Integrating Anonymous Credentials with eIDs for Privacy-respecting Online Authentication

Ronny Bjones[1], Ioannis Krontiris[2], Pascal Paillier[3], Kai Rannenberg[2]

[1] Microsoft Corporate, Belgium
ronny.bjones@microsoft.com
[2] Goethe University Frankfurt, Chair of Mobile Business & Multilateral Security,
Grueneburgplatz 1, 60323 Frankfurt, Germany
{ioannis.krontiris,kai.rannenberg}@m-chair.net
[3] CryptoExperts, Paris, France
pascal.paillier@cryptoexperts.com

Abstract. Electronic Identity (eID) cards are rapidly emerging in Europe and are gaining user acceptance. As an authentication token, an eID card is a gateway to personal information and as such it is subject to privacy risks. Several European countries have taken extra care to protect their citizens against these risks. A notable example is the German eID card, which we take as a case study in this paper. We first discuss important privacy and security threats that remain in the German eID system and elaborate on the advantages of using privacy attribute-based credentials (Privacy-ABCs) to address these threats. Then we study two approaches for integrating Privacy-ABCs with eID systems. In the first approach, we show that by introducing a new entity in the current German eID system, the citizen can get a lot of the Privacy-ABCs advantages, without further modifications. Then we concentrate on putting Privacy-ABCs directly on smart cards, and we present new results on performance, which demonstrate that it is now feasible for smart cards to support the required computations these mechanisms require.

1 Introduction

A number of countries have already introduced or are about to introduce electronic identity cards (eID) and drivers licenses. Electronic ticketing and toll systems are also widely used all over the world. As such, electronic devices become widespread for identification, authentication, and payment. Several European Union countries have already rolled out electronic ID cards and several others have committed to rolling out electronic ID cards and are in various stages of planning [1]. The increasing number of electronic identity management infrastructures are creating opportunities for pan-European initiatives of trustworthy services in e-government and e-commerce and set the basis to overcome fragmentation, closed solutions and lack of user control and transparency [2].

As an authentication token and personal data source, an eID card is a gateway to personal information. This implies a set of risks to the privacy of the citizen, via the unwanted disclosure of personal information and its subsequent misuse.

These privacy risks could become even more prominent in the future, if citizens would be using their eIDs not only for e-government services, but also in e-commerce for shopping online, checking into hotels, renting cards online, opening bank accounts, etc.

A recent position paper issued by ENISA on "Privacy Features of European eID Card Specifications" [3] underlines this need for "privacy-respecting use of unique identifiers" in emerging European eID cards and mentions that countries such as Austria and Germany have taken some important steps in this direction. However, some important security and privacy threats still remain. In this paper we take as an example the German eID card, since many consider it to be the most advanced eID deployment [4] and we discuss three of these threats.

Technologies that can help to enhance existing eID card privacy functions are based on privacy-enhanced attribute-based credentials (Privacy-ABCs). In particular, Privacy-ABCs can help prevent monitoring and profiling of the citizens based on the usage of the eID cards, enforce minimal disclosure, offer the choice of complete anonymity for the user, but also help improve the scalability of the underlying infrastructure.

However, although these technologies have been available for a long time, there has not been much adoption in mainstream applications and eID card implementations [3]. We identify three reasons for this: first the available technologies based on Privacy-ABCs use different terminology for their features and even different cryptographic mechanisms to realize them, resulting in a difficulty for developers to understand, compare and use them. Second, the performance of Privacy-ABCs on smart cards (like eIDs) was poor and did not allow practical deployment. And third, Privacy-ABCs are very complex and hard to understand for non-specialists.

Since then, a lot of progress has been made in addressing the above problems. The goal of this paper is to describe this progress and show that Privacy-ABCs are now attractive to be incorporated in eID solutions. In particular we report on the progress being made by the EU-funded project ABC4Trust in bringing together different Privacy-ABC technologies and abstract away their differences. Then we discuss how one of these technologies (namely U-Prove) can be integrated with the German eID card, given the current infrastructure, in order to show that today's eID systems can enjoy some of the benefits of Privacy-ABCs. Finally, we report on the new results we got from experimenting with U-Prove directly on contactless smart cards, indicating that both issuance and presentation can be brought down to the order of milliseconds, making Privacy-ABCs perfectly practical on smart cards.

