D3.1 Scientific comparison of ABC protocols

Part II: Practical comparison

Fatbardh Veseli, Tsvetoslava Vateva-Gurova, Ahmad Sabouri, Michael Østergaard Pedersen, Jesus Luna

Editors:	*Fatbardh Veseli (GUF)*
Reviewers:	*Pascal Paillier (CRX), Jonas Lindstrøm Jensen (ALX)*
Identifier:	*D3.1*
Type:	*Deliverable*
Version:	*1.1*
Date:	*14/07/2014*
Status:	*Final*
Class:	*Public*

Abstract

This part of deliverable D3.1 presents a practical comparison of two instantiations of Privacy-ABC technologies, namely of U-Prove and Idemix in terms of efficiency, functionality, and security assurance. The comparison is made based on the framework for benchmarking Privacy-ABC technologies, which developed in the project, providing a wide range of results following the lifecycle of Privacy-ABCs, starting from issuance, presentation, inspection, and revocation.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 257782 for the project Attribute-based Credentials for Trust (ABC4Trust) as part of the "ICT Trust and Security Research" theme.

Members of the ABC4TRUST consortium

1.	Johann Wolfgang Goethe Universität Frankfurt	GUF	Germany
2.	Alexandra Institute AS	ALX	Denmark
3.	Research Academic Computer Technology Institute	CTI	Greece
4.	IBM Research – Zurich	IBM	Switzerland
5.	Miracle A/S	MCL	Denmark
6.	Nokia-Solutions and Networks Management International GmbH	NSN	Germany
7.	Technische Universität Darmstadt	TUD	Germany
8.	Unabhängiges Landeszentrum für Datenschutz	ULD	Germany
9.	Eurodocs AB	EDOC	Sweden
10.	CryptoExperts SAS	CRX	France
11.	Microsoft NV	MS	Belgium
12.	Söderhamn Kommun	SK	Sweden

List of Contributors

Chapter	Author(s)
Executive Summary	Fatbardh Veseli (GUF)
Chapter 1	Fatbardh Veseli (GUF), Michael Østergaard Pedersen (MCL), Tsvetoslava Vateva-Gurova (TUD), Jesus Luna (TUD)
Chapter 2	Fatbardh Veseli (GUF)
Chapter 3	Ahmad Sabouri (GUF), Fatbardh Veseli (GUF)
Chapter 4	Tsvetoslava Vateva-Gurova (TUD), Jesus Luna (TUD), Fatbardh Veseli (GUF)

Executive Summary

This is the second part of the deliverable "D3.1 Scientific Comparison of ABC Protocols", which presents the results of the practical comparison of two implementations of Privacy-ABC technologies. The comparison is done following the work done in the project on defining a framework for benchmarking Privacy-ABC technologies, which will be separately published as deliverable D2.3 "Benchmarking Criteria" [D2.3] and is grouped into three main dimensions: *efficiency*, *functionality*, and *security assurance*. Wherever applicable, we follow the lifecycle of Privacy-ABCs for organising further the comparison results, starting from the issuance, presentation, inspection, and revocation.

The results presented in this deliverable present the comparison between two instantiations of Privacy-ABC technologies, namely *U-Prove* based on Brands [Brands00] signatures, and *Idemix* based on the Camenisch-Lysyanskaya [CL03] scheme, both further developed as part of ABC4Trust to fit under a unified architecture [H2.1]. The document starts with an introduction chapter, defining the focus of the comparison (the compared Privacy-ABC technologies), explaining the foundation of the comparison results, and the focus of each of the compared dimensions of Privacy-ABC technologies.

In our work, we follow a user-focused approach for comparing the two instantiations of Privacy-ABC technologies. In this regard, efficiency is considered to be very important, as it directly impacts the potential acceptance of the technologies by the users and it can determine also its adoption by service providers. We pay a special focus to this dimension and therefore define three efficiency sub-dimensions, namely the computational, communication, and storage efficiency.

On top of that, we provide a comparison of the supported Privacy-ABC features as defined in the ABC4Trust architecture document [H2.1], consider also additional functionality aspects of the technologies, which may determine their suitability in different platforms, such as mobile devices and devices with low storage and computational capability. Finally, we consider a comparison based on the security assurance the technologies provide.

Table of Contents

Index of Figures

Index of Tables

1 Introduction

The goal of this deliverable is to present the results of the practical comparison of two different Privacy-ABC technologies based on different efficiency, functionality, and security assurance criteria. The comparison is based on a common framework for benchmarking Privacy-ABC technologies, which defines the criteria for the above-mentioned dimensions, which is published in a separate document, namely in "D2.3 Benchmarking Criteria" [D2.3].

The herein presented results bring a comparison of the Privacy-ABC schemes (combinations of different cryptographic building blocks) as they are implemented in the ABC4Trust Reference Implementation.

1.1 The compared Privacy-ABC technologies

During the ongoing work in the project, there have been many developments in the architecture for Privacy-ABCs, especially on the crypto architecture, making it more modular, where different cryptographic building blocks can be easier plugged and integrate with the rest of the building blocks, making a new instantiation of a Privacy-ABC scheme. In this regard, the current reference implementation has so far integrated a number of cryptographic building blocks, all of which are based on well-known constructions from the cryptographic literature.

The most central building block of Privacy-ABC systems is the signature building block. Currently we have implemented two different instantiations of the signature building blocks: one is Idemix, based on the Camenisch-Lysyanskaya [CL03] signature scheme over a group of signed quadratic residues modulo a safe RSA modulus, while the other one is U-Prove, based on the Brands [Brands00] signature scheme. The rest of the building blocks so far only have a single instantiation shared between the two instantiations in the current implementation. Therefore, the current results reflect this fact[1].

The revocation and inspection scheme is studied separately, as it can be integrated in different schemes and is not inherent to the given instantiations of the Privacy-ABC schemes in the project. The revocation scheme is based on the Camenisch-Lysyanskaya accumulators [CL02] and the inspection scheme is based on the Camenisch-Shoup verifiable encryption scheme [CS03].

There are also a number of other building blocks used to provide pseudonyms, non-equality proofs, inequality proofs, set-membership proofs, as well as a number of helper building blocks.

1.2 Comparison dimensions

This document presents results of different comparison criteria organised into three main dimensions, namely efficiency, functionality, and security assurance.

[1] The Privacy-ABC system that uses CL signatures was also known as Idemix from IBM, while the other technology, based on Brands signatures, is used in Microsoft's U-Prove. However, we have used updated implementations of these two technologies, which have further been developed as part of ABC4Trust project, which may include additional features from the current versions that may be publicly available from the two providers. While IBM is making the ABC4Trust version as their main brand (Idemix), Microsoft may chose a different version for branding its current U-Prove.

1.2.1 Efficiency comparison

Comparison of the efficiency of operations of the two chosen objects under comparison is done for all the stages in the lifecycle of Privacy-ABCs, starting from issuance, presentation, inspection, and revocation. The results presented in this document reflect the overall efficiency of the tested operations not only on the crypto level (CE level), but they also include the additional overhead on the ABCE, including processing XML code, and the additional calls to the lower layers, as defined in the ABC4Trust architecture. Furthermore, the current reference implementation of ABC4Trust has not been optimised (yet), so better efficiency results could certainly be possible.

1.2.1.1 Efficiency comparison and the security levels

Increasing the key size of a secure cryptographic scheme will usually lead to an increased security level at the cost of lower performance of the cryptographic operations. Therefore it is important that when comparing different cryptographic schemes, they are compared at the same security level. Since different cryptographic schemes often require different key sizes to provide same level of security, the public document H2.1 ”ABC4Trust Architecture for Developers” [H2.1] introduced the concept of security levels to aid users of the ABCE API in choosing a security level adequate for their needs without worrying about the actual key size of the underlying cryptographic schemes. This security level would correspond to the security level of an ideal symmetric cipher with the given key size. Besides convenience for the user, this concept of security levels is also important due to the fact that the Cryptographic Architecture of ABC4Trust allows users to mix and match cryptographic schemes, and by specifying a security level instead of an actual key size, the Cryptographic Engine can generate keys for the various cryptographic components to ensure that they all provide the same security level. In Table 1.1 we list the various security levels based on the ECRYPT II recommendations [Smart11].

The concept of security levels has however not been implemented yet, so for the time being one still needs to specify an actual key size when instantiating the Cryptographic Engine. For the comparisons listed in this document that means that one specifies a common key size for CL signatures, CL accumulators and Brands signatures used in U-Prove, and also for the groups used for pseudonyms.

Fortunately in this case (in our experiments) they all provide the same security level for a given key size and the efficiency on a given key size reflects the comparison on the same security level. Both CL signatures and CL accumulators are based on the Strong RSA assumption, so assuming that factoring is the best attack to break these schemes, we can take their key size from the “asymmetric” column in the ECRYPT II recommendations. There is no security reduction for U-Prove in our implementation, but assuming that solving the discrete logarithm problem is the best way to break the scheme, we must take the key size from the “logarithm group” column of the ECRYPT II recommendations, since our U-Prove implementation currently does not use elliptic curves. Pseudonyms and commitments are also based on discrete logarithms, and therefore also use the “logarithm group” key size. Looking at those numbers, one can see that ECRYPT recommends identical key sizes for the same level of security in all cases. Mapping this to the security levels from H2.1 [H2.1] we get that 1024-bit keys correspond to a security level between 72-80 bits, and 2048-bit keys correspond to a security level between 96-112 bits. For discrete logarithms this works because we currently do not implement U-Prove over elliptic curves.

