(R, r_I)
	Issuer	None	None
	Verifier	Not applicable	Not applicable
Show	User	generate r_S PoK1 { C_F, C_S *on* b_I } PoK2 {{$C_{F,i}$} = C_F, {$C_{S,i}$} = C_S} Pok3 |b|-t * (PK {$b_{I,i}$ == $b_{S,i}$ })	generate r_S PoK1 {CR_I, CR_S on R}
	Device	C_S = PC::Commit(b', r_S)	$P = D(P_e, k)$ $R' = FE::Rec(b', P)$ $CR_S = PC::Commit(R', r_S)$

	Issuer	not applicable	not applicable
	Verifier	PoK1, PoK2, PoK3 above	PoK1 above

Table 5. Protocol Comparison

The most important differences between the computation costs for the 2 algorithms are in a) the device's role in issue and show, and in b) the user and verifier's roles for proofs of knowledge in the show protocol. We consider each of these in turn.

Computation on the Device

The device in Alg_0 captures the biometric, performs template extraction accepts a random value from the user and creates a Pedersen commitment using the extracted template and the random value. Creating a commitment requires 1 modular multiplication and 2 modular exponentiations so $\kappa_{\mathrm{PC}} = \langle 0,1,2 \rangle$. The exponentiation dominates the operation with cost O(n^2).

As in Alg_0, the complexity of the device in Alg_1 cannot be lower than O(n^2) since Alg_1 also includes the Pedersen commitment. In addition, Alg_1 includes calculation of $<P_e, R>$ using $FE_{Ind} :: gen(\ldots)$ which adds the cost of the encryption $P_e = E(P, k)$, as well as the cost of error correction from FE_{Trad}. The block encryption algorithm [Gol04], uses the Naor-Reingold construction for PRFs [NR97] which requires k multiplications and 1 exponentiation, and is thus dominated by the exponentiation, having then the same order of magnitude as the Pedersen commitment.

Each of Alg_0, Alg_1 include image processing and template extraction which occurs on the device: these are common costs and are not included for the purposes of this analysis.

In summary, the devices in Alg_0, Alg_1 are all bounded $O(n^2)$ due to the exponentiation in Pedersen commitment. In implementation, there are constants to be aware of which will affect running time. Alg_1 for example includes a constant of 4 due to the calls to PRF used in encryption.

Proofs of Knowledge

As described above, Alg_1 has a more complex biometric sensing device. In exchange for this increase, however Alg_1 does not require 2 proofs of knowledge used by Alg_0.

The show protocol from in Alg_0 includes 3 interactive proofs of knowledge between the user and the service provider:

PoK1) proof that c_I and c_S are commitments on the same biometric b_I;

PoK2) proof that $\{C_{F,i}\}$ $\{C_{S,i}\}$ are bit commitments on the biometrics which underlie the commitments c_F and c_S respectively; and

PoK3) $|b| - t$ proofs of bit-wise equality between private b_I and b_S.

As summarized in Table 5, Alg_1 in requires only $PoK1$ avoiding both $PoK2$ and$PoK3$. The resulting saving is significant.

Note: A proof of knowledge may be repeated a number of times to reduce error, and increase certainty. For a Σ-protocol, for example, the error factor is given by $\frac{1}{2^k}$ where k is the number of repetitions. In our analysis below, we use k = 20 which results in a $\frac{1}{1{,}048{,}576}$ chance of error.

Requirement for PoK1

Both Alg_0 *and* Alg_1 need $PoK1$. As discussed in Section 2.2.5.2, this proof of knowledge convinces the verifier that two commitments are on the same value while keeping one of the commitments private. In Alg_0 the protocol is used to show that the public commitment C_F is on the same issue-time biometric as the private commitment C_I which been certified by the Issuer. In Alg_1 the same proof of knowledge is used to show that the private issue-time commitment CR_I and the public show-time commitment CR_S are both commitments on the biometrically derived authentication key R.

