Credential Design in Attribute-Based Identity Management

Gergely Alpár[1]
Radboud University Nijmegen, ICIS Digital Security and TNO Security, The Netherlands
✉ gergely@cs.ru.nl

Bart Jacobs
Radboud University Nijmegen, ICIS Digital Security, The Netherlands
✉ bart@cs.ru.nl

Abstract Attribute-based credentials are cryptographically secured carriers of properties that hold for a particular individual. They are the basic building blocks of many upcoming privacy-enhancing technologies and user-centric identity management systems. There are a number of limitations and requirements besides security and privacy, such as usability and efficiency, that have to be taken into account when designing specific credentials in practice.

This paper elaborates several realistic on-line and off-line use cases in attribute-based identity management; moreover, it identifies and analyses some of the design issues that require a decision or solution. It provides the most important credential design principles and also shows how setting up an attribute-based credential system formalises identity relationships in society.

Keywords attribute-based credential, smart card, pilot, identity management, identity card

Introduction

Authorisation requires authentication: before letting someone do or use something, it must be clear that this someone is actually allowed to do so. Traditionally, authentication is understood as proof of identity, for instance, by means of a password or an identity document. But precisely identifying people, using uniquely identifying numbers and names—such as a social security number (SSN), credit card or bank account number—is often an overkill. In many situations it suffices to know some attribute (property) of a person in order to authorise a transaction. If a hairdresser offers a cheap haircut to students, it is not necessary, or even desirable, that the hairdresser learns a (uniquely identifying) student number as part of the proof of 'studentship'. Similarly, buying an alcoholic drink only requires a proof that the buyer is above a certain age limit (16, 18, or 21). Attribute-based authentication aims to provide a mechanism for precisely doing this: allowing transactions on the basis of those attributes which are required for the transaction. The main advantages are:

- it is privacy-friendly, in the sense that it is based on the idea of data minimisation and that it provides unlinkability among user transactions;
- it offers protection against identity fraud: if one's identity is not involved in a transaction, it cannot be stolen;
- it provides a new, more flexible approach in identity management and authentication, in particular, an approach that is based on attributes instead of unique identities.

Attribute-based authentication is not new. Attribute certificates [10] were defined in the X.509 stack over a decade ago. They enable authentication that does not require identification; *e.g.*, role-

[1] Supported by the research program Sentinels as project 'Mobile IDM' (10522).

based access or proof of membership. However, they are (1) linkable (each transaction is linked to the same public key) and (2) transferable (delegateable). Attributes in the context of attribute-based credentials and in this paper are different; they provide security, unlinkability, and untransferability simultaneously (see details about security and privacy properties in Section 2). Cryptographic techniques that enable secure and privacy-friendly attribute-based authentication have also been around for more than a decade, see [4, 7, 8, 14]. But what is new is that the latest generation of smart cards is powerful enough to perform the required (non-trivial) cryptographic operations in an adequately efficient manner. Hence only now we see efforts to actually deploy attributes in practice. This paper is based on the experiences in one such deployment in the course of a pilot project, namely the IRMA project[2] in The Netherlands. It relies on the Idemix technology [13] and uses personal smart cards as carriers of credentials and attributes—see the next section for more details. Getting attribute technology up-and-running brings us into largely unexplored territory that poses a multitude of technical and organisational challenges. But also it leads to new (research) questions and forces us to think deeper and more systematically about the technology and its implications. As its main contribution, the current paper explores these matters. It concentrates on the issues that arise regarding the organisation of *multiple* attributes and of the dependencies between them, and on the decisions that need to be made to make these cryptographic techniques and their implementation practical while preserving their advanced properties. Many other interesting topics are out of scope, like the underlying cryptography [7, 8], the smart card technicalities, or a detailed security analysis.

To the best of our knowledge, there are two other pilot projects in the context of attribute-based credentials. Both of them are carried out by the EU-sponsored ABC4Trust [6]. The Swedish pilot [3] gives anonymous access for elementary school pupils to on-line resources (*e.g.*, chat room), while the Greek pilot [1] enables university students to evaluate lectures anonymously. In both cases eligibility and privacy are of primary importance. Although our pilot uses the same underlying technology, the objective of our research is more general as we investigate a *broad variety* of attributes and applications. The kind of challenges investigated in this paper do not appear in these ABC4Trust pilots since each focusses on a single context.

One may view an individual's identity as the collection of all attributes that hold for him/her. We can imagine that using a personal smart card, people manage dozens of attributes for various authentication goals, determined by the organisations that they interact with. Given that there are many dependencies between all these attributes, the question of how to organise them in a logical/coherent and intuitive manner is non-trivial and not free from politics (information is power). This is the main topic of this paper. We make the various issues explicit that we came across in the context of our pilot project and explain the choices we have made. This is certainly relevant beyond this particular project.