Note that this only works for the specific combination of cryptographic building blocks we use for these benchmarks.

Table 1.1 - Security Levels (symmetric equivalent) based on ECRYPT II

Symmetric Security (bits)	Protection	Comment
32	Attacks in “real-time” by	Only acceptable for auth. Tag

	individuals.	size
64	Very short-term protection against small organizations.	Should not be used for confidentiality in new systems.
72	Short-term protection against medium organizations, medium-term protection against small organizations.	
80	Very short-term protection against agencies, long-term protection against small organizations.	Smallest general-purpose level, $\leq$ 4 years protection.
96	Legacy standard level.	Approx. 10 years protection.
112	Medium-term protection.	Approx. 20 years protection.
128	Long-term protection.	Good, generic application independent recommendation (approx. 30 years protection).
256	"Foreseeable future"	Good protection against quantum computers unless Shor's algorithm applies.

1.2.1.2 Experiment setup

The development environment used the same setting as the environment for the development of the Reference Implementation, so the experiments were developed using the Java SDK using an Eclipse[2] environment for the project integration.

The experiments are meant to be as realistic as possible and are founded on the existing scenarios defined in the Soederhamn pilot, such as credential specifications, issuance and presentation policies, but further refined to include other scenarios that we deemed important to provide more complete results.

Especially for the computational efficiency comparison, it is important to note that the figures reflect the computational efficiency (time) performed on a computer with the configuration given in Table 1.2 running on an OS X 10.8.5 (Macintosh) operating system.

Table 1.2 - Hardware configuration of the experiment setup environment

Processor	1.8 GHz Intel Core i7
Number of processors:	1
Number of Cores:	2
L2 Cache (per Core)	256 KB
L3 Cache	4 MB
Memory	4 GB 1333 MHz DDR3

[2] Information and download available at http://www.eclipse.org/kepler/.

1.2.2 Functionality

The functionality comparison is mostly qualitative and is oriented towards comparing the given instantiations of the Privacy-ABC schemes based on their support for different functional features. Typical comparison factors include the supported features of Privacy-ABCs of the given instantiations of Privacy-ABC technologies, as well as identification of other practical differences between these technologies related to the suitability of the technology in different platforms (e.g. offline devices) and the limitations of the technology that may impact the ease of use.

1.2.3 Security assurance

The aim of this document is to compare Privacy-ABC technologies by using a set of security assurance metrics that were mainly developed taking into account the specific properties of the technologies. The results of the security assurance comparison aim to provide information related to the security of the technologies. They should show how well the technologies can support the users in terms of incidents such as compromised key or what mechanisms the specific technology applies in order to keep the sensitive information secure.

1.3 Presentation of the results

The comparison in this document is made based on the framework for benchmarking Privacy-ABC technologies, which defines the set of benchmarking criteria. Each criterion has a name and ID, and the comparison based on the criteria should, at minimum, the information shown in Table 1.3, possibly complemented with additional information, whenever necessary. However, wherever feasible, results will also be presented visually in charts, especially for the efficiency comparison.

Table 1.3 - Model for a table summarizing the results

Attribute	Value
Name	Computational efficiency for Advanced Issuance and Credential Update (If supported)
ID	Iss-P3
Result	
Comments	

2 Efficiency

As these technologies are built on different cryptographic building blocks, it is interesting and useful to see how efficient these technologies are for different types of operations and features. Especially for the users, it is important that the technologies are as efficiency as possible, in order to be suitable to different application scenarios and to be acceptable by the users.

In this regard, we distinguish between three different types of efficiency comparison types, namely the *computational* efficiency, which measures to time to perform certain operations (features) of Privacy-ABCs; the *communication* efficiency, which focuses on measuring the data sizes produced by certain operations and exchanged between parties during those operations; and *storage* efficiency, which compares how the compared Privacy-ABC technologies differ in terms of the size of the data the User needs to store.

2.1 Computational efficiency

Computational efficiency is a crucial distinguishing factor for the acceptance of a certain Privacy-ABC technology. A computationally more efficient Privacy-ABC technology enables users to quickly perform the underlying cryptographic operations during the issuance and presentation phases, leading to a better performance of the application using the Privacy-ABC technology. The computational efficiency results are expressed in time units (seconds) and the following section presents a summary of the most important comparison results along the lifecycle of the Privacy-ABCs, namely issuance, presentation, inspection, and revocation.

2.1.1 Issuance

As there are different types of issuance possible, the computational efficiency for each of them may differ. For this purpose, we compare the difference in the efficiency of "simple" from the more "advanced" issuance types, such as issuance with key binding, and issuance carrying over attributes from another credential.

2.1.1.1 "Simple" issuance

"Simple issuance" or issuance is the simplest form of issuance of Privacy-ABCs, whereby the issuance starts "from scratch", meaning that the User does not need any prior Privacy-ABC to get issued a credential for this type of issuance. The credential issued in this case had 6 different attributes and is used for comparison with other experiments in the following sections.

Figure 2.1 shows the results of the comparison for the two given instantiations of Privacy-ABCs using both Idemix and U-Prove for two different key sizes, namely 1024 and 2048 bits. As we can see, both technologies provide a similar computational efficiency, with a slight but negligible advantage for Idemix. An increase in the key size has a very strong (negative) impact on the computational efficiency of issuance, where the impact is almost linear to the factor of increase in the key size (in our case, doubling the key size results in almost double time to perform issuance).

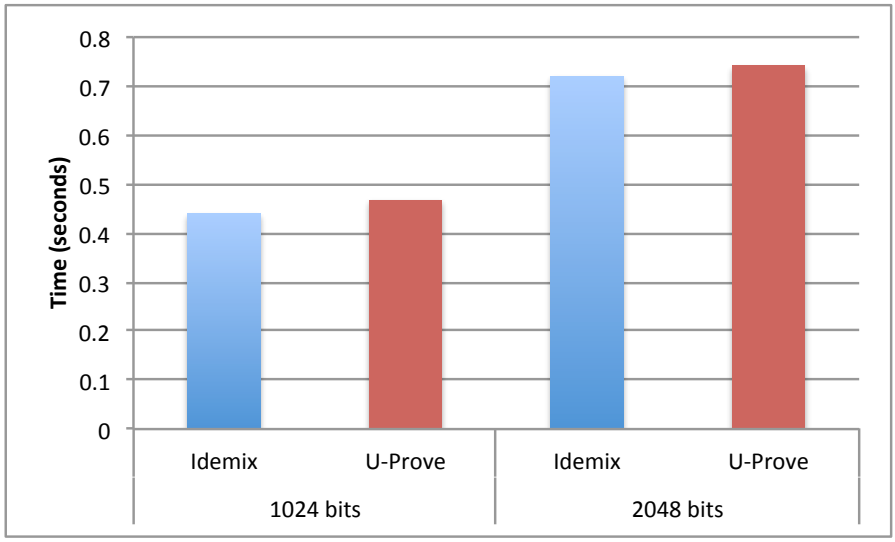

Figure 2.1 - Comparing the computational efficiency for simple issuance for Idemix (CL) and U-Prove (Brands)

2.1.1.2 "Advanced" issuance

In our tests, we compared the cost of using advanced issuance features to the simple issuance. The advanced issuance tests include additional presentation policies from the user, where the User must show some additional proofs in order to be issued a new credential. In Figure 2.2 we present a comparison between the simple and advanced forms of issuance for both Idemix and U-Prove using a key of 2048 bits size.

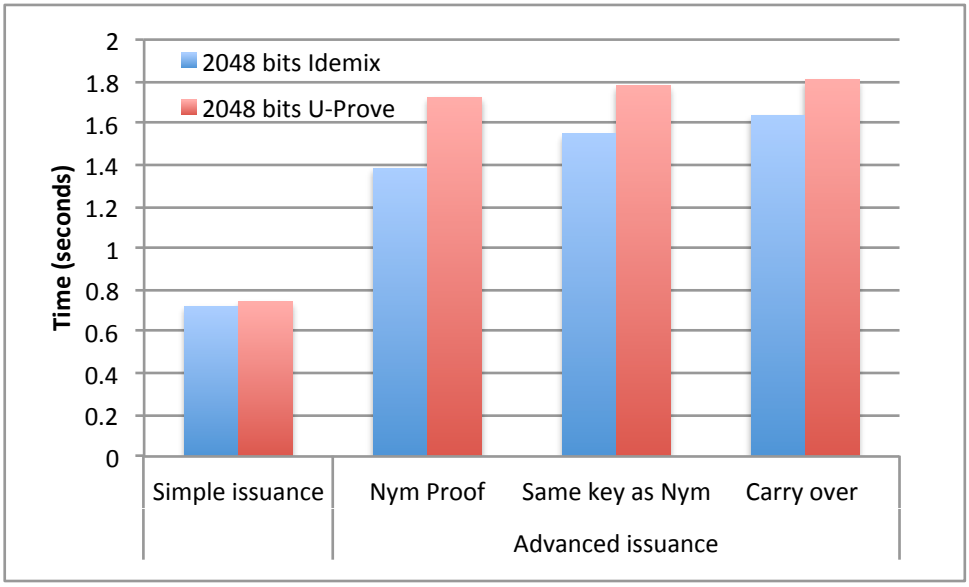

Figure 2.2 - Computational efficiency for simple and advanced issuance for Idemix and U-Prove

The advanced issuance includes three different issuance policies, namely one where a user needs to show possession of a scope-exclusive pseudonym (*Nym Proof*), where a credential is bound to the same secret key as the pseudonym (*Same key as Nym*), and issuance of a credential by *carrying over attributes* from another credential (*Carry Over*), without the Issuer knowing the value of the "carried over" attribute(s). One has to note that the issuance with carry-over attributes includes a proof of possession of a credential, from which the attribute needs to be carried over. In our case, we carried out a single attribute from a credential with 6 attributes.