The rest of the paper is organized as follows. Section 2 analyses the current solution for authentication through the German eID card and discusses the most important privacy and security threats that relate to this. Section 3 introduces Privacy-ABCs and shows their significant potential in addressing these problems. Section 4 shows a case of how U-Prove can be integrated with the German eID system and finally Section 5 takes a step further and discusses the possibility of putting Privacy-ABCs directly on smart cards.

2 Current eID Solutions for User Authentication

The German eID card translates privacy into a set of features. First of all, services must authenticate themselves to citizens. The possibility to choose certain attributes, so that the user can controls the transmission of his/her data is another important feature. Moreover, citizens must consent to every access. On-card verification supports uses such as age verification, while releasing minimum information. Finally, restricted identification creates service-specific pseudonyms that are unlinkable across services [4].

However, the authentication scheme based on the German eID card still raises security and privacy concerns. In Section 2.2 we elaborate on them, but before that we need to understand the entities that are involved in the authentication protocol, as well as the steps of the protocol. We do so in the following subsection.

2.1 The eID Function

The German Federal Office for Information Security's technical guideline TR-03127 [5] specifies the eID card system's architecture. Figure 1 shows an overview of this architecture for online authentication. Three main components participate in the protocol, namely the *user*, the *service provider* and the *eID server*.

The user wants to use an online service through the use of his browser. For that, he must provide part of his personal data to the service provider in order to authenticate. For that purpose, he uses his personal eID and the accompanying software on his personal computer. The service provider offers online services that can be used only by authenticated users. For authenticating the user, the service provider uses the services of a trusted eID server, through which it can query the data in the eID card of the user. The eID server operates as an Identity Service Provider and answers requests for the personal data of users by service providers. It might be operated by the service provider itself or by a third party as an external service. In the latter case, the eID server offers its services to the service providers who want to support the eID functionality within their Web applications. In this case, the eID server reads the data on the eID that are required by the service provider. Furthermore, it stores and manages the authorization certificates and revocation lists.

Figure 1 shows the involved entities, as well as the phases of the authentication process that are executed, when a citizen wants to use an online service and authenticates to the service provider through her eID. As the figure shows, the authentication process takes place according to the following steps:

1. The citizen wants to authenticate with the use of her eID card to the service provider. The service provider forwards the authentication request to the associated eID server. Corresponding to that, the user is presented with the list of functions and data that the service wants to read.
2. A secure channel between the eID server and the eID client is established by use of cryptographic protocols (PACE, terminal and chip authentication).
3. The eID client displays the requested data to the user.

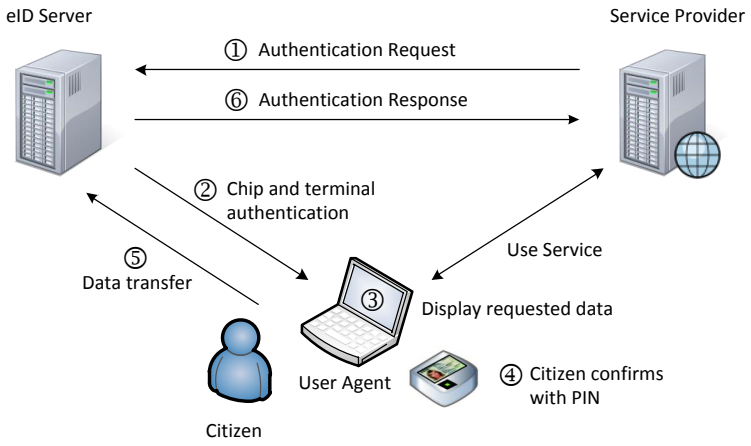

Fig. 1. The steps of online authentication to a service using the German eID card.

4. After reviewing the service information and restricting which data the service provider is allowed to get, the user enters her eID PIN to express consent.
5. The data is transmitted to the eID server. The eID server reads the subset of the eID data according to the corresponding authorization, on behalf of the service provider.
6. The eID server forwards the data back to the service provider as part of the authentication response. Corresponding to that, the service provider verifies the results and decides whether the authentication is successful.

