VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ

Ústav telekomunikací

Ing. Petr Dzurenda

Cryptographic protection of digital identity

Kryptografická ochrana digitální identity

ZKRÁCENÁ VERZE PH.D. THESIS

Obor:	Teleinformatika
Školitel:	doc. Ing. Jan Hajný, Ph.D.
Oponenti:	
Datum obhajoby:	

KEYWORDS

Cryptography, Privacy, Group Signatures, Attribute-Based Credentials, Anonymity, Smart Cards, Authentication, Elliptic Curves, Bilinear Pairing, Constrained Devices

KLÍČOVÁ SLOVA

Kryptografie, Soukromí, Skupinové Podpisy, Atributová Pověření, Anonymita, Čipové Karty, Autentizace, Eliptické Křivky, Bilineární Párování, Omezená Zařízení

MÍSTO ULOŽENÍ PRÁCE

Disertační práce je k dispozici na Vědeckém oddělení děkanátu FEKT VUT v Brně, Technická 10, Brno, 616 00.

ISBN 80-214-
ISSN 1213-4198

CONTENTS

INTRODUCTION

We live in the Information Age. The time when the ownership of a computer or the Internet access was just a privilege for rich people only has already gone. Current "*smart*" devices are permanently connected to the Internet and provide us a great deal of different cloud services. Smart devices, the Internet and many of cloud solutions form our daily life. The Internet is no longer used just to search for information. For example, new smart televisions (TV) allow us to watch on-line streaming videos and record movies which are stored to the cloud. Smart phones are not used just to make calls. For instance, they can be used for sport activities (as a personal trainer, e.g. Endomondo), listening to music (on-line streaming music services such as Spotify), chatting with friends, and living our social life (on Facebook, Google, WhatsApp etc.). Our data are always available thanks to services such as Dropbox, Google Drive, OneDrive, and iCloud. Moreover, we never get lost, since our smart phone is equipped with Global Positioning System (GPS). At present, there is almost nobody who misses an account held under Facebook, Google, Amazon or Apple. These internet giants collect our data and profile us. They may do that for improving and optimising the services or for better understanding our behaviour and preferences [18]. But how can we be sure that these data are not collected to track us and sell our profile?

Smart grids, smart metering or smart cities are the current terms as well as the Internet of Things (IoT). Electronic devices start to communicate with each other without human interaction, they send (or exchange) many of user data through the Internet. New data published by Juniper Research [63] show that the development of smart grids linked to the smart cities will result in citizens saving $14 billion per annum in energy bills by 2022. Most of the big cities (such as London, Brussels, Barcelona and many others) apply the low emission zones at the city centers to minimize the pollution. In this scenario, only registered cars have access to the center. The bicycle– , scooter– or even car–sharing is an actual service in the most modern cities. In many cases, just a pre-installed application in a smart phone with Bluetooth or Near Field Communication (NFC) technology support is required to unlock and use these vehicles. The public transportation system gets more and more integrated, and, at the same time, supports smart cards with prepaid fare. Countries issue electronic IDentity (eID) cards as in the case of Germany [61] and Czechia [55]. These may lead to people tracking anywhere at any time.

The current systems are required to provide standard security properties. The data has to be protected against modification (data *integrity*) and eavesdropping (data *confidentiality*). The data recipient has to be sure that the data was sent by a known sender (*authentication*) and the sender cannot deny having sent the data (*non-repudiation*). Unfortunately, the standard systems use the identity-based authentication approach, where a user must identity himself at first. To do that, he sends his unique identifier (which is associated with his real identity), and then, he proves the proclaimed identity using the corresponding private key. This security context has a big impact on user's privacy, since the user identity is always disclosed. The verifier or service provider can profile the user, track his movement and behavior. Therefore, the standard security requirements are insufficient. In many scenarios user identification is not necessary and a service provider needs to know only whether a user has access to the required service (i.e., holds a valid ticket) or not. No other personal information is needed. The requirements on development of more privacy-friendly applications have been already demanded since 2011 by United States (US) [69] and European Union (EU) [50] institutions.