As we can recognize from the figure, Idemix has an advantage over U-Prove in terms of computational time to complete the issuance. Furthermore, all three advanced features include a computational overhead on both technologies compared to the "simple" issuance, as an additional presentation needs to be done. However, the overhead of using "same-key binding" or "carry-over" is small and practically negligible for both technologies.

2.1.2 Presentation

Computational efficiency of presentation is certainly one of the most important criteria for comparing different Privacy-ABC technologies. In our tests, the computational efficiency for presentation is measured in second and includes both the time to do the proof (proving) and to verify the proof (verification). Figure 2.3 shows an overview of the efficiency for different types of presentation policies for the two instantiations of Privacy-ABC technologies (Idemix and U-Prove) in seconds.

The "basic" proof of a credential (*Cred*) is the most efficient one, as it is simpler and requires a fewer number of operations and building blocks. Each of the subsequent policies require some additional proof besides proof of credential, such as for instance a combination of a credential and a pseudonym (*Cred+Nym*), which has a clear overhead on the basic proof, as well as binding a credential and the pseudonym to the same secret key (*Cred+Nym+Key Binding*), which has an additional (but smaller) overhead on the previous one.

Privacy-ABCs also support a number of predicates, and we tested the equality proof of an attribute with a constant (*Equality with constant*) and with a different credential attribute (*Equality with attribute*). As the equality with the other attribute requires proof on two credentials, the overhead is taken by comparing it to the proof of two credentials (*2 Creds*). Similarly, we can compare the overhead of additional credentials on the computational efficiency of presentation by comparing the times for the policies requiring different number of credentials, namely *Cred*, *2 Creds* and *3 Creds*.

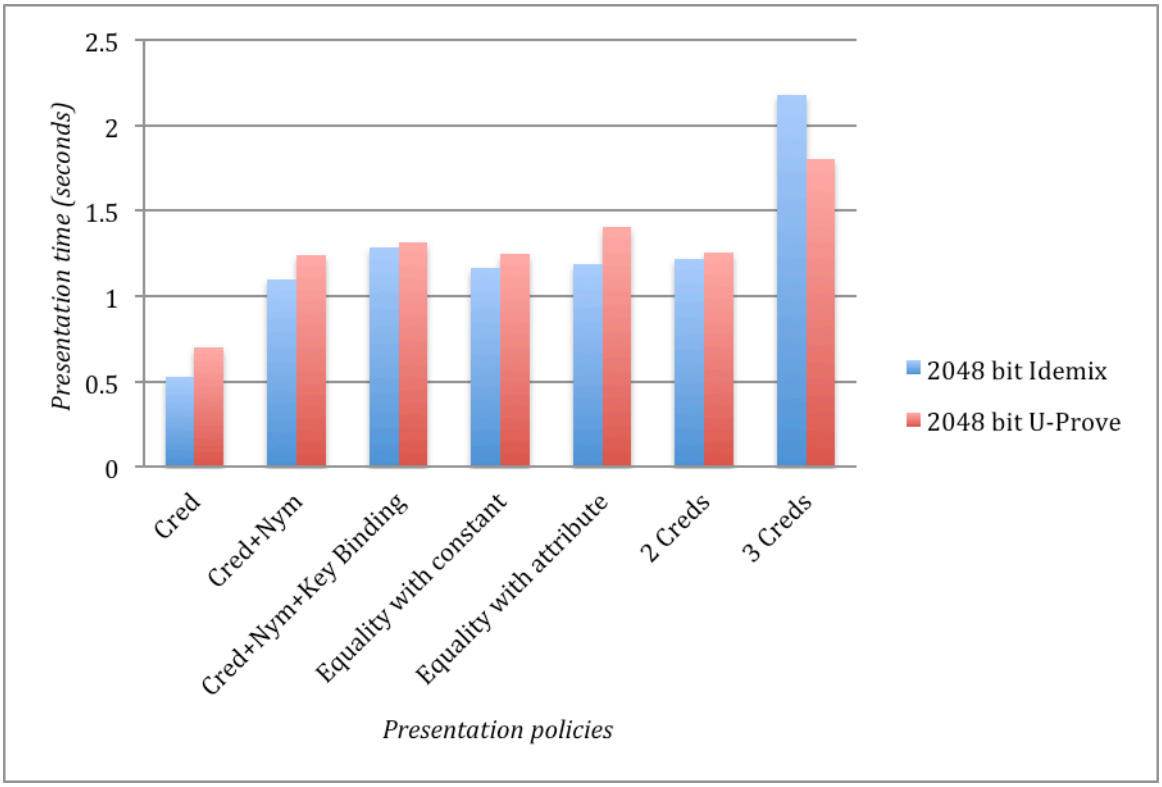

Figure 2.3 - Computational efficiency of presentation (proving + verification) for different presentation policies for the two instantiations of Privacy-ABC technologies

2.1.3 Inspection

Whether or not a presentation token is inspectable is defined in the presentation policy. In case a presentation token is inspectable, the User has to perform additional computation on her side, to verifiably encrypt the inspectable attribute. This has an added computational complexity, thereby reducing the computational efficiency for presentation. In Figure 2.4 we can clearly see the computational overhead of inspection on presentation.

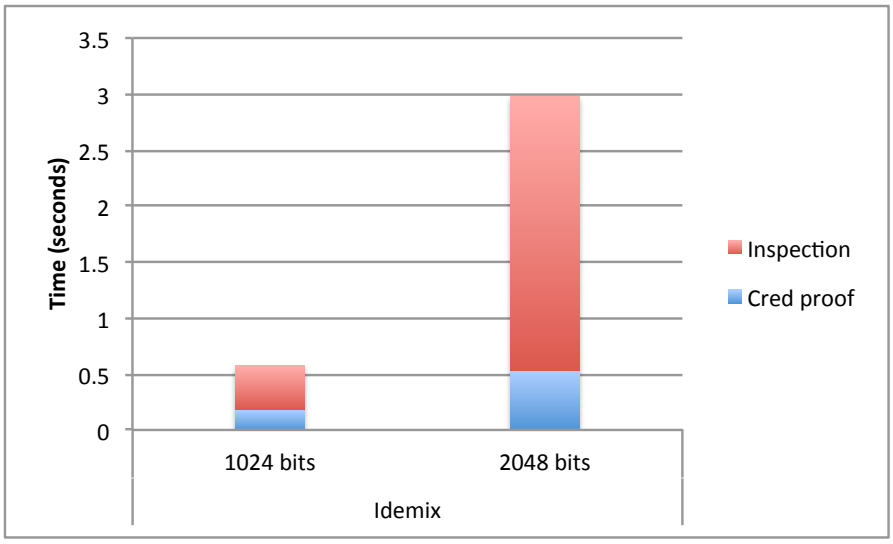

Figure 2.4 - Overhead of the inspection on the computational efficiency of presentation

However, the number of inspectable attributes has also an impact on the computational efficiency of the presentation. This is clearly shown in Figure 2.5 and the explanation behind this is that for each of the inspectable attributes, the User has to perform a verifiable encryption, whereas the Verifier needs to verify each of them separately. As a side note, this has probably the good effect that one would not chose to have inspection unless it is really necessary, as this would result in a longer time to do presentation.