From the process described above, we should note that it adheres to two important privacy features, namely notice and selective disclosure. Indeed, the user is duly informed of the scope of the transaction, i.e., of which identity information is transferred to the application owner and for which purposes the data will be processed. The user is also given the possibility to decide on which identity attributes to disclose and to what extent. These features are in accordance to the privacy requirements of electronic ID cards, as defined by ENISA in 2009 [6].

2.2 Security and Privacy Problems

When compared to the privacy features offered by other European eID card specification [3], the German new eID card is one of the most privacy-friendly solutions. However, it follows the passive authentication protocol with bearer tokens that we described in the previous section. Bearer tokens (security tokens) containing user's claims are delivered by the eID server to the service provider without user intervention. This model is subject to several threats [7]. Here we will focus on the most important ones, relevant to security, privacy and availability.

eID server knows all user transactions Even though the eID server does not necessarily need to know where the user is authenticating and which service she is requesting, this knowledge is passed by design to the eID server in the current eID solution. More specifically, the eID server is involved each time a user authenticates to a service provider using her eID, and is able to keep track of the user actions. This enables the eID server to trace and link all communications and transactions of each user.

This pattern is followed in most federated identity management systems today, and it can be also observed in STORK's architecture for eID interoperability between different European countries [8]. In the physical world we might have to show a government issued picture ID on different occasions. The issuer of those picture IDs is not aware that we show those at this specific location. In the digital world however, the default case is that the issuer knows when you present your ID.

eID server knows all customers of the service provider Reversing the above threat, the involvement of the eID server in every user authentication constitutes a privacy threat for the service providers as well, since the eID server learns all the customers trying to access a service. Especially if the eID server is operated by a private company, it might be a competitive threat, if it can learn all the customers of another company (i.e. the service provider).

User impersonation Since the user does not perform an active role in the information exchange between the eID server and the service provider, there is a high security risk of user impersonation by insider attackers at the eID server or outsider intruders when they would gain access to the eID server's resources.

An eID server under control of an attacker (insider or outsider) has the ability to impersonate every user at applications using eIDs for authentication. For example, insiders can copy or alter user's credentials and as such steal the identity of a user. In general, in a federation scenario, the insiders or outsiders who learn a user's credentials can impersonate the user and get access to the assets at different applications involved in the federation.

Availability The eID server becomes a business critical component as it is needed for every transaction the user does with the applications. Denial of Service attacks towards the eID server will impact all applications using the service. Attacking this component may have a huge economic impact because the attack spreads over different services.

All of the above problems become critical when there are currently only a couple of eID servers operating, despite the view of the German government that this service will be offered by multiple servers. Meanwhile, the requirement that the eID providers are not able to track the behaviour of eID holders is becoming more prominent. In the evaluation assessment of the recent proposal of a Regulation "on electronic identification and trusted services for electronic transactions in the internal market" [9] it is stated that a solution to this tracking problem should be aligned with the current ongoing revision of the Data

Protection Directive and include specifically privacy-by-design rules. In the next section we discuss specifically how the above threats can be addressed with the privacy-by-design model.

3 Privacy-ABCs to the Rescue

To alleviate the above threats and offer more flexibility, governments can turn to a claims-based architecture [10]. The claim-based architecture is a design pattern used by system architects to implement claims-based identity. The main purpose of claims-based identity is to externalize authentication. The service provider's interest is not to authenticate the user, but rather receive verified claims about the user, based on which access to the service is decided. That is, the service provider publishes a policy on accessing a specific resource and expects to receive claims and identity tokens from trusted sources that satisfy this policy. The trusted sources that issue such security tokens are the *identity service providers* (IdSP), sometimes also called identity providers for simplicity. In the particular eID system that we are studying in this paper, the eID server has the role of the IdSP.

The claimed-based architecture allows separation between service providers and identity service providers, so that there is no direct exchange of information between them. Instead, the user lies in the middle, having control of the exchange of his identity information. Then, on one side identity providers authenticate the user and issue security tokens, and on the other side service providers consume tokens. Because a service provider relies on the IdSP to provide authentic information about the user, it is called the *relying party* (RP).