One execution of $PoK1$ incurs approximately $\kappa_{\mathrm{PK1}} = \langle l, 4l, l \rangle$ executions of the measured operations where l is the number of attributes supported by the digital credential. For the sake of illustration, consider that the Canadian e-Passport includes to the order of 15 fields. If these were the fields encoded into the digital credential, $PoK1$ would require $\kappa_{\mathrm{PoK1}}\langle 15{,}60{,}15 \rangle$ operations, a cost dominated by the

exponentiations. To execute the proof once, using our sample execution times would require $\kappa_{\text{POK1}}\langle 15,60,15\rangle \cdot \kappa_{1024}\langle 0.002,0.059,41.0\rangle = \kappa_{(\text{POK1})MS}\langle 0.036,3.599,738\rangle$. It must be remembered that to reduce error, the PoK should be repeated a number of times, for example 20 times, to reduce the chance of error to $\frac{1}{1,048,576}$, leading to a cost of approximately 20 repetitions * ¾ sec/repetition = 15 seconds.

Saved cost for PoK2 and PoK3

The remaining proofs of knowledge in Alg_0 are not needed in Alg_1. Alg_0 uses $PoK2$ to establish that two collections of bit commitments $\{C_{F,i}\}$ and $\{C_{S,i}\}$ are on the bits of b_I and b_S which are the values committed to by C_F and C_I. Proof PoK3 then chooses |b| - t bits in which the biometrics are equal and engages in a proof of knowledge for each.

In Alg_1 the fuzzy extractor has the responsibility of assessing whether the verification biometric and the enrollment biometric are within the required threshold for the given distance metric. This removes the need for the |b| - t proofs of bitwise equality in $PoK3$, and also removes the need for the intermediary setup done by $PoK2$.

Cost of PoK2

In general terms, Alg_0 performs a proof of knowledge which first creates and reveals a set of commitments $\{C_{F,i}\}$, and then proves that these commitments are on the bits which make up a value contained in another commitment C_F. In Alg_0, this process is performed twice: once for the bits of C_F and another time for the bits of C_S.

For each of C_F and C_I, $PoK2$ must: 1) create the collections of bit commitments; 2) Proof of knowledge that committed value is 0 or 1; 3) compute a verification relationship to establish correspondence on committed values. The requirement for this proof of knowledge is removed in Alg_1.

Cost of PoK3

The third proof of knowledge in Alg_0, $PoK3$ presents a series of Schnorr-like proofs of knowledge to prove bit equality without revealing the value of

the bits (See "Protocol 3" in [Ada11]). The proposed protocol completely does away for the need for this set of proofs. One round of the protocol execution requires that U make 3 additions, 5 multiplications, and 4 exponentiations and that the SP make 4 multiplications and 6 exponentiations for a total of $\kappa_{\text{PK3}} = \langle 3,9,10 \rangle$.

This protocol must be repeated for as many bits as must be shown equal for the given hamming distance. If we assume, for the sake of illustration, the size of a biometric template to be 2048 and supported hamming distance of 256, we then have a similarity threshold t of 1792: 1792 repetitions of $PoK3$ would be required to prove equivalence of b_I and b_S under the given Hamming Distance. This would result in $t * \kappa_{\text{POK3}} = 1792 * \kappa_{\text{POK3}}$ = 1792 * $\langle 3,9,10 \rangle$ = $\langle 5376,16128,17920 \rangle$. Using a biometric of size 160 bits, and a threshold of 20 bits, 140 proofs would be required, thus: 140 * $\langle 3,9,10 \rangle$ = <420, 1260, 1400>, to calculate total milliseconds: 420 * 0.002 + 1260 * 0.059 + 1400 * 41 = 0.84 ms + 74.34 ms + 57400 ms $\cong$ 57 seconds; repeating each proof of knowledge by a factor of k =20 results in a cost of 57 seconds * 20 $\cong$ 20 minutes. This cost is entirely avoided in the protocol proposed in this thesis.