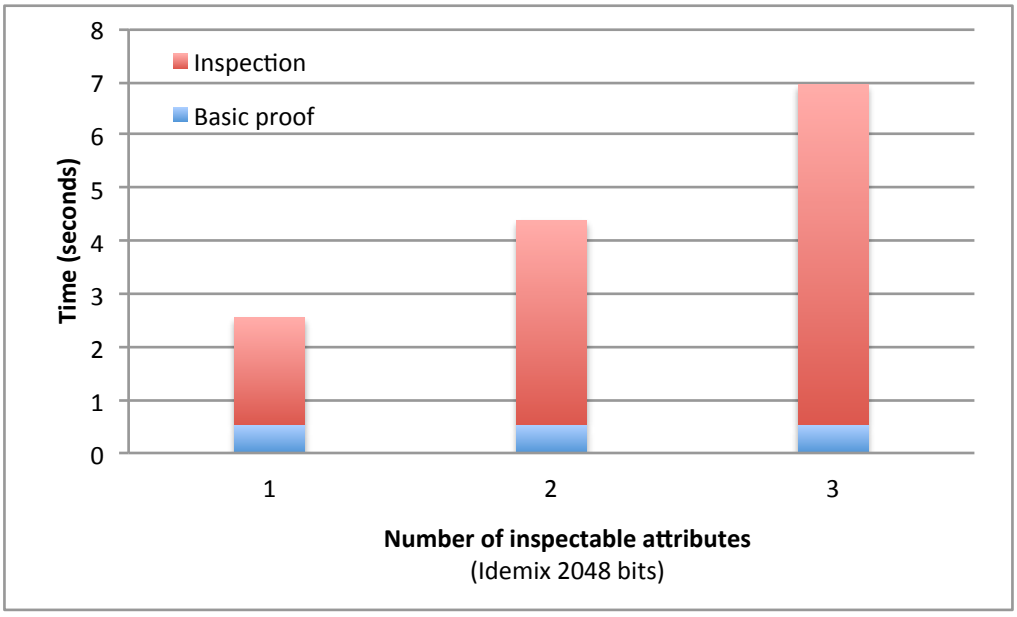

Figure 2.5 - The impact of the number of inspectable attributes on the computational efficiency of presentation

2.1.4 Revocation

Similar to the inspection, performing a non-revocation proof has a direct impact in the computational efficiency of presentation. Since both instantiations of Privacy-ABC technologies shared the same revocation technology, the overhead is the same. In Figure 2.6 we show the overhead of doing a non-revocation proof by comparing it with the efficiency of the same presentation policy without non-revocation proof (in this case, for U-Prove, but the overhead is the same for both). Clearly the non-revocation proof is costly in both cases, and the change in the key length (from 1024 bits to 2048 bits) has also an impact on the efficiency of proving and verifying non-revocation.

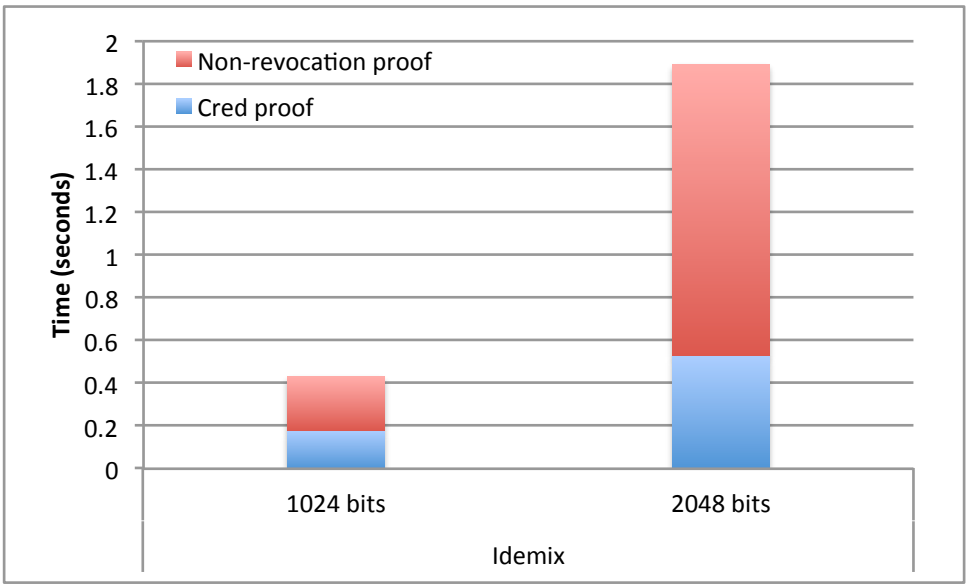

Figure 2.6 - Impact of non-revocation proof on the computational efficiency of presentation

2.2 Communication efficiency

2.2.1 Issuance

As issuance is an interactive protocol between the User (requesting for a credential issuance) and the Issuer (issuing the credentials), the comparison of the communication efficiency focuses exactly in comparing the difference in the two instantiations of:

- the number of issuance protocol rounds, where one round starts with a message from the User to the Issuer and may or not involve additional rounds, depending on the type of issuance and on the signature scheme used.
- the number of messages exchanged between the Issuer and the User, which is related to the number of rounds of communication; and
- the size of each message during each message exchange.

Finally, we take a look at the total size of the traffic the received and sent from the User.

As for the other measurements, the comparison takes into account also the impact of the security level.

2.2.1.1 Number of messages and number of communication rounds

During the simple issuance, there is only one round of communication between the Issuer and the User during the issuance using CL signature scheme, whereas for the Brands scheme, there is an additional round of communication. For the advanced issuance, both of the schemes require an additional round for presentation. The summarized number of messages exchanged between the Issuer and the User are presented in

Table 2.1.

Table 2.1 - Number of rounds and number of messages exchanged between the Issuer and the User during the issuance protocol

Type of issuance	Number of messages (rounds)	
	CL	Brands
Simple issuance (from scratch)	1 (1 round)	3 (2 rounds)
Advanced issuance forms (with key binding, carry over)	3 (2 rounds)	5 (3 rounds)

2.2.1.2 Total size of the incoming and outgoing messages

The messages exchanged between the Issuer and the User during the issuance protocol are XML-formatted messages, as defined in the ABC4Trust architecture deliverable [H2.1]. For this reason, the size of the messages contains not only the cryptographic part of the message, but also the additional structure of an XML document, resulting in some overhead in the overall message size.

The issuance protocol starts with a request from the User for an issuance policy from the Issuer. Depending on the type of issuance ("simple" vs. "advanced") and on the signature scheme used (Idemix vs. Brands), the number of issuance rounds, as well as the size of the messages exchanged during each round may be different. A summarized presentation of the total size of the incoming and outgoing (for the User) traffic (message sizes) for both signature schemes and different issuance scenarios is presented in Figure 2.7, where one can see also the overhead of the other forms of issuance on the communication efficiency, namely that advanced forms of issuance are less efficient in terms of communication size.

In general, the issuer messages are longer using CL signature scheme (Idemix) than using Brands (U-Prove), however, CL has one round of issuance less than Brands signature scheme for every type of issuance. We can also clearly notice that the number of attributes has a direct impact on the communication size (compare *Cred6Atts*, *Cred12Atts*, and *Cred24Atts*) for both signature schemes.

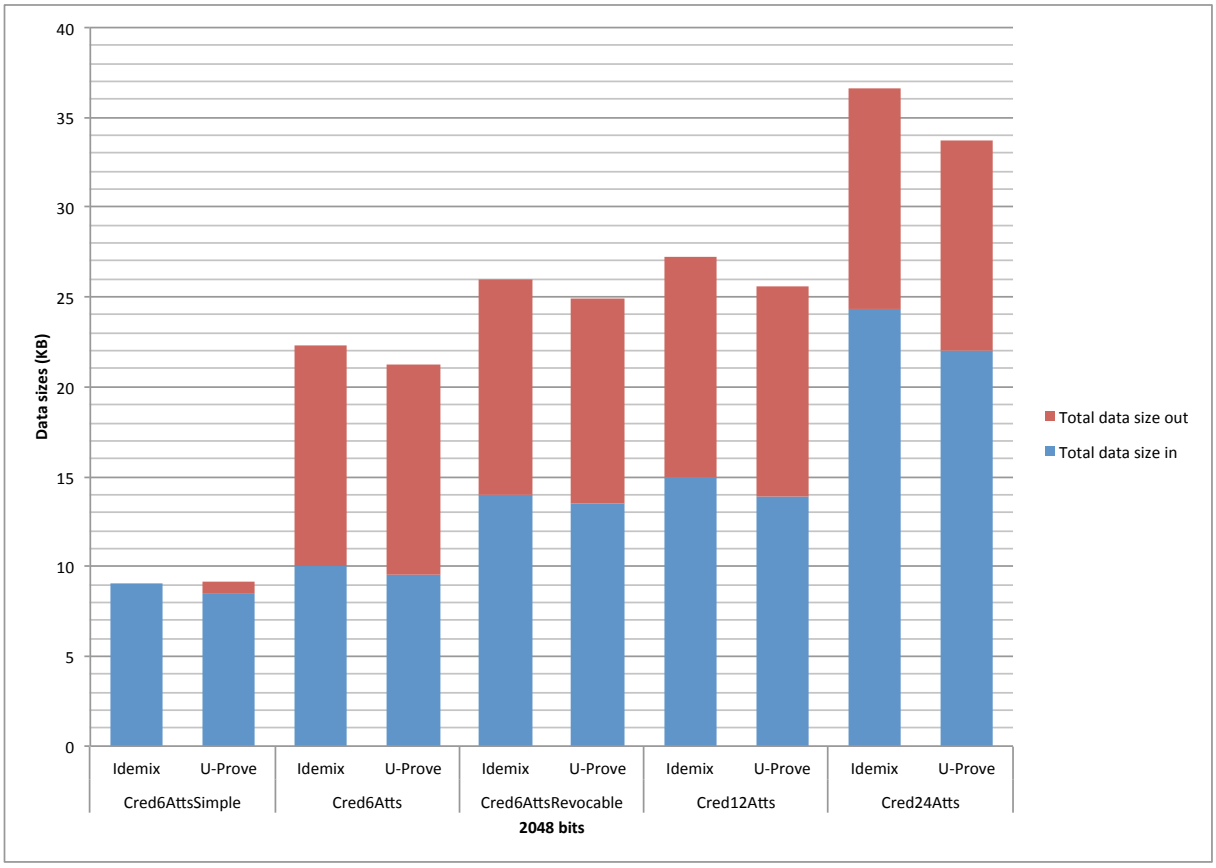

Figure 2.7 - Variation of the sizes of the incoming and outgoing data during different issuance types

2.2.2 Presentation

For presentation, we investigated the impact of different features used in the presentation phase on the size of the presentation tokens. A brief summary of the size of the presentation tokens for different types of presentation policies for the two Privacy-ABC technologies (Idemix and U-Prove) is presented in Figure 2.8. The results show varying sizes of presentation tokens for different presentation policies, starting from basic proof of a credential (*Cred*), combination of a credential and a pseudonym (*Cred+Nym*) as well as binding them to the same secret key (*Cred+Nym+Key Binding*), use of predicates (equality proof) of an attribute with a constant (*Equality with constant*) and with a different credential attribute (*Equality with attribute*), presentation of two credentials (*2Creds*), and presentation with three credentials (*3Creds*). On top of that, we investigated how the number of attributes impacts the size of the presentation token by testing presentation for credentials using respectively 6, 12 and 24 attributes (*Cred*, *Cred 12 Atts*, and *Cred 24 Atts*).