An example of claim-based architecture is the Identity Metasystem [11]. Claim-based architectures can use privacy-respecting credential systems (Privacy-ABCs) to provide untraceability and minimal disclosure. Examples of such credential systems are Idemix [12] and U-Prove [13]. Over the last few years, Idemix and U-Prove have been developed to offer an extended set of features, even though these features are named differently and they are realized based on different cryptographic mechanisms. Recently, the European research project ABC4Trust [14], was initiated with the goal to alleviate these differences and unify the abstract concepts and features of such mechanisms. In particular, it brings them under the common name *privacy-preserving attribute based credentials*, or *Privacy-ABCs* [15]. So, Privacy-ABCs are privacy respecting credentials that are defined over these concepts and features and are independent from the specific cryptographic realization beneath. Overall, Privacy-ABCs offer the following advantages [15]:

- Privacy-ABCs are by default untraceable. Even when they are obtained on-demand, IdSPs are not able to track and trace at which sites the user is presenting the information.
- Privacy-ABCs can be obtained in advance and stored by the user while still being able to disclose the minimal amount of information needed for a particu-

lar transaction. So, the real-time burden of the IdSP is diminished, improving scalability.
- To prevent identity theft and "credential pooling", i.e., multiple users sharing their credentials, credentials can be bound to a *secret key*, i.e. a cryptographically strong random value that is assumed to be known only to a particular user. A presentation token derived from such a key-bound credential always contains an implicit proof of knowledge of the underlying secret key, so that the verifier can be sure that the rightful owner of the credential was involved in the creation of the presentation token. As an extra protection layer, the credentials can also be bound to a trusted physical device, such as a smart card (i.e. the eID card itself), by keeping the secret key in a protected area of the device. That is, the key cannot be extracted from the device and so it is not possible to make a presentation proof without the device.
- Instead of complete anonymity, if desired, users can generate an unlimited number of pseudonyms or a batch of Privacy-ABCs and use them at the same or different relying parties. Presentations of pseudonyms or different Privacy-ABCs are cryptographically unlinkable, meaning that given two different presentations of the credentials, one cannot tell whether they were generated by the same user. In cases where it is undesirable that users are able to generate multiple identities on the same site, the relying party can impose a *scope-exclusive pseudonym*, meaning that for a scope string (e.g. a URL) the user can only register a single pseudonym. This feature is useful in applications where the user should not be able to create multiple identities based on a single credential, like for example in online petitions.

So, privacy-ABCs have significant potential to enhance existing eID card privacy functions. Their integration is perfectly realizable today, and does not necessarily require modifications at the current infrastructure of the eID server and the eID cards. This is demonstrated in the next section, where we take as a paradigm one of the Privacy-ABC technologies, namely U-Prove, and show how it can be integrated in the German eID system.

4 Integrating Privacy-ABCs to Existing eID Systems

U-Prove has been integrated with the German eID system and it has been demonstrated in a typical e-Participation scenario [16]. In particular, it was demonstrated in a local referendum application, where the citizens had to prove their eligibility to participate, by proving properties of their identities, while at the same time their anonymity is preserved. In this section, we generalize the discussion and show the entities and the protocol involved. Even though in our discussion below we still use U-Prove, the same would apply for other Privacy-ABC systems as well (e.g. Idemix). We only use U-Prove here as an instance, since it was used in the initial implementation [16].

Compared with the standard German eID system we discussed in Section 2, the entities remain the same only that a new entity has been introduced and in

particular the U-Prove issuer. The U-Prove issuer has two responsibilities: first to validate the claims issued by the eID server and second to issue a U-Prove token that contains these claims. By deploying the U-Prove issuer, applications can leverage U-Prove Tokens providing unlinkability and anonymity to the users.

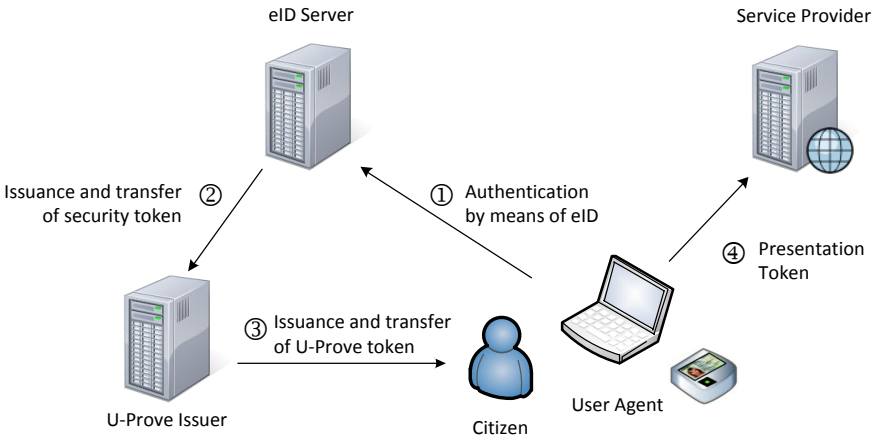

Fig. 2. U-Prove Integration with the German eID system.