[2] See `irmacard.org`, where IRMA is an abbreviation for: I Reveal My Attributes.

2 Technical Background

Technically, digital credentials, containing attributes, form a coherent unit. In our discussion, however, attributes play a more important role conceptually. We can simplify it and say that credentials are issued and attributes are shown. In this section we describe some abstract technical details of the technology, the participants, and our implementation.

Attributes In the current context an attribute is some property of or a piece of data about a person that some party (most often some authority) attested to. We briefly elaborate.

Some attributes are identifying and some are non-identifying, i.e., some attributes hold for a single individual (in a particular context) whereas other attributes hold for many people. For instance, the attribute 'male' is in general not identifying, but the attribute 'bank account is …' identifies the (sole) holder of the account. The phrase 'anonymous credential system' is often used in the literature for systems like U-Prove and Idemix, but in the current context attributes need not be anonymous (non-identifying).

What is important is that for a particular individual an attribute either holds or not, at a particular point in time. So, for instance, the attribute 'under 18' may hold now for my son, but may no longer hold next year: the validity of personal attributes is time-dependent.

In this context it is assumed that there is some authority that can decide whether attribute A holds for person P at time t, and that this authority is willing to provide this attribute to P with its digital signature. For instance, my bank can digitally sign the statement what my bank account is at this moment, and provide the result in a credential to me. In some cases it is obvious for a given attribute which authority is in the best position to issue it in a credential: my bank is most authoritative when it comes to my bank account. But in other cases there may be multiple authorities. An example might be my address attribute, which can be provided either by the municipal authorities or, for example, by the postal service. We return to this matter later on.

Part of such a digital signature on an attribute is usually an expiration date. The expiration date may be necessary because the attribute may no longer hold after some time (like for 'under 18'). But expiration may also be used to limit the usage period of an attribute. For instance, the signature on the attribute containing my home address may expire after a year in order to ensure that it is reasonably fresh (and thus accurate).

Credentials A credential, in the context of this paper, is a cryptographic container for attributes. It is digitally signed by a trusted party, the issuer (see more details below). This digital signature provides certainty about the validity of the attributes within the credential and also about the fact that they have not been changed since issuance. Furthermore, credentials hide the attributes; so, seeing a credential, one cannot deduce any information about the attribute values in it. The structure of a credential (i.e. the semantics and types of attributes in it), unlike its content, is public. This enables a card holder

and a verifier to select the appropriate credential(s) for a certain scenario (see example scenarios in Section 3).

Anonymous credentials were already proposed over 25 years ago by David Chaum [9]. They enable individuals to authenticate without identification and to perform unlinkable actions. Stefan Brands [4] suggested practical and efficient cryptographic protocols for implementing digital credentials that include multiple attributes. Recently this notion was renamed to attribute-based credentials (ABCs). An ABC may contain several attributes that can be shown independently of one another. Brands' protocols belong now to Microsoft's U-Prove technology [5, 12] and replace Microsoft's earlier Windows CardSpace approach. Jan Camenisch and Anna Lysyanskaya [7, 8] proposed another technology for attribute-based credentials, using zero-knowledge proofs. These schemes are now collected in IBM's Idemix [13]. ABC4Trust [6] aims to create a common architecture for these technologies.

Our pilot project uses an efficient smart card implementation of Idemix; but conceptually it could also use the U-Prove technology. A smart card may contain dozens of credentials, each with multiple attributes. In a particular attribute-based authentication proof, any subset of attributes in a single credential may be revealed, without revealing the remaining attributes. This is called selective disclosure. Also, several attributes from different credentials may be revealed, like 'over 21' and 'Student'.

Within the context of this project at most four attributes are grouped together in a credential, see Figure 1. The number four is chosen pragmatically, mainly for implementation reasons, but other reasons turn out to confirm this choice. On the one hand, having many attributes in one credential means that if only one attribute is revealed, all the others remain hidden. Hiding more attributes requires more time, and thus reduces the performance. On the other hand, the number four seems to be reasonable to form a coherent set of attributes, issued jointly by a single authority.

All credentials are required to contain two additional basic attributes. First, an expiry date has to be determined at issuance, and it is included as an attribute applying to the whole credential. When the credential is verified, the expiry date can be revealed to confirm validity. Second, each user has a master secret key, stored in the smart card's secure storage, which is also incorporated—technically, like an attribute—in all credentials.