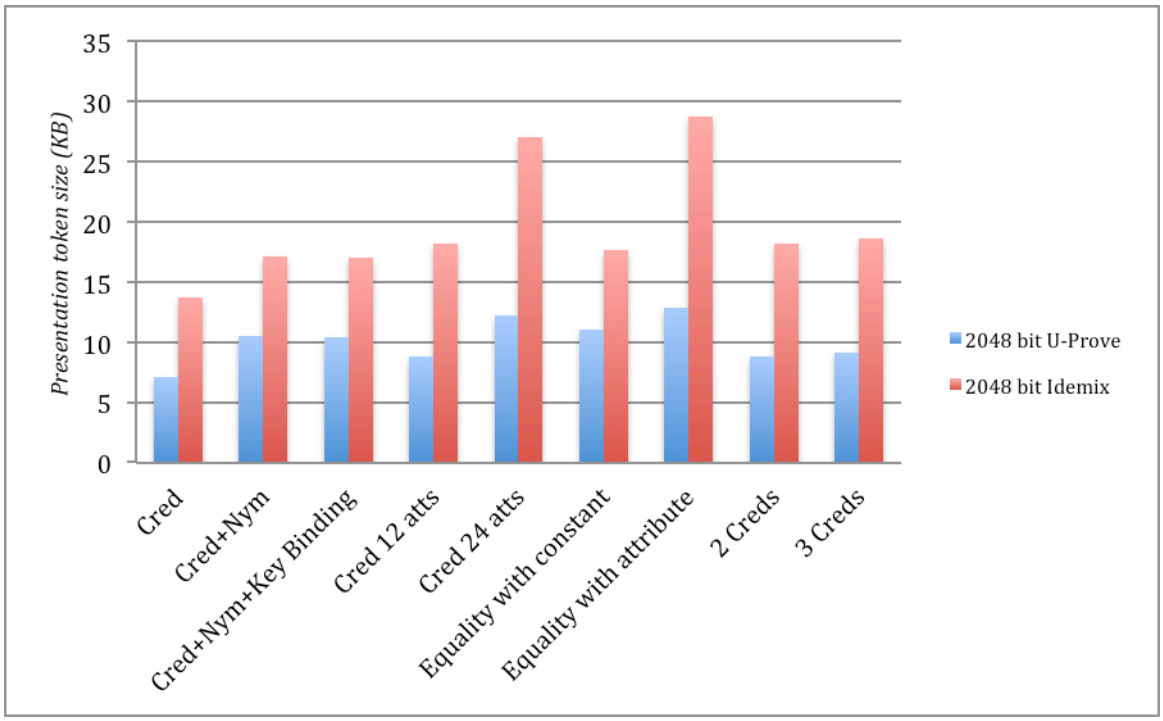

Figure 2.8 - Variation of the presentation token sizes for different features for both instantiations of Privacy-ABC technologies

2.2.3 Inspection

The impact of inspection on the presentation is of interest in order to see how the communication efficiency of the presentation changes on different inspection cases. A first view into this impact has been shown in Figure 2.9 for both key sizes of 1024 and 2048 bits for U-Prove, where a token for a policy which contains one inspectable attribute is compared to the same policy without inspection. The result is that the presentation token size changes for different key lengths, but the overhead of inspection remains constant (no impact from the key size).

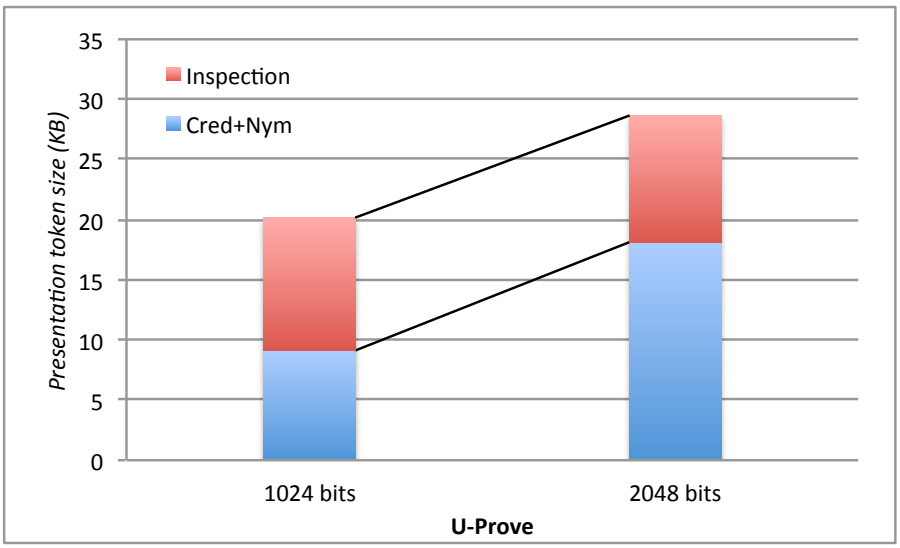

Figure 2.9 - The cost of inspection on the size of the presentation token for two different key sizes

Another important aspect of inspection that we evaluated was the impact of the number of inspectable attributes on the size of the presentation token. This is presented in Figure 2.10, where it is clearly visible that the number of attributes has a direct and significant impact on the size of the presentation token.

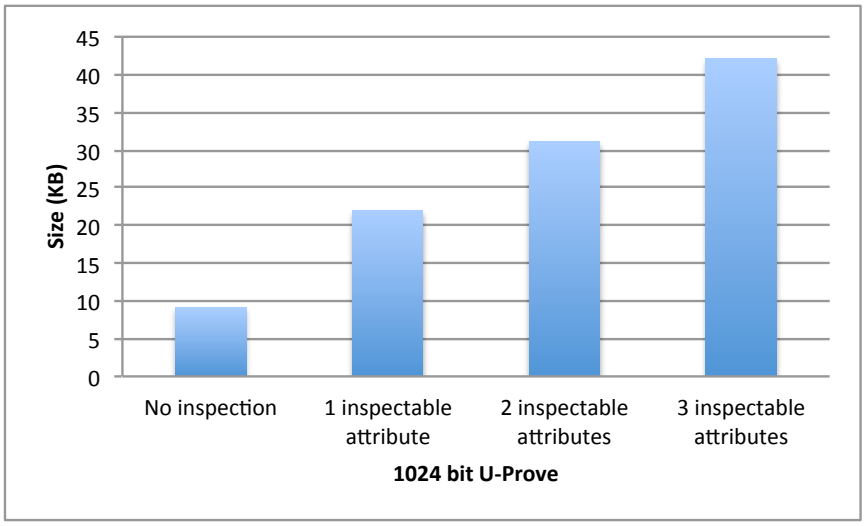

Figure 2.10 - The impact of the number of inspectable attributes on the computational efficiency of presentation

2.2.4 Revocation

The impact of the non-revocation proof on the size of the presentation token is presented in Figure 2.11 (for U-Prove). As we can see, non-revocation proof costs more than credential proof in terms of communication efficiency. On top of that, key length is also a factor that influences the both the efficiency of credential proof, but also the efficiency of non-revocation proof.

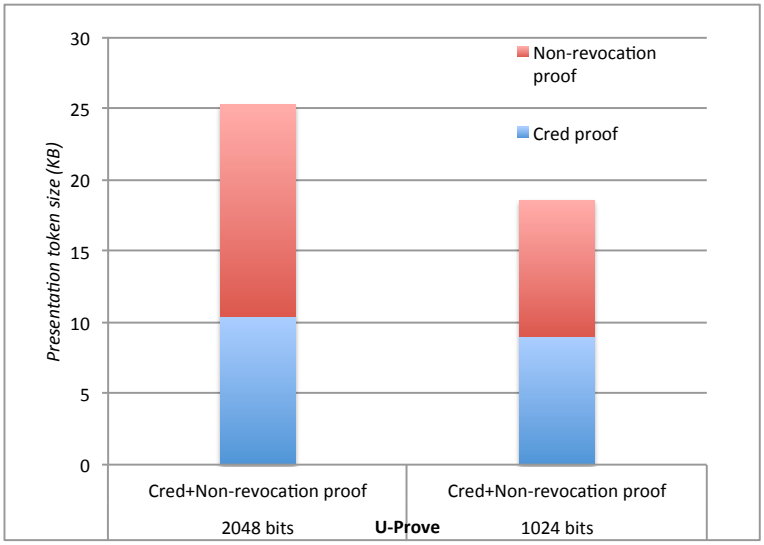

Figure 2.11 - Impact of the non-revocation proof on the presentation token size for two different key lengths

2.3 Storage efficiency

The last message from the Issuer to the User during the issuance protocol contains the cryptographic material, which the User needs in order to generate the full credential (Idemix), or may contain the credential itself (U-Prove). The size of the credentials may differ under different technologies and using different key sizes. Following are the comparison results for the issuance scenarios described in the previous sections of this chapter, namely simple issuance, issuance with key binding, issuance with carry-over attributes, and the issuance with different number of credentials.