In terms of computations for proofs of knowledge, Alg_1 may be preferable

5.5. Communication Cost

Similar to the discussion on computation cost, the differential in terms of communication centers on the proofs of knowledge during the show protocol. Since Alg_1 offers an improvement, doing away with the need for PoK2 and the |b| -t repetitions of PoK3.

5.6. Biometric Privacy

In Alg_0, the user receives the biometric b_I during the issue protocol and biometric b_S during the show protocol. Furthermore, b_I is stored on the user's device. In this situation, a malicious user or a compromised user device defeats biometric privacy entirely. While we note that the liveness assumption ensures that the com-

promised biometric cannot be used in a replay attack against either [Ada11] or this thesis, the leaked biometric could be used against other systems in multiple attacks and thus is a significant privacy and security concern.

In contrast, our fuzzy extractors based construction provides biometric privacy. In terms of biometric privacy Alg_1 offers clear advantages.

5.7. Storage

Both Alg_0 and Alg_1 require additional storage on the user's computer.

On the users device, Alg_0 stores the random string r and the biometric b_I on the user's computer. Alg_1 requires storage of r_I, CR_I; R_I, and P_e on the user's computer.

Alg_0 offers clear advantages over Alg_1 in terms of storage

5.8. Chapter Summary

In summary when compared to Alg_0, Alg_1 as described in this thesis presents benefits of biometric privacy, generality of biometric modalities and efficiency. Alg_0 on the other hand offers a benefits in terms of system maintenance, simplicity of biometric device, and required storage on the user's computer.

Chapter 6. Conclusions and Future Work

6.1. Conclusion

This work sought to improve on the feasibility result presented by Adams [Ada11], which we use as our baseline for comparison, and which we refer to as Alg_0. The algorithm proposed in this thesis, which we refer to as Alg_1, achieves notable simplification, increased generality and biometric privacy through the use of fuzzy extractors equipped with an indistinguishability adapter. Alg_0 however, features a simpler biometric device with easier maintenance, and less required storage on the user computer.

The benefits in terms of biometric privacy are due to the fact that in Alg_0 the actual biometric template is stored on the user's computer, while in Alg_1, a fuzzy extractor indistinguishability adapter allows identify verification without any biometric being stored. In terms of generality of biometric modalities, Alg_0 is constrained to the hamming distance metric, whereas Alg_1 uses fuzzy extractors and so supports set distance, vector distance, and edit distance resulting in a wider variety of supported biometric modalities. In terms of efficiency, Alg_0 conducts *n-t* Schnorr-like proofs of knowledge on the bits of the biometric template to prove that the issue-time biometric and show-time biometric are within the prescribed hamming distance of each other. Alg_1 does away with the need for these proofs of knowledge by moving the responsibility for similarity calculation to the fuzzy extractor.

While Alg_1 offers some benefits there are also associated costs. The biometric device of Alg_1 is more complex than that of Alg_0. Beyond template extraction, Alg_0 requires only the Pedersen Commitment algorithm. The biometric device of Alg_1 on the other hand, requires the Pedersen commitment algorithm, the fuzzy extractor indistinguishability adapter (which in turn needs IND-CCA2 encryption), and whichever traditional fuzzy extractor is selected. The symmetric keys and associated parameters for fuzzy extractors must also be configured onto the devices.

6.2. Future Work

The focus here has not been to apply the construct to a specific biometric modality. Work in this area can yield interesting results. A number of instantiations of Secure Sketches and Fuzzy Extractors exist on biometric modalities such as fingerprints and iris codes. It could be interesting to select such an application and examine the results of constructing the validating approach presented here. This could give an idea of the implementation challenges when targeting a current biometric device.

This work presents a general technique and an instantiation on Stefan Brands' Digital Credentials. It should also be possible to apply the technique presented here to Anonymous Credential by Camenisch Lysyanskaya. This can be the subject of follow-up work.

The approach presented here may offer benefit as it may form the base of a biometrically derived pseudonym system which can be plugged into generically other systems requiring non-transferable pseudonyms. This thesis, and ideas from [PM04] may provide a starting point for this research.