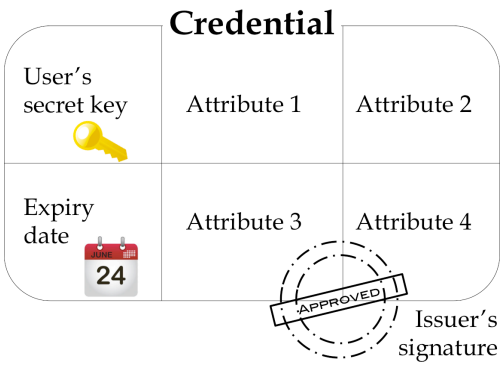


Figure 1: The structure of an attribute-based credential with two reserved and four 'free' attributes.

Roles In attribute-based identity management we distinguish the following roles.

1. **Users** are people who own a smart card that holds valid attributes; validity means that the attributes on the card are valid for the card holder (and are not expired).
2. **Issuers** are the authorities that sign credentials with attributes and provide them to Users. For instance, citizen registration authorities are the obvious issuers of 'over 18' attributes (and of many other attributes as well) and banks are authoritative issuers of bank account number attributes.
3. **Verifiers** (also called **relying parties**) are the parties that verify a subset of the available attributes on a card in order to authorise a transaction. An example verifier is a website that wants to verify the attribute 'over 18' before it allows me to view a certain video online.
4. The **scheme manager** is an independent, non-profit organisation that sets the rules for the different parties (users, issuers and verifiers) and is responsible for the software and smart card management. Of course, these roles can be split up and assigned to different organisations, but that is not so relevant for the current discussion.

Security and privacy properties Attribute-based credentials are assumed to provide the following security properties. (1) The issuer's digital signature ensures authenticity: the credential originates from the issuer, and this issuer assures that the attributes hold for the person. (2) This signature also guarantees integrity: the attributes contained in the credential have not been changed since they have been issued. (3) A credential is non-transferable as it is bound to the card of the person involved in the issuing protocol.

Furthermore, an attribute-based credential protects the privacy of its owner by the following cryptographic properties. (a) A credential hides its content, so it does not reveal the attributes that it contains. (b) Issuer unlinkability assures that any information gathered during issuing cannot be used to link the credential when it is shown. (c) Multi-show unlinkability guarantees that when a credential is shown multiple times, these sessions cannot be linked. The privacy of users is protected by both of these unlinkability properties even if the credential issuer and all verifiers collude.

Implementation used in this pilot In this paper we rely on technical assumptions from the pilot project that we are working on. We make these assumptions explicit.

1. The smart cards are MULTOS cards, because they provide relatively easy access to cryptographic primitives. The cards communicate via a wireless interface. Preferably, card readers are used that have a (secure) PIN pad. In the (near) future, widely employed card readers will probably be smart phones and tablets with their NFC interface.
2. Attributes are stored on smart cards in credentials; each credential can store up to four attributes, which are collectively issued (and signed) by one issuer. Hence, one design criterium for the contents of credentials is that all the attributes involved should fall under the responsibility of a single issuer.
3. Selective disclosure is an essential functionality. The verification of one or more attributes from the same credential can be done rather efficiently, taking on average in the order of one second[3]. Verification of multiple attributes from multiple credentials is also possible (within one session), but then the verification times add up, proportional to the number of credentials.
4. Issuing takes place per credential (and not per attribute) and is rather slow: in the order of 3 to 4 seconds. Typically, issuing is done either during a physical session (e.g., at the town hall) or online at a device that the user trusts (e.g., a personal tablet or a home PC).

   As a result, attributes are appropriate for rather static scenarios, and not for dynamic scenarios, such as an electronic purse, where the monetary value on the card is stored as an attribute: spending money would involve both verification (of the old amount, before paying) and re-issuing (the new amount, after paying). This is simply too slow with the current smart card technology.
5. Users have a 'card management' environment at their PC or other device, in which they have read/delete/update access to all the data on the card. Within this environment they can see dependencies (in tree-form, like in Figure 2) and inspect access logs. Furthermore, users can delete credentials or initiate to update them.
6. The whole process in relation to attributes and credentials takes place using open standards (and to a large extent even via open source software). This means that, in principle, every organisation or individual can use the same card for their own purpose, by issuing and verifying their own attributes. However, the scheme manager controls access to the cards (see also in Section 5). This happens by special certificates that terminals need to have before cards are willing to communicate with them. The role of the scheme manager enforces a certain level of consistency among issuers and verifiers and (thereby) protects the card holders.

[3] This one second is good enough for verifications online or offline, say in a shop, but too slow for entrance control like in public transport; in such cases the required maximal transaction time is typically 0.3 second.

3 Use Cases

This section gives an informal description of some of the use cases that we foresee for attribute-based authentication. As the current discussion considers attributes of a wide variety, we let attributes be non-identifying as well as identifying. We do not address however the problem of attribute semantics or anonymity sets in different scopes. While ABCs were originally devised for anonymous applications, we are convinced that they provide many more usage and application opportunities with (partly) identifying attributes. The use cases described shortly below form the basis for some further discussion of issues analysed in Section 5.