2.3.1 The impact of the number of attributes on the credential size

Our experiments show that the number of attributes in a credential has a strong impact on the size of the credentials. Figure 2.12 shows a comparison of the two schemes for three credentials with different number of attributes, respectively 6 (*Cred6Atts*), 12 (*Cred12Atts*), and 24 (*Cred24Atts*). Both signature schemes are similarly efficient with regards to storage, with CL-based signature scheme (Idemix) being slightly more efficient. On the other hand, doubling the key size (from 1024 to 2048 bits) has a very small overhead on the credential size for both Privacy-ABC technologies.

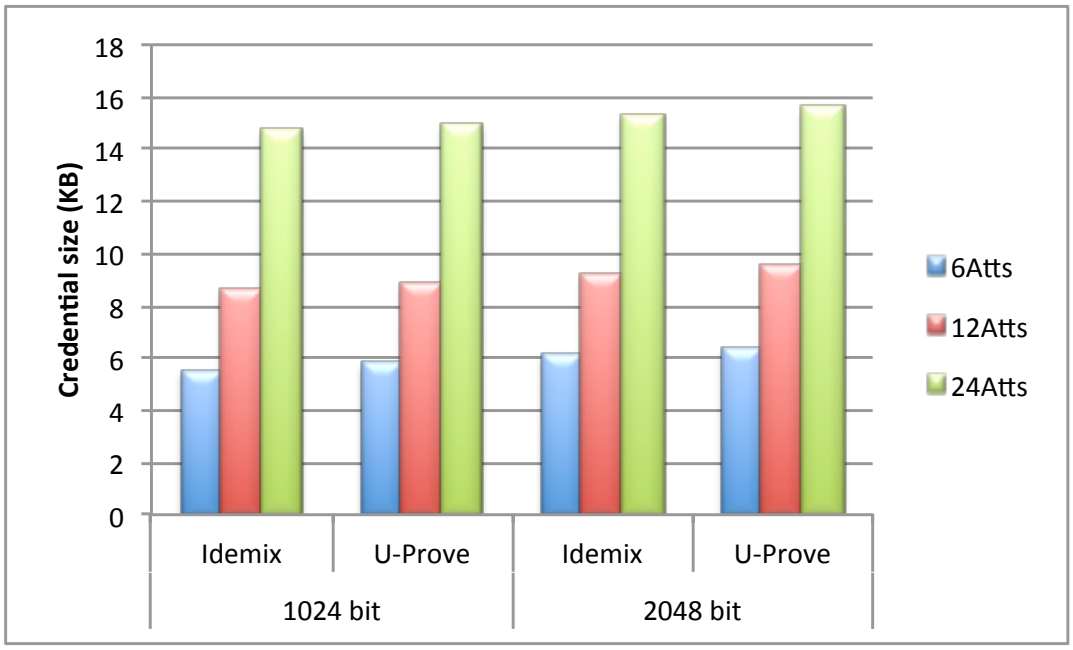

Figure 2.12 - The impact of the number of attributes on the credential size

2.3.2 Impact of the revocation information on the credential size

It was also interesting to see the additional storage required for the revocation information, as the User has to store this information for all revocable credentials, in order to prove, during the presentation, that this credential is not revoked. As both technologies tested used the same revocation technology, the impact of the revocation information on the overall storage efficiency was constant, as shown in Figure 2.13 (U-Prove), where we clearly distinguish the overhead of revocation information marked in red.

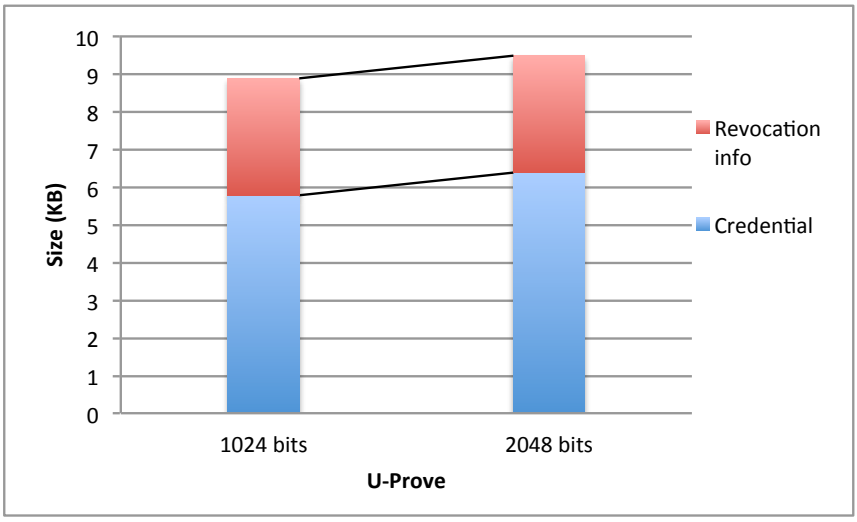

Figure 2.13 - Storage overhead of revocation information

3 Functionality

This chapter compares the two instantiations of Privacy-ABCs for their support of different privacy features for the user, but also for the additional implications that a certain technology may have on the users or other entities. In addition, we investigate the potential for the suitability of the technology to different scenarios, such as smart cards and other offline devices. The comparison results are presented in the tabular format that we introduced in Section 1.3, in correspondence with the criteria defined in (upcoming) deliverable D2.3.

3.1 Issuance

In this section, we compare the Issuance phase of two Privacy-ABC technologies used in the ABC4Trust project with regard to their functional features. The comparison is performed according to the function comparison criteria defined in the (upcoming) deliverable D2.3. The two comparison results for this phase concern the support of the two instantiations of Privacy-ABC technologies, namely Idemix and U-Prove, for the different types of issuance, and whether it is possible for them to (blindly) update attributes in existing credential via advance issuance methods. An example of credential update would be to increase the value of an attribute by 1 without learning about the value, but this could also be useful for revocation, when an attribute could be the validity date of the credential (but also possibly other applications).

Attribute	Value
Name	Supported types of issuance
ID	Iss-F1
Result	Both instantiations (based on Brands and CL signatures) used in the ABC4Trust project support: - Issuance from scratch - Advanced issuance of carry-over attributes - Issuance of key-bound credentials. Neither of the signature schemes currently supports: -Advanced issuance with self-claimed carry over attributes -Issuance of jointly random attributes
Comments	- It is technically possible to have "self-claimed carry-over attributes" for ABC4Trust implementation of U-Prove and Idemix but the upper layer interfaces are not provided yet. - Issuance of "jointly-random" attributes is not yet supported for both U-Prove and Idemix.

Attribute	Value
Name	Support for Credential Update
ID	Iss-F2
Result	This feature is not yet supported in either of the schemes used in ABC4Trust.
Comments	

3.2 Presentation

In this section, we compare the presentation-related functional features supported in the two instantiations of Privacy-ABC technologies. On top of that, some of the comparison results are present additional features, which could be useful for particular scenarios, where anonymous but accountable access is desired (such as controllable spending of credentials). As for the other comparisons, we do the comparison here as well according to the function comparison criteria defined in the (upcoming) D2.3. The six criteria defined for this phase concern different types of Privacy-ABCs feature supported by each technology (Pre-F1), the types of predicates that can be applied in a presentation proof (Pre-F2), whether it is possible to limit the number of usages of a credential (Pre-F3), types of supported Pseudonyms (Pre-F4), possibility of checking whether credentials and pseudonyms used in a presentation session are bound to the same key (Pre-F5), and the option to combine credentials from the same or different issuers in a presentation (Pre-F6).

Attribute	Value
Name	Privacy features for the User
ID	Pre-F1
Result	Features to compare: - Unlinkability: is achieved in both technologies but in different ways. In the case of U-Prove, the user receives a larger number of single-use U-Prove tokens. But if the user uses the same token in two different sessions, they will be linkable. In the case of Idemix, the process is different and the same credential can be shown multiple times without the concern of being linkable. -Untraceability: Both technologies offer untraceability, meaning that the issuance and presentation token cannot be linked. -Selective disclosure: For both U-Prove and Idemix, the attributes in the credential can be selectively disclosed. -Anonymous comparison over attributes (predicates over attributes): The ABC4Trust implementation of both U-Prove and Idemix support predicates over attributes.
Comments	

Predicates over attributes are also features of Privacy-ABCs. Using predicates, a user could perform mathematical-logical operations of her attribute values and prove, e.g. that their year of birth is before a certain (constant) date, or that two attributes from different credentials have the same (or different) value without revealing the attributes themselves. In the following table, we provide a list of currently supported predicates.