The algorithm presented here was to prove the feasibility of improving [Ada11] through the use of fuzzy extractors. It may be possible to streamline the proposed algorithm, perhaps optimizing sensor requirements, and minimizing the requirements for stored data on the user computer.

It may be of interest to gather the security models presented in the literature for credential systems and biometric privacy and study their relative strengths and weaknesses, and further assess key proposed algorithms against them.

It would be an interesting next step to look at enhancements and optimizations for the biometric sensor. One optimization would be to eliminate the need for the device to hold encryption keys.

This work has considered the use of fuzzy extractors for biometric key generation. Other techniques may also be used such as those of Tuyls and Monrose as presented in the literature review. It may be interesting to examine other approaches to biometric key generation.

References

[Ada11] Adams, Carlisle. "Achieving non-transferability in credential systems using hidden biometrics." *Security and Communication Networks* 4, no. 2 (2011): 195-206.

[Adl04] Adler, Andy. "Images can be regenerated from quantized biometric match score data." In *Electrical and Computer Engineering, 2004. Canadian Conference on*, vol. 1, pp. 469-472. IEEE, 2004.

[Adl05] Adler, Andy. "Vulnerabilities in biometric encryption systems." In *Audio-and Video-Based Biometric Person Authentication*, pp. 211-228. Springer Berlin/Heidelberg, 2005.

[BA11] M. Blanton and M. Aliasgari, On the (Non-)Reusability of Fuzzy Sketches and Extractors and Security in the Computational Setting, International Conference on Security and Cryptography (SECRYPT'11), Jul. 2011.

[Bel09] Belenkiy, Mira, Jan Camenisch, Melissa Chase, Markulf Kohlweiss, Anna Lysyanskaya, and Hovav Shacham. "Randomizable proofs and delegatable anonymous credentials." *Advances in Cryptology-CRYPTO 2009* (2009): 108-125

[BH09] Blanton, Marina, and William Hudelson. "Biometric-based non-transferable anonymous credentials." *Information and Communications Security* (2009): 165-180.

[BIOM] http://wwwbiometrics.gov accessed Sept. 29, 2012

[Bla83] Blahut, Richard E. *Theory and practice of error control codes*. Vol. 126. Reading (Ma) etc.: Addison-Wesley, 1983.

[Bleu98] Bleumer, Gerrit. "Biometric yet privacy protecting person authentication." In *Information Hiding*, pp. 99-110. Springer Berlin/Heidelberg, 1998.

[Boy04] Boyen, Xavier. "Reusable cryptographic fuzzy extractors." In *Proceedings of the 11th ACM conference on Computer and communications security*, pp. 82-91. ACM, 2004.

[Boy07] Boyen, Xavier. "Robust and Reusable Fuzzy Extractors." *Security with Noisy Data* (2007): 101-112.

[Bra00] Brands, Stefan A. *Rethinking public key infrastructures and digital certificates: building in privacy*. MIT Press, 2000.

[Bra02] Brands, Stefan. "A technical overview of digital credentials." *Available online, Feb* 20 (2002): 145-8. (credentica.com)

[Bri07] Bringer, Julien, Hervé Chabanne, Gérard Cohen, Bruno Kindarji, and Gilles Zémor. "Optimal iris fuzzy sketches." In *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*, pp. 1-6. IEEE, 2007.

[Bur99] Burmester, Mike, Yvo G. Desmedt, Toshiya Itoh, Kouichi Sakurai, and Hiroki Shizuya. "Divertible and subliminal-free zero-knowledge proofs for languages." *Journal of cryptology* 12, no. 3 (1999): 197-223.

[Can07] Canetti, Ran, Moses Samson Charikar, Sridhar Rajagopalan, Shanmugasundaram Ravikumar, Amit Sahai, and Andrew S. Tomkins. "Non-transferable anonymous credentials." U.S. Patent 7,222,362, issued May 22, 2007.