Age bounds The attribute that is most needed now is probably the minimal-age attribute, like 'over 18'. It will be useful for many online and offline transactions, such as buying/playing (violent) games, for alcoholic drinks, cigarets, (certain) movies or books, online gambling, etc. Analogously, one may form maximal-age attributes, like 'under 15'. They may be used to regulate access to certain chat rooms which are set up exclusively for minors.

Within the Idemix context there are 'interval proofs' which make it possible to derive these minimal- and maximal-age attributes from the date of birth. Such proofs are computationally rather expensive and are (currently) not included in this project. Instead, minimal-age and maximal-age credentials are foreseen consisting of the form:

minimal junior
≥ 12
≥ 16
≥ 18
≥ 21

minimal senior
≥ 60
≥ 65
≥ 70
≥ 75

maximal junior
< 12
< 16
< 18
< 21

The most authoritative issuers for such credentials are local or national authorities, using their citizen registration database.

Citizen Identity Your identity as citizen may be organised in three credentials:

name
family name
first name
full first names
initials

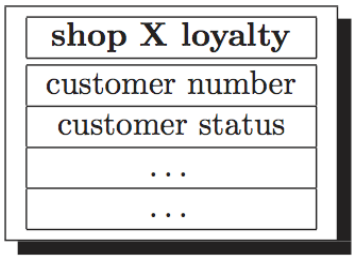
identity
social security nr.
date of birth
place of birth
gender

address
country
city
street + number
postal code

As before, public authorities are the most authoritative source to issue such credentials. Recall that each of these attributes can be used separately in authentication. But also combinations of these (and other) attributes are possible.

Loyalty Cards and Pseudonyms Shops, or other commercial organisations such as airlines, like to build a relationship with their customers using loyalty cards, giving them selected benefits when they

have accumulated enough loyalty points. Applying such cards, these shops can keep track of who purchases what and this allows them to build up detailed profiles of their customers. In practice, each chain of shops issues its own (virtual) loyalty card. This is no longer needed with an open card, since each chain can add its own loyalty credential to it.

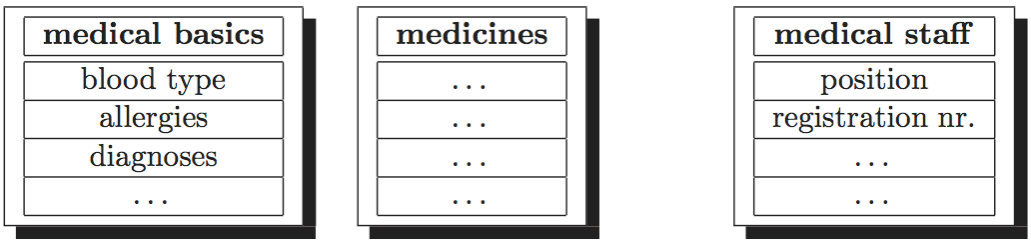


The customer number in the credential acts as a key for a database entry in the back office that contains the actual purchase history of the customer (card holder). On the basis of this history, a customer may reach a certain status, like bronze/silver/gold. In each shopping situation the customer may be offered the option to buy anonymously, using only the status attribute to get certain benefits, or to buy non-anonymously using also the customer number. Only in the latter case, the purchase is added to the personal history (in the back office) and contributes to the status build-up. The remaining two attributes, written as '…', are left open and can be used for other customer relationship management (CRM) purposes. They can also be left empty (blank).

A card holder may use his/her card with this credential offline, in a 'brick and mortar' shop. But it can also be used online, to purchase something, or to access an overview of the card holder's purchase history and, possibly, to update the status attributes. For these purposes, the loyalty number attribute is sufficient as authentication. Of course name & gender are nice to have for communication purposes, but they need not be the real ones. An address credential may be required in case of delivery. It can be verified per transaction, and need not be stored centrally.

Such customer numbers in credentials may thus be used as pseudonyms, one for each commercial relationship (with shops X, Y, Z, etc.). There is a potential privacy risk when many commercial organisations decide to cooperate and use one number for all of them. In this way they can profile customers across different organisations, a bit like it is done now via third party cookies or device fingerprinting. Such broad commercial use of a single pseudonym, possibly at a national level, may be forbidden by the scheme manager and/or by the relevant data protection authority.