As Figure 2 shows, the authentication process takes place according to the following steps:

1. The user wants to authenticate by means of eID card to the application. For that, the user is redirected to the eID server of her choice and she is prompted to present the eID card and PIN-code. The eID server then validates the identity of the user.
2. The eID server produces a security token containing the claims on user attributes that are requested by the application. This token arrives at the U-Prove issuer.
3. At the U-Prove issuer, the current security token is exchanged for a U-Prove token, after the U-Prove issuer has validated the received token.
4. The user presents the U-Prove token to the application through a U-Prove Presentation Proof.

The above protocol achieves a challenging combination of high assurance on the identity of individuals through their eID and full anonymity while using a service. User anonymity is made possible since the presentation token cannot be tied back to the true identity of the user. The true identity of the user was established during the authentication at the eID server, but at Step 3, the eID token was exchanged for a U-Prove token. U-Prove ensures the unlinkability between the issuance of the U-Prove token and its usage through the presentation

proof (Step 4). Even if the U-Prove issuer and the service provider collude, there is no way to link the two together. Actually, this offers the advantage to assign an extra role to the U-Prove issuer, if desirable: that of the validator of the U-Prove tokens at the RP. That would alleviate this extra burden from the RP without having to introduce an extra entity in the architecture and without having to make any compromise in terms of security and privacy.

The example above illustrates how Privacy-ABCs and eID systems can be combined. The idea of having high assurance on the identity by means of a smart card and being anonymous in the actual transaction on the RP sounds esoteric, but it can be easily accomplished by means of Privacy-ABCs. This combination is possible by leveraging e.g. a service in the cloud. This service will learn the user's attributes coming from the eID server but will not learn where the user is using them. The U-Prove issuer will learn as much information as the eID server and so both services are equal from a privacy threat modeling perspective and should be protected in a similar way.

However a few issues of trust management need to be addressed, if a new type of entity is introduced into the system. The U-Prove issuer learns about the attributes that are requested and so (over time) can build a profile of the user and the attributes that the user needs for her service at which time, e.g. (being adult, asking for the respective credential Friday night, having been checked for AIDS recently with no AIDS having been detected, being eligible for medical consultation via a special type of assurance).

One could say, that the cleanest solution would be to simply regulate the U-Prove issuer to not store any attributes after credential issuance and to audit him for this. However experience with ID issuers raises doubt, that a "no-records-taken"-policy would be accepted by all stakeholders, who want to look after the U-Prove issuer and who may require some record-keeping, though the authors do not really see a hard reason for record keeping. Quick re-issuance of credentials after a user has lost them does not seem to be so important, that it would justify record-keeping with that many privacy implications.

Actually, one measure to mitigate the privacy risk at the U-Prove issuer lies in the heart of Privacy-ABCs philosophy. The eID server should always issue tokens containing claims for *all* attributes in the eID cards, and let the user decide which of these attributes to reveal to the RP, during the presentation proof. In this way, the eID server cannot draw any conclusion on the type of the application the tokens are being used for. At the same time, it is important that several U-Prove issuers are available; this allows users to spread the knowledge of certain types of attributes over their selection of providers.

Besides privacy reasons, the organizational setting of the U-Prove issuer needs to assure that no monopoly situation can arrive, as a monopoly would also be risky from an availability and cost perspective. While monopolies can achieve scale effects in network-based industries and therefore can have cost advantages, the cost risk from the point of user is the lock-in situation, that comes with the monopoly and that allows the monopoly provider (in this case the issuer) to dictate prices. Actually a U-Prove issuer could and should fall under the rulings

of the recently proposed regulation on electronic identification and trusted services for electronic transactions in the internal market [17]. Article 11 (4) therein explicitly mentions pseudonyms (i.e. a special type of attributes), which could be issued by eID servers (and also U-Prove issuers). The requirements set out in the proposed regulation can be expected to establish appropriate trust into the token issuers.