[Cha82] Chaum, David. "Blind signatures for untraceable payments." In *Advances in Cryptology: Proceedings of Crypto*, vol. 82, pp. 199-203. 1982.

[Cha85] Chaum, David. "Security without identification: Transaction systems to make big brother obsolete." *Communications of the ACM* 28, no. 10 (1985): 1030-1044.

[CL01] Camenisch, Jan, and Anna Lysyanskaya. "An efficient system for non-transferable anonymous credentials with optional anonymity revocation." *Advances in Cryptology—EUROCRYPT 2001* (2001): 93-118.

[CL06] Chase, Melissa, and Anna Lysyanskaya. "On signatures of knowledge." *Advances in Cryptology-CRYPTO 2006* (2006): 78-96.

[CEvdG88] David Chaum, Jan-Hendrik Evertse, and Jeroen van de Graaf. An improved protocol for demonstrating possession of discrete logarithms and some generalizations. In David Chaum and Wyn L. Price, editors, Advances in Cryptology — EUROCRYPT ’87, volume 304 of Lecture Notes in Computer Science, pages 127–141. Springer-Verlag, 1988.

[CE86] Chaum, David, and Jan-Hendrik Evertse. "A secure and privacy-protecting protocol for transmitting personal information between organizations." In *Advances in Cryptology—CRYPTO’86*, pp. 118-167. Springer Berlin/Heidelberg, 1987.

[CG08] Camenisch, Jan, and Thomas Groß. "Efficient attributes for anonymous credentials." In *Proceedings of the 15th ACM conference on Computer and communications security*, pp. 345-356. ACM, 2008.

[CKL03] Clancy, T. Charles, Negar Kiyavash, and Dennis J. Lin. "Secure smartcard-based fingerprint authentication." In *Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*, pp. 45-52. ACM, 2003.

[CL03] Camenisch, Jan, and Anna Lysyanskaya. "A signature scheme with efficient protocols." *Security in communication networks* (2003): 268-289.

[Che95] Chen, Lidong. "Access with pseudonyms." In *Cryptography: Policy and Algorithms*, pp. 232-243. Springer Berlin/Heidelberg, 1996.

[CP92] Chaum, David, and Torben Pedersen. "Wallet databases with observers." In *Advances in Cryptology—CRYPTO'92*, pp. 89-105. Springer Berlin/Heidelberg, 1993.

[CP07] Camenisch, Jan, and Birgit Pfitzmann. "Federated identity management." *Security, Privacy, and Trust in Modern Data Management* (2007): 213-238.

[CS97] Camenisch, Jan, and Markus Stadler. "Efficient group signature schemes for large groups." *Advances in Cryptology—CRYPTO'97* (1997): 410-424.

[CY08] Chen, Ning, and Zhiyuan Yan. "Complexity analysis of Reed-Solomon decoding over GF (2 m) without using syndromes." *EURASIP Journal on Wireless Communications and Networking* 2008 (2008): 16.

[DA05] Dodis, Yevgeniy, and Adam Smith. "Correcting errors without leaking partial information." In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pp. 654-663. ACM, 2005.

[Dam88] Damgård, Ivan. "Payment systems and credential mechanisms with provable security against abuse by individuals." In *Advances in Cryptology—CRYPTO'88*, pp. 328-335. Springer Berlin/Heidelberg, 1990.

[Dam99] Damgård, Ivan. "Commitment schemes and zero-knowledge protocols." *Lectures on Data Security* (1999): 63-86.

[DRS04] Dodis, Yevgeniy, Leonid Reyzin, and Adam Smith. "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data." In *Advances in cryptology-Eurocrypt 2004*, pp. 523-540. Springer Berlin/Heidelberg, 2004.

[DFM98] Davida, George I., Yair Frankel, and Brian J. Matt. "On enabling secure applications through off-line biometric identification." In *Security and Privacy, 1998. Proceedings. 1998 IEEE Symposium on*, pp. 148-157. IEEE, 1998.