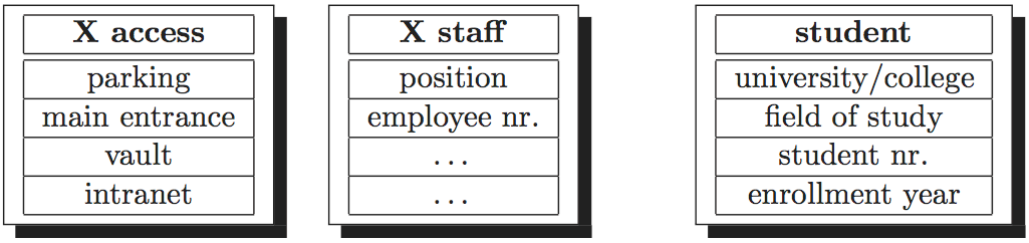
Medical information In a medical context one can envisage attributes for patients and for medical staff. Patients can carry for instance credentials with attributes containing essential personal medical information in a micro-dossier, see the first two credentials below. Medical staff can use credentials that describe their medical role and access rights to patient files, as suggested in:



The first two credentials may be issued by health authorities (hospitals, or even general practitioners). They are useful in medical emergency situations, like after an accident. The last credential falls under the responsibility of health staff registration authorities. The 'position' attribute typically determines access right to medical records, such as: doctors may both read and write, but nurses may only read. For accountability, the registration number should be used in each such transaction in order to monitor who accesses which file.

At this stage, it becomes clear that designing the content of credentials is not entirely trivial, and requires knowledge of the relevant domain of use. Another thing to note is that the names of the card holder are not included in these credentials. For now it suffices to say that the name occurs in a Name credential, so there is no need to repeat it. But this 'overlap' matter will be discussed further below.

Access control and role/claim-based access control Within one company/organisation X, a credential can be designed for specific access rights, roles, positions, etc., as suggested in:



Issuing a mobile phone number credential So far we have concentrated mostly on the contents of credentials. We now look at how the issuing of credentials might work. Suppose you wish to obtain a credential containing your mobile phone number. The obvious issuer is your mobile network operator (MNO). The issuing procedure might work via the following steps.

1. You go to the website of the MNO, using https, and prove using your IRMA card your name and date of birth.
2. The MNO looks in its database if there is a contract with this name and date of birth[4]; if not, it aborts; if so, it sends a one-time code over SMS to the (mobile) phone number associated with this contract.
3. Upon receiving this one-time code, you feed it back into the website (within the same https session).
4. The MNO now issues the credential containing your phone number, possibly together with some other attributes, to your card.

What is interesting about this protocol is that it involves authentication that uses both existing credentials and an out-of-band channel. The use of existing credentials leads to dependencies among credentials, as described in Section 4.

[4] In many countries, before obtaining a mobile phone subscription, a copy of an identity document must be handed over; this is assumed here.

Festival ticket We conclude this list of use cases with a non-standard application of attributes, in order to suggest the great variety and breadth of possible usage scenarios. If you wish to get a ticket online for a pop concert or other festivals, you often need to fill out long forms requiring personal information. The main purpose—apart from profiling—seems to be to prevent transfer of tickets. One may also provide such a ticket in electronic form, after payment, as a credential for the festival at hand, containing for instance: the festival name & date, a ticket number, any additional pre-paid consumptions, etc. Upon entering the festival terrain, the presence of a valid ticket on a card can be checked (and consumption vouchers can be handed over). The next day the ticket/credential is unusable, and can be removed from the card (by the card owner).

4 An Example Credential Tree

As we saw in Figure 1, credentials are containers of attributes signed by an authoritative issuer. An issuing procedure requires some sort of authentication to prove that a specific card is entitled to hold a credential. This authentication can include the verification of already existing credentials on the card. On the one hand, so-called root credentials do not rely on other credentials on the card. They require only out-of-band authentication. Dependent credentials, on the other hand, are issued only after verifying at least one other existing credential on the card. Technically, it is essential that the verification and the issuance happen in the same secure session.

Figure 2 shows a dependency graph, a possible arrangement of digital credentials logically residing on a card in the IRMA project. In this example, there are two root credentials. An Academia credential represents the card holder's identity in the national education system. A Citizen root credential can be used by a broader audience in a broader context. These root credentials can be issued after a personal, face-to-face identification accompanied by a physical identity document authentication.

A Student credential, for instance, relies only on the Academia root. After a student proves that he or she has such a root credential with the appropriate attributes of Organisation and unique student identifier (SID), the organisation can look up all relevant personal data in its database and issue the Student credential. Note that this issuing procedure requires identification since a Student credential is bound to a specific person. A university's Library credential can be issued similarly relying on an already existing Student credential. It can depend on policies, defined by the Scheme manager, which attributes a particular issuer is eligible to verify in relation to issuing a particular type of credential.

Issuance therefore often requires verification of credentials from the card, not only an out-of-band authentication. The simplest case is when only one credential is verified. But authentication can include multiple credentials residing in different parts of the dependency graph. Business scenarios, involving legal obligations, often require credentials from the citizen 'tree'—not only from the one that provides discount for the customer. A festival, for instance, may offer cheaper tickets for students (academia) while requiring certain minimum age (citizen) to give a voucher for alcoholic drinks.