The most important aspect of the U-Prove issuer is that its cloud instances do not learn the relationship between the user and RP. It is actually this relationship that affects the privacy of the user because it makes profiling of the user possible. It is in the interest of the eID server to apply Privacy-ABCs, in order not to be seen as a "monitoring beacon". Privacy-ABCs will also protect the privacy of the RPs because now there is no third party (eID server) which learns all their customers. Especially when the eID server is run by a private company, learning all the RP's customers is seen as a competitive threat for the RP.

Germany has gone a long way in adding privacy to the eID card, much further as any other system. While this is certainly to the right direction, most eID systems being deployed in Europe would benefit even more from Privacy-ABCs. For that, it is crucial that Privacy-ABCs become part of the eID card itself. So the delivery of the claims to the RP is done under control of the user and the eID card. This removes immediately many of the threats discussed in Section 2.2 and increases the privacy of the user and RP. Can we put Privacy-ABCs on eID cards? Could they run efficiently on the smart cards? This will be discussed in the next section.

5 Privacy-ABCs on Smart Cards

There have been several approaches to implement Privacy-ABCs on smart cards. Bichsel [18] and Balasch [19] focus on providing the arithmetic functionality required, i.e. fast modular arithmetic. Balasch implemented the arithmetic using AVR microcontrollers, whereas Bichsel used the JCOP platfom. Later, Bichsel et al. presented the first practical implementation of a Camenish-Lysyanskaya-based Direct Anonymous Attestation scheme on a Java Card 2.2.1 [20] with a performance close to 7.5 seconds. Tews and Jacobs [21] considered U-Prove and succeeded in performing a presentation proof in about 5 seconds for 2 attributes and 8s for 4 attributes. Batina et al. [22] suggest to use self-blindable certificates and put forward an implementation that requires about 1.5s to perform presentation for 1 attribute. In 2011, Mostowski and Vullers implement U-Prove on a MULTOS card and reach about 0.5s (resp. 0.8s) for 2 (resp. 5) attributes. Up to our knowledge, no implementation of a Privacy-ABC system is available on a contactless smart card at this time.

We have chosen to focus on the full-fledge version of U-Prove as opposed to the device-binding version [23], thus showing the applicability and user-friendliness of a complete Privacy-ABC system running on an eID card. The chosen smart card platform is a 32-bit chip made available by Invia [24]. The component features a Sparc v8 Leon II core and embeds a lightweight public-key

coprocessor called MEXPA running at 33MHz. We assumed an ISO/IEC 14443 contactless interface running at a pessimistic baudrate of 106 Kbits per second.

U-Prove describes an issuance phase and a presentation phase. The two protocols may employ either a group of integers modulo a prime number or an elliptic curve defined over a field of large prime characteristic. We have considered the case of an implementation based on elliptic curves for increased flexibility at the algorithmic level, although some of our optimizations are readily applicable to groups of integers.

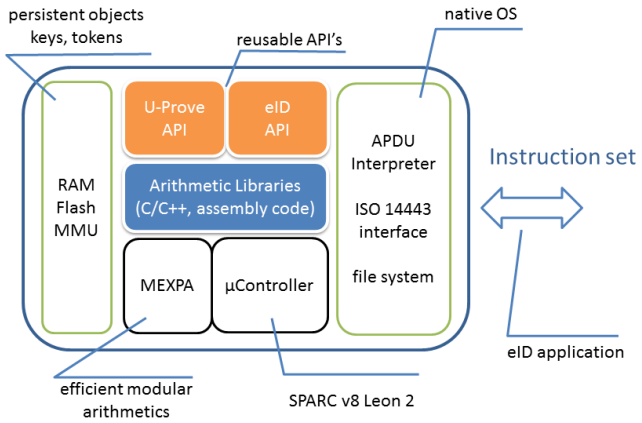

Fig. 3. Schematics of a contactless smart card integrating U-Prove and eID application.