[DY05] Dodis, Yevgeniy, and Aleksandr Yampolskiy. "A verifiable random function with short proofs and keys." *Public Key Cryptography-PKC 2005* (2005): 416-431.

[FCV2000] Biometric System Laboratory is active at the University of Bologna First International Competition for Fingerprint Verification Algorithms http://bias.csr.unibo.it/fvc2000/default.asp

[FS86] Fiat, Amos, and Adi Shamir. "How to prove yourself: practical solutions to identification and signature problems." In *Advances in Cryptology—Crypto'86*, pp. 186-194. Springer Berlin/Heidelberg, 1987.

[GMR85] Goldwasser, Shafi, Silvio Micali, and Charles Rackoff. "The knowledge complexity of interactive proof systems." *SIAM Journal on computing* 18, no. 1 (1989): 186-208.

[Gol00] Goldreich, Oded, *Foundations of Cryptography: Basic Tools, Volume 1*. Cambridge University Press, New York, NY, 2000

[Gol04] Goldreich, Oded. *Foundations of Cryptography: Volume 2, Basic Applications*. Vol. 2. Cambridge university press, 2004.

[HAD06] Hao, Feng, Ross Anderson, and John Daugman. "Combining crypto with biometrics effectively." *Computers, IEEE Transactions on* 55, no. 9 (2006): 1081-1088.

[Huf03] W. Cary Huffman . Fundamentals of Error-Correcting Codes Vera Pless, University of Illinois, Chicago ISBN:9780521782807, June 2003

[IM03] Impagliazzo, Russell, and Sara Miner More. "Anonymous credentials with biometrically-enforced non-transferability." In *Proceedings of the 2003 ACM workshop on Privacy in the electronic society*, pp. 60-71. ACM, 2003.

[JS02] Juels, Ari, and Madhu Sudan. "A fuzzy vault scheme." *Designs, Codes and Cryptography* 38, no. 2 (2006): 237-257.

[JW99] Juels, Ari, and Martin Wattenberg. "A fuzzy commitment scheme." In *Proceedings of the 6th ACM conference on Computer and communications security*, pp. 28-36. ACM, 1999.

[KAK96] Koc, C K., Tolga Acar, and Burton S. Kaliski Jr. "Analyzing and comparing Montgomery multiplication algorithms." *Micro, IEEE* 16, no. 3 (1996): 26-33.

[Koc94] Koç, Ç. K. "Montgomery reduction with even modulus." In *Computers and Digital Techniques, IEE Proceedings*-, vol. 141, no. 5, pp. 314-316. IET, 1994.

[LRSW99] Lysyanskaya, Anna, Ronald Rivest, Amit Sahai, and Stefan Wolf. "Pseudonym systems." In *Selected Areas in Cryptography*, pp. 184-199. Springer Berlin/Heidelberg, 2000.

[LT03] Linnartz, Jean-Paul, and Pim Tuyls. "New shielding functions to enhance privacy and prevent misuse of biometric templates." In *Audio-and Video-Based Biometric Person Authentication*, pp. 1059-1059. Springer Berlin/Heidelberg, 2003.

[Mao98] W. Mao and C. H. Lim, Cryptanalysis in prime order subgroups of Z_n^*. Advances in Cryptology ASIACRYPT '98 (K. Ohta and D. Pei, eds.), Lecture Notes in Computer Science, vol. 1514 Springer Verlag,1998 pp. 214-226

[MvOV97] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. Handbook of Applied Cryptography. CRC Press, New York, 1997.

[MRV99] Micali, Silvio, Michael Rabin, and Salil Vadhan. "Verifiable random functions." In *Foundations of Computer Science, 1999. 40th Annual Symposium on*, pp. 120-130. IEEE, 1999.

[NNJ08] Nagar, Abhishek, Karthik Nandakumar, and Anil K. Jain. "Securing fingerprint template: Fuzzy vault with minutiae descriptors." In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pp. 1-4. IEEE, 2008.