We foresee that the scheme manager decides in a contract with each Issuer what the dependencies and (out-of-bound) authentication methods are (required for issuing). These matters will then be made public, so that others (esp. Verifiers) know what they can/cannot rely on.

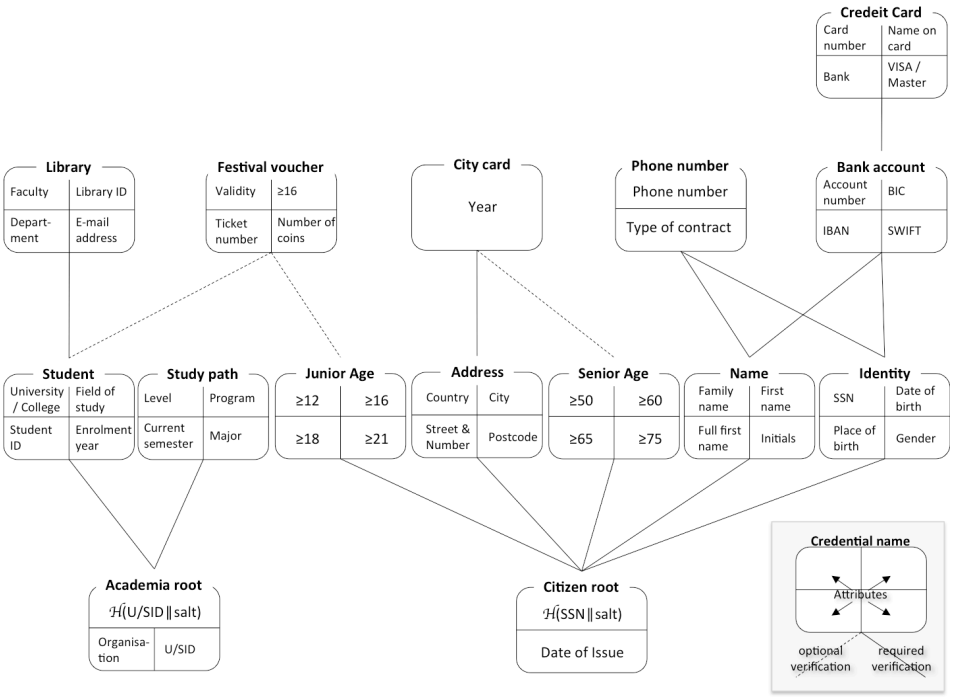


Figure 2: An example for credentials and dependencies.

5 Problems and Decisions

This section discusses several issues that we came across in setting up the IRMA pilot project. Although we recognise the importance of cost and liability in deploying a new technology, these considerations are out of scope in this study. In the decisions we took the main motivations were: simplicity of the set-up, intuitiveness of usage, protection of privacy, and security.

Outside of a card In online usage the outside of a card is irrelevant for the issuer or the verifier. The only practical requirement is that the card owner should recognise his/her own card (to prevent confusion). In offline scenarios, however, the verifier should be able to check that the person presenting a card is the card holder. This is done via two mechanisms:

- on the front of the card there is a picture of the cardholder—and nothing else;
- the verification of many attributes is only possible after a PIN is entered; as a result, if someone else wishes to use your card, you should also give your PIN. This works as hindrance.

At the back of a card there is (general) information about how lost cards can be returned. Additionally, there is a card-specific number. It can be used to look up the owner of a lost-and-returned

card. The card number is a dangerous addition that could make it possible to trace cards. Therefore, the card number is used only externally, and not internally, in the chip.

Restricting relying parties One serious challenge is how to make sure that a verifier (relying party) does what it promises: if a web shop says it only needs to see my 'over 18' attribute, how do I know that it does not read all other attributes as well? There are several possible approaches.

1. A purely **legal** one: let verifiers sign a contract with the scheme manager in which they commit themselves to behave as they promise.
2. Add a posteriori **monitoring**: make sure that the card logs all transactions, and take (legal) action if a verifier reads too much.
3. Add a priori **technical restrictions**: verifiers obtain a certificate from the scheme manager that will be checked by the card and that contains the attributes that the verifier may read; the card is programmed in such a way that it only reveals the attributes that are listed in the certificate.

The first two options provide no protection against rogue verifiers, operating outside the span of control of the scheme manager. The last solution is therefore the most secure one, but also the most inflexible and complicated one, since it requires an elaborate certificate management policy. It is the preferred solution within the IRMA project (although it is not yet implemented).

With this third solution in place, protecting card reading by a PIN is less urgent—increasing user convenience. For instance, it is not wise to protect medical emergency data by a PIN, since the card holder may not be able to provide it when needed most. But by providing only to medical emergency services a certificate to read the medical data, the privacy risks are reduced.