At a high-level view, we have undertaken and combined the following approaches:

1. reformulation of U-Prove's protocol flow to identify the critical operations performed by the smart card and minimize on-board computations;
2. maximal delegation of unsensitive computations to the Issuer and Verifier within the limits of neutrality towards cryptographic security;
3. off-line/on-line optimization using a maximal number of precomputed values stored in non-volatile memory as coupons. Coupons are internal variables that can be generated by the smart card beforehand, thus reducing the total latency when transactions (either issuance or presentation) take place;
4. boosting point operations using the best suited coordinate system (Jacobian, affine or mixed Jacobian/affine) for each operation performed on the curve;
5. optimizing scalar multiplications by aggregating multiple products: computing a double multiplication $([k_1]B, [k_2]B)$ with the same basepoint B allows to share intermediate variables even when B changes from one execution to the next;
6. finely compare implementation strategies and trade-offs given the performance of low-level hardware operations to find optimal settings.

U-Prove Issuance Further optimizations are made possible by the use of NIST curves as recommended in the specifications of U-Prove [23]. Taking the NIST curve P-256, we get an estimated cost of about 4.1 million clock cycles for the most critical part of the issuance phase, neglecting modular additions and subtractions. The memory size required remains moderate, namely of the order of 1KB of RAM. Under a clock frequency of 33MHz, 4.1 million cycles correspond to 124ms. We add a 30% overhead to take into account modular additions, pointer management and minor CPU-operated instructions. A pessimistic additional 20ms is added to reflect one-time minor operations such as hash computations.

The total bitsize of transmissions in the issuance protocol amounts to $10 \log_2 p$ where p is the field characteristic which, assuming a baudrate of 106 Kbps (slowest configuration), leads to an estimated 78ms. Putting it altogether, and neglecting the execution time of off-board operations, we end up with an expected running time of about 259ms for the complete issuance protocol.

Presentation Proof For the presentation phase, we take the typical case where the user generates the presentation proof, in which some attributes are disclosed and some are not. Similar to the issuance phase, there are many possible algorithmic options for this phase as well and we may rely again on precomputations (coupons) and various algorithmic optimizations of aggregated scalar multiplications.

Overall, we find that the cost of the presentation proof amounts to $38.42 \times (n - |D|)$ milliseconds on the target chip, where $n - |D|$ is the total number of undisclosed attributes. We upper bound the extra time needed by the remaining computations by about 45 to 50ms. This gives a typical presentation phase of 434ms for 10 unrevealed attributes, thus providing evidence that both the issuance and the presentation phase of U-Prove can be efficiently implemented on a state-of-the-art contactless card.

Areas for Further Optimization Operations on the elliptic curve could be made faster by using efficiently computable endomorphisms over the group of points as with the GLV method [25]. For instance, curves over a field extension allow to use the Frobenius map to speed-up scalar multiplication. Also, curves with coefficients $a = 0$ or $b = 0$ that have endomorphisms that one can evaluate using roots of unity are quite appealing (other examples with specific values for a and b are known). Also, field arithmetics can be boosted using extended fields. Taking a curve over $\mathbb{F}_{p^2} = \mathbb{F}_p/(p^2 + 1)$, a field multiplication which usually requires two operands of n bits boils down to 3 multiplications with half-size operands and one can replace a modular reduction from $2n$ bits to n bits with two reductions from n bits to $n/2$ bits.

On a general-purpose 32-bit CPU, taking a field extension with a pseudo-Mersenne characteristic and a sparse irreducible polynomial would probably be the best possible choice as one can rely on both fast arithmetics and efficient endomorphisms. Also, selecting an Edwards curve would slightly improve speed. As investigated in [25], the best timings on a 64-bit Intel processor when no

crypto-coprocessor is available are realized with Edwards curves over $\mathbb{F}_{p^2}$ using endomorphisms as per the GLV and GLS techniques.

6 Conclusions

A potential future deployment of Privacy-ABCs in eID schemas would allow going beyond the existing privacy-preserving capabilities of the German model. In this paper we have showed the benefits of such an integration in terms of preserving the privacy of the user. Overall, based on several properties, Privacy-ABCs bear a high potential to challenge what must have been considered as necessary processing of personal data in the past. If deployed broadly, Privacy-ABCs would enable the revision of the understanding of necessary processing and require reassessment of existing systems. Integrating them into the upcoming European framework on electronic identification and trust services for electronic transactions in the internal market seems possible, though further details need to be analysed. For example, one could extend further the discussions in this paper on establishing the appropriate trust on the token issuers, as well as continue the analysis of further optimizations of the performance on smart cards, as discussed in the last section.