Attribute	Value
Name	Supported Predicate functions over attributes
ID	Pre-F2
Result	The ABC4Trust implementation of both U-Prove and Idemix support the following predicates over attributes: - equality of strings - equality of integers - equality of booleans - equality of times - equality of dates - inequality of strings - inequality of integers - inequality of booleans - inequality of times - inequality of dates

Comments	Additional predicates are being implemented and more may soon be supported. The current list only describes the predicates currently implemented in ABC4Trust.

In certain application scenarios, it may be useful to issue credentials which can be used a limited number of times. Similar to the e-cash concept, these credentials would become then useless if the maximum usage number allowed is reached. This could be useful in certain marketing campaigns where customers are given anonymous "coupons" with a certain usage limit. This is not defined as a Privacy-ABC feature, but it could be useful in practice.

Attribute	Value
Name	Controllable "Spending"
ID	Pre-F3
Result	It is not possible neither for U-Prove nor Idemix to strictly restrict the number of presentation that a User can make with a credential. However, in both technologies, a credential can be blocked from further usage by the mean of revocation.
Comments	As a side note, in the case of U-Prove, since the credentials work based on tokens, the user will face the danger of linkability of sessions when a token is used more than one time. As a result, it is possible in some cases to discourage the users from using a credential more than the number of issued tokens.

Users may chose to use certain pseudonyms, or may be required to do so by the Verifiers. In any case, the ABC4Trust architecture defines three types of pseudonyms. Here we compare whether both instantiations of Privacy-ABC technologies support the different types of pseudonyms.

Attribute	Value
Name	Supported types of Pseudonyms
ID	Pre-F4
Result	ABC4Trust implementation of both U-Prove and Idemix support the following types of pseudonyms: - *verifiable pseudonyms* - *certified pseudonyms,* and - *scope-exclusive pseudonyms.*
Comments	*Further information about these pseudonyms are available in Page 19 of H2.1.*

Privacy-ABCs can be bound to a secret key, thereby increasing the security whenever authentication of users would require their proof of knowledge of the secret key, similar to a public key certificate. Furthermore, it may be possible to bind different credentials to the same secret key during the presentation.

Attribute	Value
Name	Support for Key Binding
ID	Pre-F5
Result	Both U-Prove and Idemix implemented in ABC4Trust support presentation where credentials or pseudonyms can be bound to a user-secret key. In addition, both technologies support binding of different credentials/pseudonyms to the same secret key during presentation.
Comments	

Users may possess credentials from different issuers. It is convenient for them to be able to do proofs by combining credentials from different issuers. Therefore, it is interesting and useful to understand whether any of the given instantiation of Privacy-ABC technologies has a limitation in the support for doing presentation using credentials from different issuers.

Attribute	Value
Name	Supported Issuers in Presentation
ID	Pre-F6
Result	Both ABC4Trust U-Prove and Idemix can use credentials from the same or different issuers in the same

	presentation proof.
Comments	

3.3 Inspection

In this section, we compare the Inspection phase of two Privacy-ABC technologies used in the ABC4Trust project with regard to their functional features. As inspection may be a desired feature of Privacy-ABCs, where a separate trusted entity (the Inspector) can "inspect" (reveal the identity of) an anonymous user (presentation token), it is important to understand whether the two instantiations of Privacy-ABC technologies support the inspection of presentation tokens. In ABC4Trust implementation, there is currently only one inspection scheme implemented, which both instantiations of Privacy-ABC technologies share. Hence, this is no comparison between the two schemes, but rather a benchmark on the implementation.

Attribute	Value
Name	Support for inspection
ID	Ins-F1
Result	Both U-Prove and Idemix technologies implemented in ABC4Trust support inspection.

3.4 Revocation

In this section, we provide the benchmarking results for the functionality analysis of the revocation strategy implemented in ABC4Trust. Since both privacy-ABCs use the same revocation scheme, we cannot provide a comparison, but a benchmark. The 10 criteria for this phase concern the online connectivity of parties (Rev-F1), support for different types of revocation (Rev-F2), possibility to work with offline users (Rev-F3), support for immediate revocation (Rev-F4), the possibility to distribute the scheme over multiple entities (Rev-F5), privacy threats (Rev-F6), compatibility with different Privacy-ABCs (Rev-F7), frequency of required communication between the parties (Rev-F8 and F10), and confidentiality of the revocation information published by the Revocation Authority (Rev-F9).

The Revocation Authority can serve as a central point for different processes in the lifecycle of Privacy-ABCs. For one, during the issuance, it can collaborate with the Issuer in order to generate the necessary information for proving non-revocation for a given credentials. Furthermore, during presentation, depending on the revocation scheme, users and/or verifiers may need to connect to the Revocation Authority to get the latest updates about the revoked/unrevoked credentials. This is especially an issue for the User, in cases when it is desired to achieve an offline system using Privacy-ABCs on devices which are not connected to the Internet (such as smart cards), but may also delay the process of presentation. Finally, the more frequent the connectivity is desired, the more important the role of the Revocation Authority becomes.

Attribute	Value
Name	Connectivity requirements to Revocation Authority
ID	Rev-F1
Result	-*Requirement for contacting the RA during presentation (User)* – The User can contact the Revocation Authority during each presentation, on a regular basis (using the expiry date of the RI) or whenever the current non-revocation evidence is not up-to-date with the revocation information specified by the Verifier. In the ABC4Trust implementation, the RA is contacted each time the User creates a proof that her credential was not revoked. -*Verifier's Active Connectivity with the Revocation Authority* - This criterion depends on the Verifier's requirements on how often the revocation information should be checked/updated.

	-Issuer's connectivity with the Revocation Authority – The Issuer requires connection with the Revocation Authority upon revoking a credential.
Comments	

ABC4Trust defines two types of revocations: one is local and can be maintained by a Verifier, which is called Verifier-driven revocation, and the other type is "global" in its impact, in the sense that when the Issuer revokes a credential, it becomes permanently useless for any Verifier. Hence, we want to find out whether both revocation types are currently supported.

Attribute	Value
Name	Support for Issuer- and Verifier-driven revocation
ID	Rev-F2
Result	The implementation by ABC4Trust supports the Issuer-driven revocation but not Verifier-driven. However, it is feasible to add this feature if needed.
Comments	

As discussed above, it may be useful (or necessary) to have an offline presentation supported, in the sense that the User only needs to contact the Verifier in order to do the presentation. This is especially important if one decides to implement Privacy-ABCs on smart cards, or on other devices without connectivity. In addition, the additional connectivity may always create an additional delay during the issuance in terms of time required to perform presentation.

Attribute	Value
Name	Offline Usage
ID	Rev-F3
Result	The revocation scheme implemented in ABC4Trust does not offer Offline Usage. Therefore, the user might need to connect the Revocation Authority to receive some updates in order to prove the validity of her credentials.
Comments	During presentation, the Verifier specifies the version of the revocation information, which the user needs to prove non-revocation against. In case the Verifier has a newer revocation information than the User can prove, that the User needs to connect to the Revocation Authority to fetch the latest updates.

Revocation is not easy for Privacy-ABCs, as it needs to preserve unlinkability. Therefore, the different strategies proposed to achieve revocation sometimes hard-code validity time of the credential and the credential cannot be revoked before this expiration time comes. For our implementation, we provide the following table, explaining the support of the chosen revocation strategy for immediate revocation.

Attribute	Value
Name	Support for immediate revocation
ID	Rev-F4
Result	In principle, revocation of credentials can happen immediately and the chosen revocation scheme achieves this. It is then up to the Verifiers to decide how often they would like to check for the latest updates with regard to revocation information. Immediate revocation is only achieved when the Verifiers fetch the latest Revocation Information upon any changes by the RA, and enforce them immediately.
Comments	

To increase both efficiency and security of the system providing the revocation information, it is useful to be able to distribute the service of the Revocation Authority to different entities.

Attribute	Value
Name	Scheme distribution
ID	Rev-F5
Result	This feature is not implemented but theoretically it is possible, if this makes sense. Outsourcing the revocation operation would involve distributing the secret key of the RA. The RI is public so can be distributed by different sub-entities. The revocation handles can also be pre-generated. All entities of one RA, however, should share a single database.

Comments	

Revocation and proving non-revocation are some of the challenges, which come with Privacy-ABCs. In some schemes, a revocable credential becomes linkable, violating the privacy of the Users, known as *backward-linkability*. In general, the User must remain anonymous and unlinkable not only to the Verifier, but also towards the Revocation Authority when they update their non-revocation information.

Attribute	Value
Name	Supported privacy features
ID	Rev-F6
Result	- *Backward-unlinkability after revocation* – In the implemented revocation mechanism, it is not possible to link or discover previous sessions of the holder of the revoked credential. - *Unlinkability of the users during non-revocation update* – The Revocation Information is publicly available and can be downloaded anonymously. As a result, there is no risk for the User if the RA and the Verifier collude. - *Anonymous update of non-revocation status of own credentials* – The update of non-revocation information is done locally by the user after downloading publicly available revocation information.
Comments	

As mentioned before, certain revocation technologies may require a more frequent connectivity of the User/Verifier to the Revocation Authority. It is therefore useful to make this fact transparent in order to be able to compare the chosen revocation strategy against potentially other revocation strategies.