An alternative solution is to use designated proofs [2] in which the prover can control which verifier can receive particular attributes. Selective disclosure (see in Section 2) enables to restrict which attributes are revealed, while designation enables to restrict which verifiers can receive those attributes. When this technique is applied, verifiers are required to have secret keys for being able to compute those attributes. As this technology entails an additional infrastructure for designation keys (or certificates) and is still in its infancy, we do not use it at this stage of the IRMA project.

PIN use and card management In general, access to a card can be protected by a PIN. This is used to ensure:

- **confidentiality**: to prevent unauthorised reading of private data, for instance, after a card loss; the use of certificates (as discussed above) restricts this risk to some extent, but does not remove it;
- **user consent**: to make sure that a card is only used when the card holder agrees;
- **authentication**: the card is only usable by the card owner; in particular, someone else who obtains/finds a card cannot use it.

It is clear that the addition of new credentials to a card should be protected by a PIN, to guarantee consent & authentication. But when should revealing of attributes be protected by a PIN?

You may think of the fairly innocuous ‘over 18’ attribute. But it should not be possibly that my little nephew temporarily borrows my card to do/obtain ‘over 18’ stuff online. Hence the the age credential should be PIN-protected. Attributes that give access to a parking or open an entrance are typically not PIN-protected, except for high-security facilities.

If some credentials require PIN-protection and others do not, the question arises: who decides about this? Of course it can be left to the card reader or the user to set PIN-protection, but probably following some general policy is better. This policy should be set in general terms by the scheme manager, and elaborated in detail with each credential issuer.

Card hand-over A User obtains a card during a face-to-face protocol, called card hand-over. It involves verification of the (external) photo, PIN setting by the new card owner, and issuance of a number of root credentials. In the database of the scheme manager an entry will be maintained involving the external card number, contact details of the card owner, and a timestamp recording the hand-over.

Expiry and revocation In the current stage of the pilot project revocation will not be implemented although in a large-scale project this functionality is essential. Recent developments [11] show that privacy-friendly revocation techniques are reaching performance figures that make addition of revocation possible at some later stage in the project. Expiry data in credentials, see Figure 1, put some limit on the usability of credentials after a card loss. Additionally, some identifying attributes, containing for instance a registration number, can be blacklisted on the basis of their content.

Attribute duplication in the tree? In the credential examples in Section 3 we have seen that a card holder’s name occurs in the Name credential (obviously!), but not in a medical staff or employee credential. This may look unexpected at first. In principle, there could be multiple name attributes, issued by different parties (like local authorities, or Facebook; see below). Similarly, multiple accounts at different banks or different phone numbers can be issued in separate credentials. It is the role of the scheme manager to decide which organisations are authoritative about a type of credential. Verifiers can then decide which issuer they wish to trust for having attested to certain attributes. However, we propose as few attributes to be issued by multiple issuers as possible for simplicity and efficiency. In fact, so far we are excluding any duplication of attributes (same content, different issuers).

Facebook – root credential or not? In what follows, we take Facebook as example in considerations that apply to many other, similar organisations. If you sign up for Facebook, you choose the name that you like (within certain technical/decency limits). Facebook has a Real Name Policy, but it has no way of checking that the name you provide is your real one. Many people like to use a pseudonym on Facebook and currently this is possible.

Now suppose Facebook wishes to join the project at hand and use smart card based credentials for authentication. The credential only needs to contain Facebook’s user ID. An interesting question is: should this be a root credential or not? This technical question has wide societal relevance.

1. Facebook probably does not want to have a root credential: it likes to first verify the (real) name on the card (and probably more attributes), before issuing its own credential. In this way Facebook can enforce its Real Name Policy.
2. People who don't wish to use their real name on Facebook expect Facebook's credential to be root, not depending on any other.

There will be many other organisations like Facebook who are interested in issuing and using their own credentials if they can be based on other (reliable) attributes: probably Skype, but possibly also your favourite book store chain. Should the scheme manager allow this, and on which grounds? These decisions are political in nature, and they involve the identity fabric of our society and also considerable commercial interest. For this reason, we firmly believe that the scheme manager should be set-up and run as an independent non-profit and potentially distributed organisation.

(Within the pilot phase Facebook is not involved, but a similar issue has come up; we decided in favour of a root credential, thus preventing dependencies and verifications of other attributes.)

Omitted functionalities In this project, we deployed an efficient implementation of the basics of Idemix. This attribute-based credential technology provides several advanced features that we did not include in this pilot for usability and/or for efficiency reasons. Nevertheless, future use cases and developments may require these functionalities.