Especially recently, the European Commission has adopted many new regulations and strategies with close relation to the user privacy. For example, the General Data Protection Regulation (GDPR) [40] is the regulation of EU law from 2016. In particular, GDPR aims primarily on data protection and privacy. Thanks to this regulation, users gain higher control over their data. The European Network and Information Security Agency (ENISA) demands on privacy-preserving features of European eID [54]. The European Strategy on Cooperative Intelligent Transport Systems (C-ITS) [39] aims to improve road safety, traffic efficiency and comfort of driving, by helping drivers to take the right decisions and adapt their route to the traffic situation. In this context, C-ITS assumes that there is a communication between vehicles and a transport infrastructure. Drivers are exchanging information about their locations and other important data. In the same time, C-ITS must protect the location privacy of drivers to avoid their tracking.

Modern cryptographic constructions may prevent privacy leaks in current scenarios. For example, group and ring signatures significantly increase user's privacy. Users only prove their membership in the specific group, while their identity remains hidden. Furthermore, Attribute-Based Credential (ABC) schemes allow users to prove the possession of personal attributes, while no more additional information or user's identity is revealed. Therefore, these schemes are suitable for privacy-friendly systems. Unfortunately, the schemes are usually more computationally expensive compared to standard signature and authentication schemes, since they use more arithmetic operations. In particular, the modular exponentiation and bilinear pairing operations are widely used and directly affect the scheme efficiency, and, hence, its practical usability.

1 THESIS OBJECTIVES

The general objective of this thesis is to design novel privacy-enhancing cryptographic schemes for practical use in current Information and Communication Technology (ICT) scenarios, especially in access control systems, but also in data collection and notification systems. The current systems use identity-based authentication (and authorization) approaches to control the access to services. This affects directly user privacy and digital identity protection. Therefore, we are mainly interested in developing novel privacy-friendly cryptographic schemes that address these shortcomings and threats. First of all, we require that the scheme provides both *security* and *privacy* properties. The scheme must by **provably secure**, i.e., the scheme security holds under cryptographic hardness assumptions, and meets both **completeness** and **soundness** properties. Furthermore, we are going to involve advanced cryptographic primitives, such as **zero-knowledge** protocols, to control the amount of released sensitive information during the authentication process. Besides the *security* properties, the scheme has to meet at least the following *privacy* properties:

- **Anonymity**: the user's identity remains hidden during the authentication process. Hence, there is no privacy threats for honest users. The verifier may only check, whether the user is authorized to access the service or not.

- **Unlinkability**: all transactions (sessions) of a single user are mutually unlinkable and completely indistinguishable from the transactions of other users. It prevents linking individual sessions together and profiling users.

- **Untraceability**: the proofs generated by users are randomized, hence, not even the issuer is able to track issued credentials, i.e., users' behaviour or movement.

Moreover, we require that the scheme provides efficient **revocation** and **identification** mechanisms. This allows a service provider to learn the user's identity in case of malicious intents. If the user loses his credentials (typically a smart card with stored user secret keys and relevant attributes), the service provider can revoke the user from the system, by putting the user revocation handlers on the blacklist.

Most current scenarios involve many constrained devices (wearables, smart meters, sensors, RFID tags, smart cards etc.) with computation and memory limitations. Accordingly, we require the scheme to be **sufficiently fast even on constrained devices, in particular on smart cards**. Smart cards are considered to be tamper-resistant devices and, therefore, they provide secure storage for sensitive data, including user private keys. For this reason, we design novel schemes based on **elliptic curve cryptography** to reduce computational and memory resources on smart cards. It is important to notice that some operations (such as bilinear pairing) cannot be used on smart cards due to their unavailability on these devices.

2 STATE OF THE ART

2.1 Group Signatures

A group signature is a cryptographic primitive widely used for providing user privacy and anonymity. The basic idea is to hide a user inside the bigger group of other users. Hence, a verifier is not able to learn any personal information (including the identity) of a signer. The only information that the verifier receives is whether the signer is a member of the group or he is not. In other words, an (anonymous) group signature allows users to sign a message on behalf of the group, in such a way that a signature does not disclose which user was signing the message. In the classical digital signature scheme, each signer holds his own keypair consisting of two specific keys: one private and one public key. The group signature scheme is similar to the classical digital signature scheme. In case of group signatures, there is one public key which is related with a set of private keys. A group signature scheme usually involves the following entities:

Currently, there are many group signature proposals that mostly fulfill security and privacy requirements described above. The first group signature schemes where introduced by Chaum and Heyst [35] in 1991. These signatures are important especially from the theoretical point of view, since they are very inefficient. The inefficiency is given particularly due to big sizes of signatures and public keys together with their linear dependence in the number of group members. Over time, newer schemes were proposed. These proposals focus not only on privacy requirements but also on efficiency and practical usage, i.e. dynamism, speed, size of signature and public key, their independence in the number of group members (system or black listed users), and revocation techniques. For more details see paper [10].