Attribute	Value
Name	Frequency of User's contact with the Revocation Authority
ID	Rev-P8
Result	The User can contact the Revocation Authority during each presentation, on a regular basis (using the expiry date of the RI) or whenever the current non-revocation evidence is not up-to-date with the revocation information specified by the Verifier. In the ABC4Trust implementation, the RA is contacted each time the User creates a proof that his credential was not revoked. In general, this can be implemented/decided depending on the requirements of the application using the revocation strategy.
Comments	

Attribute	Value
Name	Frequency of Verifier's contact with the Revocation Authority
ID	Rev-F10
Result	This criterion depends on the Verifier's requirements on how often the revocation information should be checked/updated. If the Verifier would like to have the latest Revocation Information, it has to contact the RA upon revocation of every credential.
Comments	

4 Security assurance

In this chapter, we provide an overview of the security-related issues with Privacy-ABC technologies and how ABC4Trust addresses some of the identified potential security issues. Furthermore, we discuss about the security of the underlying building blocks used in the implementation compared in the previous chapters, and discuss the security of the individual instantiations described. We will therefore divide the chapter into two main parts, one describing the general security aspects that are common to the ABC4Trust architecture, and the other one focusing more on the individual security aspects of the two compared technologies.

4.1 General security aspects on the implementation of the architecture

In the following sections, we provide an overview of certain security-related aspects of Privacy-ABCs, targeting particular building blocks for certain features, but for which a comparison of the two instantiations of Privacy-ABCs is not provided, as these are common for both. We will discuss certain security aspects for the assurance of the security of the different lifecycle stages of Privacy-ABCs, such as revocation and inspection, and measures to assure the security of the user's private information.

4.1.1 Integrity and authenticity of the Revocation Information

It is important that the revocation information, which the Verifier verifies against, is authentic and preserves its integrity, among other things. Otherwise, breaking any of these two goals would result in the loss of the goals of the technology, where users would be wrongly granted or denied access. For this purpose, a system using the revocation technology from ABC4Trust must provide clear answer to the question below, which is one of the benchmarking criteria from the upcoming deliverable D2.3 "Benchmarking Criteria".

Survey Question: *Which mechanisms have been implemented by the Privacy-ABC (revocation) technology to protect the Revocation Information's integrity and authenticity?*

Attribute	Value
Name	Mechanisms used by the Privacy-ABC technology to guarantee the integrity and authenticity of the Revocation Information.
ID	Rev-S1
Result	In ABC4trust is currently implemented only the issuer-driven revocation. The Verifier uses an authenticated channel to the Revocation Authority responsible for publishing the revocation information i.e., in Patras Pilot the Verifier connects to the Revocation Authority via an SSL/TLS channel.
Comments	Implementation evidence can be found in D2.1

4.1.2 Support for private key revocation

There may be different strategies to revoke a Privacy-ABC. In some scenarios (and also in ABC4Trust implementation), a special number called a "revocation handle" is a special attribute in the credential, which enables its revocation. However, in cases where a user may have different credentials bound to the same private key, it may be more efficient to simply revoke the private key to which these credentials are bound to than revoking each revocation handle from each credential separately. For this reason, we have investigated whether the revocation strategy implemented in ABC4Trust supports such key-revocation, as defined in the following.

Survey Question: *Is there a process to request the end-user's private key revocation (along with related Credentials)?*

Attribute	Value
Name	Support of key-revocation
ID	Rev-S2
Result	Private key revocation is not supported. Only the revocation of credentials or specific attributes is supported by the technologies.
Comments	Revocation is done by revoking the revocation handle, which is issued to the user together with the credential. Implementation evidence can be found in [H2.1].

4.1.3 Confidentiality of the central revocation information

Like in traditional PKI systems, such as X.509 certificates, a central authority (Revocation Authority) keeps a database of revoked credentials, and updates it periodically. Verifiers can then download this information and ask users to show that they are not revoked. Depending on the chosen revocation strategy, the access to this database may be open to any party, meaning that it is public and access to this information would not give an adversary any additional benefit to reduce the security goals of the system, or it may be confidential, in which case each user would get a different, prepared information that proves that they are not revoked. These two strategies require different mitigation mechanisms and it is therefore important to distinguish this fact in order to assess the appropriate mechanism to assure the desired security of this information, and to assess the potential danger on the user's privacy and security.

The answer to the survey question given below provides information regarding the access level to revocation handles.

Survey Question: *Which of the following access restrictions apply to revocation handles:*

- *Public or private,*
- *Learnt by RA or by Verifier only.*

Attribute	Value
Name	Access to Revocation Handles.
ID	Rev-S4
Result	The list of revoked Revocation Handles is contained (cryptographically *accumulated*) within the Revocation Information, which is public information, to which any part can have access to (both Verifiers and Users).
Comments	The Revocation Authority publishes cryptographic information called an accumulator, which maps all the unrevoked revocation handles into a single value. This is the value to which all the parties have access to, and it reveals no more information to the other entities about the individual revoked handles that are "accumulated".

4.1.4 Preventive measures against authority misuse of the Inspector

Inspection is an optional feature of Privacy-ABCs, which can increase the accountability of the users participating in a system using Privacy-ABC technologies. However, proper care should be taken to assure that the Inspector only performs inspection when strictly defined conditions are met, which is defined in the inspection grounds. This may require that one trusts the Inspector to strictly comply with the predefined inspection grounds for inspection, but additional measures can be taken to prevent misuse of the inspection capability.

This security metric aims to provide information regarding how authority misuse is prevented from the person in charge of inspection. It considers the measures applied to eliminate the ways of misusing User's data, and is related to the design of the respective technology.

Attribute	Value
Name	Technical Preventive measures against authority misuse.
ID	Ins-S1
Result	The measures applied to eliminate the ways of misusing User's data are related to the design of the respective technology. The Inspector is trusted not only by the Verifier to assist by providing the required information in case of abuse, but also by the User not to uncover identities unnecessarily. At the time of creating the presentation token the User is aware of the inspection grounds related to specific attributes, of the identity of the Inspector who will be contacted to reveal information, as well as of the information that will be revealed in case of evidence for inspection grounds. A number of technical preventive mechanisms were discussed in the course of ABC4Trust, such as having k-out-of-n inspector keys revoking together (and avoiding a single inspector from being able to do inspection), but this was not implemented in the course of the project.
Comments	Implementation evidence can be found in D2.1

4.2 Security of the basic schemes

This part of the chapter deals with the security comparison of the two instantiations of Privacy-ABC technologies in terms of security proofs for the basic scheme used in each instantiation. For this purpose, we aim to compare the two instantiations based on the schemes they are based on, and whether or not the implementation is made with security reductions or not.

The proposed security metrics related to the security proofs and assumptions aim to provide information regarding whether security proofs are given and under which assumptions. They should state whether the security proofs and assumptions are (i) information theoretic, (ii) computational or (iii) without security reduction/proof.

Attribute	Value
Name	Security proofs and assumptions
Result	Regarding assumptions we don't have any security reductions. However, the schemes are still based on other schemes that in most cases do have a security reduction. We don't have a security reduction for the full scheme since the composition of various secure schemes does not imply security of the composed scheme, etc.
Comments	

5 References

[LKS12] J. Luna, I. Krontiris, and N. Suri, “Privacy-by-Design Based on Quantitative Threat Modeling” In Proc. of the IEEE International Conference on Risks and Security of Internet and Systems. 2012.

[Brands00] S. A. Brands. Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy. *The MIT Press*, August 2000, ISBN 0-262-02491-8, first edition.

[CL03] J. Camenisch, A. Lysyanskaya. A Digital Signature Scheme with Efficient Protocols. In: *Proceedings of the 3rd international conference on Security in communication networks*. Springer-Verlag, 2002.

[H4.1] H. Guldage, J. Nielsen. D4.1 Initial Reference Implementation. ABC4Trust Project deliverable, June 2012.

[Smart11] Smart N. (ed.). “ECRYPT II Yearly Report on Algorithms and Keysizes (2010-2011)”. Katholieke Universiteit Leuven (KUL). Deliverable SPA-17. rob June, 2011. Online: http://www.ecrypt.eu.org/documents/D.SPA.17.pdf

[H2.1] J. Camenisch, I. Krontiris, A. Lehmann, G. Neven, C. Paquin, K. Rannenberg. H2.1 - ABC4Trust Architecture for Developers. In: *Ioannis Krontiris* (ed.), 2012. Public ABC4Trust heartbeat.

[CS03] J. Camenisch, V. Shoup. Practical Verifiable Encryption and Decryption of Discrete Logarithms. In: *Advances in Cryptology - CRYPTO 2003,* pp 126-144, ISBN 978-3-540-45146-4. 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003. Springer Berlin Heidelberg.

[D2.3] F. Veseli, J. Luna, H. Ghani, T. Vateva-Gurova, H. Zwingelberg, K. Storf, F. Bieker, D. Deibler, M. Hansen. “D2.3 Benchmarking Criteria”, in *Fatbardh Veseli* (ed.), project deliverable, 2014.