[NNJ12] Nagar, Abhishek, Karthik Nandakumar, and Anil K. Jain. "Multibiometric Cryptosystems Based on Feature-Level Fusion." *Information Forensics and Security, IEEE Transactions on* 7, no. 1 (2012): 255-268.

[NR97] Naor, Moni, and Omer Reingold. "Number-theoretic constructions of efficient pseudo-random functions." In *Foundations of Computer Science, 1997. Proceedings., 38th Annual Symposium on*, pp. 458-467. IEEE, 1997.

[Ped92] Pedersen, Torben. "Non-interactive and information-theoretic secure verifiable secret sharing." In *Advances in Cryptology—CRYPTO'91*, pp. 129-140. Springer Berlin/Heidelberg, 1992.

[PL11] www.primelife.eu website for Prime Life (part of European Union 7th Framework Project) accessed 12/21/2011

[PM04] Pashalidis, Andreas, and Chris Mitchell. "A security model for anonymous credential systems." *Information Security Management, Education and Privacy* (2004): 183-199.

[PV03] Persiano, Pino, and Ivan Visconti. "An anonymous credential system and a privacy-aware PKI." In *Information Security and Privacy*, pp. 27-38. Springer Berlin/Heidelberg, 2003

[RCB01] Ratha, Nalini K., Jonathan H. Connell, and Ruud M. Bolle. "Enhancing security and privacy in biometrics-based authentication systems." *IBM systems journal* 40, no. 3 (2001): 614-634.

[Sch91] Schnorr, Claus-Peter. "Efficient signature generation by smart cards." *Journal of cryptology* 4, no. 3 (1991): 161-174.

[SB07] Scheirer, Walter J., and Terrance E. Boult. "Cracking fuzzy vaults and biometric encryption." In *Biometrics Symposium, 2007*, pp. 1-6. IEEE, 2007.

[Sho97] Shoup, Victor. "Lower bounds for discrete logarithms and related problems." In *Advances in Cryptology—EUROCRYPT'97*, pp. 256-266. Springer Berlin/Heidelberg, 1997.

[Sho05] Victor Shoup. A Computational Introduction to Number Theory and Algebra. Cambridge University Press, 2005. Available from http://shoup.net.

[SKTP09] Simoens, Koen, Pim Tuyls, and Bart Preneel. "Privacy weaknesses in biometric sketches." In *Security and Privacy, 2009 30th IEEE Symposium on*, pp. 188-203. IEEE, 2009.

[Sou99] Soutar, Colin, Danny Roberge, Alex Stoianov, Rene Gilroy, and Bhagavatula Vijaya Kumar. "Biometric Encryption: enrollment and verification procedures." In *Aerospace/Defense Sensing and Controls*, pp. 24-35. International Society for Optics and Photonics, 1998.

[Tuy05] Tuyls, Pim, Anton Akkermans, Tom Kevenaar, Geert-Jan Schrijen, Asker Bazen, and Raimond Veldhuis. "Practical biometric authentication with template protection." In *Audio-and Video-Based Biometric Person Authentication*, pp. 1-53. Springer Berlin/Heidelberg, 2005.

[UJ06] Uludag, Umut, and Anil Jain. "Securing fingerprint template: Fuzzy vault with helper data." In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*, pp. 163-163. IEEE, 2006.

[US7711152] Davida, George I., and Yair Frankel. "System and method for authenticated and privacy preserving biometric identification systems." U.S. Patent 7,711,152, issued May 4, 2010.

[Ver01] Verheul, Eric. "Self-blindable credential certificates from the Weil pairing." *Advances in cryptology—ASIACRYPT 2001* (2001): 533-551.

[Wu11] Wu, Lifang, Peng Xiao, Siyuan Jiang, and Xin Yang. "A fuzzy vault scheme for feature fusion." *Biometric Recognition* (2011): 237-243.

[YBD05] Yang, Yanjiang, Feng Bao, and Robert Deng. "Security analysis and fix of an anonymous credential system." In Information Security and Privacy, pp. 169-182. Springer Berlin/Heidelberg, 2005.