Group signatures became part of many current ICT applications and services where the protection of user privacy is required. Nonetheless, group signature schemes are usually even more computationally expensive and produce bigger signatures in comparison with standard digital signature schemes such as RSA, DSA or ECDSA. However, the signatures complexity is the key for their practical usage. The complexity becomes more crucial in current systems such as IoT, Vehicular Ad hoc Networks (VANET), Smart Grids, Smart Cities, and Industry 4.0. In each of these systems, group signatures can be beneficial for users who are concerned about their privacy. Moreover, these systems are usually formed by many constrained devices with power and memory restrictions which must be addressed in the newest proposals.

In fact, the area of group signatures is addressed by different international standards and research papers. For example, the German Federal Office for Information Security (BSI) [52] provides a comparison of 12 selected group signature schemes which comply the basic security and privacy requirements. The paper [48] compares the performance of two group signature schemes on mobile devices, namely pairing-based BBS [22] and non-pairing-based ACJT [15] group signature schemes. The results show that the signing and verification phases of both schemes take few seconds (up to 3 s) on smartphones with Android platform and 1 GHz CPU. In case of full pre-computation use, the signing phase is even faster (up to 50 ms), since it computes one hash function and few modular multiplications and additions. However, in the case of no pre-calculations, the verification of one signature takes 14.14 s for BBS and 1.4 s for ACJT. Another closely related work is the paper written by Potzmader et al. [62]. The authors investigate the performance of three anonymous digital signature schemes on mobile devices, that are all included in the ISO/IEC 20008-2:2013 standard [12]. This standard defines seven anonymous digital signature schemes in total and provides a general description of group public key mechanisms.

Based on the papers mentioned above and the current user's privacy requirements in many ICT applications, we provide comprehensive evaluation of group signature schemes and their practical usability in current ICT applications such as access control, data collection and data notification, see paper [8] for more details. The summary of the comparison is depicted in the Table 1.

Table 1: Evaluation of group signatures schemes.

Scheme	**Sign Cost**	**Verify Cost**	**Sign Size**	**PK Size**	**Pairing**	**Assumption**	**Revoke**
BBS [24]	$9E_{\mathbb{G}_1}+3E_{\mathbb{G}_T}$	$1\mathbf{e}+8E_{\mathbb{G}_1}+2E_{\mathbb{G}_2}+3E_{\mathbb{G}_T}$	$3\mathbb{G}_1+6\mathbb{Z}_p$ (1545 b)	$4\mathbb{G}_1+2\mathbb{G}_2$ (1050 b)	✓	q-SDH, DLIN, ECDL	*sk*
DP [38]	$8E_{\mathbb{G}_1}+3E_{\mathbb{G}_T}$	$1\mathbf{e}+7E_{\mathbb{G}_1}+2E_{\mathbb{G}_2}+3E_{\mathbb{G}_T}$	$4\mathbb{G}_1+5\mathbb{Z}_p$ (1559 b)	$4\mathbb{G}_1+2\mathbb{G}_2$ (1050 b)	✓	q-SDH, XDH, DLIN, ECDL	*sk*
HLCCN [47]	$7E_{\mathbb{G}_1}+5E_{\mathbb{G}_T}$	$1\mathbf{e}+5E_{\mathbb{G}_1}+2E_{\mathbb{G}_2}+4E_{\mathbb{G}_T}$	$3\mathbb{G}_1+5\mathbb{Z}_p$ (5600 b)	$6\mathbb{G}_1+2\mathbb{G}_2$ (1400 b)	✓	q-SDH, XDH, DLIN, ECDL	*sk*
ACJT [15]	$12E_{\mathbb{G}_n^*}$	$10E_{\mathbb{G}_n^*}$	$7\mathbb{G}_n^*+1\mathbb{Z}_q$ (7328 b)	$6\mathbb{G}_n^*$ (6144 b)	✗	SRSA, DDH, DL	*cred*, *rl*
CG [28]	$10E_{\mathbb{G}_n^*}$	$10E_{\mathbb{G}_n^*}$	$8\mathbb{G}_n^*+1\mathbb{Z}_q$ (8352 b)	$7\mathbb{G}_n^*+1\mathbb{Z}_q$ (7328 b)	✗	SRSA, DDH, DL	*cred*, *rl*
IMSTY [49]	$7E_{\mathbb{G}_n^*}+8E_{\mathbb{G}_1}$	$7E_{\mathbb{G}_n^*}+8E_{\mathbb{G}_1}$	$5\mathbb{G}_n^*+5\mathbb{Z}_p+1\mathbb{Z}_q$ (6155 b)	$7\mathbb{G}_n^*+4\mathbb{Z}_p$ (7848 b)	✗	SRSA, DH, ECDL	*cred*
HM GS [45]	$9E_{\mathbb{G}_n^*}$	$10E_{\mathbb{G}_n^*}$	$7\mathbb{G}_n^*+1\mathbb{Z}_q$ (7328 b)	$5\mathbb{G}_n^*$ (5120 b)	✗	DL, IF	*rl*