- Construction of logical AND / OR zero-knowledge proofs. Proofs about attributes, provided by the card for a verifier, can be combined into one proof by the conjunction (AND) and disjunction (OR) operations.
- Combined proofs using users' master keys. A master secret key must be generated and stored on a card which never leaves it. This key, used in each credential, can then be used to construct a single proof about attributes in different credentials on that card. Applying a similar method, this key can be used to bind verification of existing credentials and issuing of a new one on the card.
  Although this feature provides high security assurance, we chose to use for the time being independent proofs within a previously established single secure session.
- Inequality and interval proofs about attributes. Using inequality and interval proofs, a user can demonstrate properties of attributes (see an example at the Age credential in Section 3). Furthermore, an identifier attribute can be demonstrated to be on a membership list without disclosing the identifier.

In spite of these omitted functionalities all privacy and security properties (see in Section 2) of the Idemix system are incorporated.

6 Conclusion

In this paper we described the relevance and challenges of credential design in attribute-based identity management. Several use cases demonstrated the breadth of possible applications on a smart

card that supports attribute-based credentials. The main reason for this diversity is that attributes, issued by the most authoritative organisations, can be disclosed independently. Therefore, verifiers learn all relevant information to authenticate and authorise users but nothing more, thus contributing to data minimisation.

Recommendations We conclude the paper with six principles for credential design in the context of attribute-based credentials.

1. Attributes in one credential form a coherent set.
2. Each attribute in one credential falls under the responsibiility of a single most authoritative issuer.
3. Attribute duplication (same content, multiple issuers) is avoided.
4. Verifiers can read only a limited, predefined set of attributes.
5. Credential dependencies are public.
6. An independent non-profit scheme manager should decide about such dependencies.

References

[1] Joerg Abendroth, Vasiliki Liagkou, Apostolis Pyrgelis, Christoforos Raptopoulos, Ahmad Sabouri, Eva Schlehahn, Yannis Stamatiou, and Harald Zwingelberg. D7.1 Application Description for Students. Technical report, ABC4Trust, 2012.

[2] Gergely Alpár, Lejla Batina, and Wouter Lueks. Designated Attribute-Based Proofs for RFID Applications. In RFID Security and Privacy – RFIDsec 2012. LNCS, 2012.

[3] Souheil Bcheri, Norbert Goetze, Monika Orski, and Harald Zwingelberg. D6.1 Application Description for the School Deployment. Technical report, ABC4Trust, 2012.

[4] Stefan A. Brands. Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy. MIT Press, Cambridge, MA, USA, 2000.

[5] James Brown, Phil Stradling, and Craig H. Wittenberg. U-Prove CTP R2 Whitepaper. Technical report, Microsoft Corporation, February 2011.

[6] Jan Camenisch, Ioannis Krontiris, Anja Lehmann, Gregory Neven, Christian Paquin, Kai Rannenberg, and Harald Zwingelberg. D2.1 Architecture for Attribute-based Credential Technologies. Technical report, ABC4Trust, 2011.

[7] Jan Camenisch and Anna Lysyanskaya. An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation. In Birgit Pfitzmann, editor, Advances in Cryptology — EUROCRYPT 2001, volume 2045 of LNCS, pages 93–118. Springer Berlin / Heidelberg, 2001.

[8] Jan Camenisch and Anna Lysyanskaya. A Signature Scheme with Efficient Protocols. In Stelvio Cimato, Giuseppe Persiano, and Clemente Galdi, editors, Security in Communication Networks, volume 2576 of LNCS, pages 268–289. Springer Berlin / Heidelberg, 2003.

[9] David Chaum. Security without identification: transaction systems to make big brother obsolete. Communications of the ACM, 28:1030–1044, October 1985.

[10] S. Farrell and R. Housley. RFC 3281: An Internet Attribute Certificate Profile for Authorization, April 2002.

[11] Jorn Lapon, Markulf Kohlweiss, Bart De Decker, and Vincent Naessens. Analysis of Revocation Strategies for Anonymous Idemix Credentials. In Bart De Decker, Jorn Lapon, Vincent Naessens, and Andreas Uhl, editors, Communications and Multimedia Security, volume 7025 of Lecture Notes in Computer Science, pages 3–17. Springer Berlin / Heidelberg, 2011.

[12] Christian Paquin. U-Prove Cryptographic Specification v1.1. Technical report, Microsoft Corporation, February 2011.

[13] IBM Research Zürich Security Team. Specification of the Identity Mixer cryptographic library, version 2.3.4. Technical report, IBM Research, Zürich, February 2012.

[14] Eric Verheul. Self-Blindable Credential Certificates from the Weil Pairing. In Colin Boyd, editor, Advances in Cryptology — ASIACRYPT 2001, volume 2248 of Lecture Notes in Computer Science, pages 533–551. Springer Berlin / Heidelberg, 2001.