7 Acknowledgements

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement no. 257782 for the project Attribute-based Credentials for Trust (ABC4Trust).

References

1. S. Ahlswede and J. Gaab, "eIDs in Europe," Deutsche Bank Research, Tech. Rep., September 2010.
2. "A Strategy for ICT R&D and Innovation in Europe: Raising the Game," Commission Communication, COM(2009) 116, 2009.
3. I. Naumann and G. Hogben, "Privacy Features of European eID Card Specifications," ENISA, Position Paper, January 2009.
4. A. Poller, U. Waldmann, S. Vowe, and S. Turpe, "Electronic identity cards for user authentication – promise and practice," *IEEE Security & Privacy*, vol. 10, pp. 46–54, 2012.
5. "Architecture electronic Identity Card and electronic Resident Permit," German Federal Office for Information Security, Technical Report TR-03127, Version 1.13, 2011.
6. I. Naumann, "Privacy and Security Risks when Authenticating on the Internet with European eID Cards," ENISA, Risk Assessment Report, November 2009.

7. R. Bjones, "Architecture serving complex Identity Infrastructures," Trust in Digital Life, Tech. Rep., November 2011.
8. I. Krontiris, H. Leitold, R. Posch, and K. Rannenberg, "eID Interoperability," in *Handbook of eID Security*, W. Fumy and M. Paeschke, Eds. Publicis Publishing, 2011.
9. "Impact Assessment accompanying the proposal for a regulation of the European Parliament and of the council on electronic identification and trust services for electronic transactions in the internal market," European Commission, SWD(2012) 136, 2012.
10. K. Cameron, R. Posch, and K. Rannenberg, "Proposal for a common identity framework: A User-Centric Identity Metasystem," in *The Future of Identity in the Information Society – Opportunities and Challenges*, K. Rannenberg, D. Royer, and A. Deuker, Eds. Springer, 2009.
11. K. Cameron and M. B. Jones, "Design Rationale behind the Identity Metasystem Architecture," Microsoft, Tech. Rep., February 2006.
12. J. Camenisch and E. Van Herreweghen, "Design and implementation of the idemix anonymous credential system," in *Proceedings of the 9th ACM conference on Computer and Communications Security (CCS '02)*, 2002, pp. 21–30.
13. S. Brands, *Rethinking Public Key Infrastructures and Digital Certificates; Building in Privacy.* MIT Press, 2000.
14. "ABC4Trust: Attribute-Based Credentials for Trust." [Online]. Available: https://abc4trust.eu
15. "D2.1 Architecture for Attribute-based Credential Technologies - Version 1," ABC4Trust, Deliverable D2.1, 2011.
16. R. Bjones, "eParticipation Scenario Reference Guide," Microsoft, Tech. Rep., October 2010.
17. "Proposal for a regulation of the European Parliament and of the council on electronic identification and trust services for electronic transactions in the internal market," European Commission, COM(2012) 238/2, 2012.
18. P. Bichsel, "Theft and Misuse Protection for Anonymous Credentials," ETH Zürich, Switzerland, Master's thesis, 2007.
19. J. Balasch, "Smart card implementation of anonymous credentials," K. U. Leuven, Belgium, Master's thesis, 2008.
20. P. Bichsel, J. Camenisch, T. Groß, and V. Shoup, "Anonymous credentials on a standard java card," in *Proceedings of the 16th ACM conference on Computer and communications security (CCS '09)*, 2009, pp. 600–610.
21. B. J. H. Tews, "Performance issues of Selective Disclosure and Blinded Issuing Protocols on Java Card," in *Information Security Theory and Practice, LNCS 5746*, 2009, pp. 95–111.
22. L. Batina, J. henk Hoepman, B. Jacobs, W. Mostowski, and P. Vullers, "Developing efficient blinded attribute certificates on smart cards via pairings," in *In Gollmann, D, CARDIS 2010*, 2010.
23. Microsoft, "U-Prove Cryptographic Specification V1.1," February 2011.
24. Invia, "Modular Exponentiation IP," available at http://www.invia.fr/Modular-Exponentiation-21.html.
25. P. Longa and C. Gebotys, "Efficient techniques for high-speed elliptic curve cryptography," in *In Advances in Cryptography, CHES 2010*, 2010, pp. 80–94.