Note: $E_{\mathbb{G}_1}$ – EC scalar multiplication in $\mathbb{G}_1$, similarly $E_{\mathbb{G}_2}$ and $E_{\mathbb{G}_T}$, **e** – bilinear pairing, *sk* – group member private key, *cred* – credential, *rl* – revocation list.

Our performance results are depicted in Figure 1. Note that non-pairing schemes show better performance results than paring-based schemes. Only in case of IMSTY scheme, we can see a significant increase of the verification and signing time. This is due to elliptic curve operations execution on Android device.

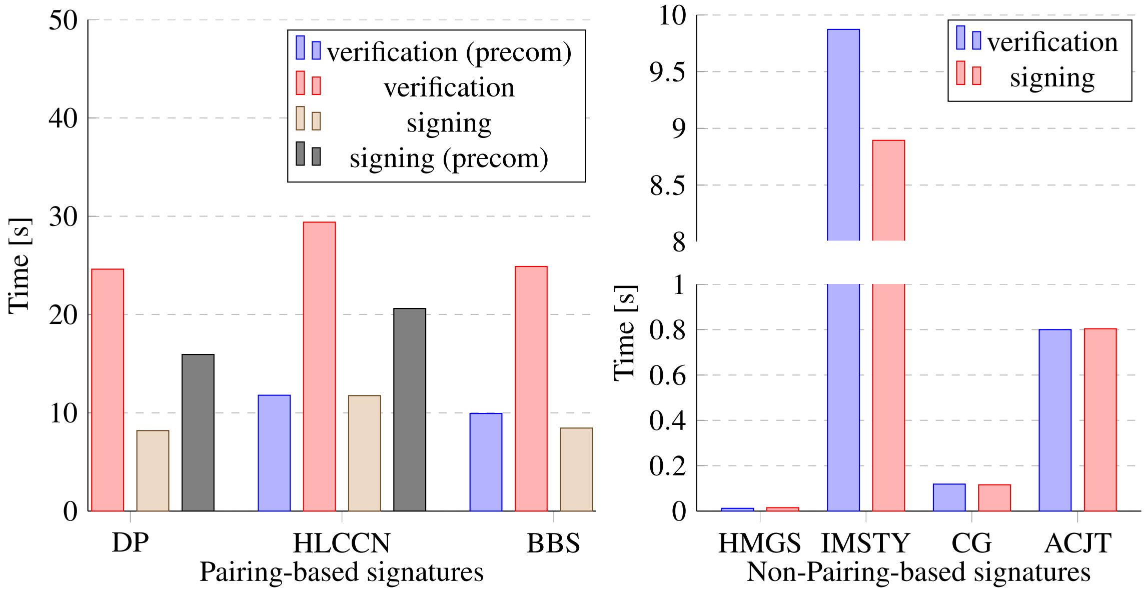


Figure 1: Performance of non-pairing and pairing based group signatures on Nexus 5 LG.

2.2 Attribute-Based Credentials

Attribute-Based Credential is a cryptographic construction, that is a basic pillar of so-called attribute-based authentication schemes. In contrast to the classical authentication schemes based on identity, ABC schemes are more privacy-friendly, since they do not disclose user identity or other private information, that is not mandatory to gain an access to the required service. In many scenarios, it is not necessary to know a user identity to get an access. More important is to know, whether the user holds some personal attributes (his specific properties) which are directly related to the service scenario. Attributes are grouped together in a cryptographic (digital) *credential* as depicted in Figure 2. The credential is a cryptographic container for attributes signed by a trusted party. In general, we say, that credentials are issued and attributes are shown. Moreover, credentials usually include user's key, which provides non-transferability. In this context we can construct different credential types including set of common attributes:

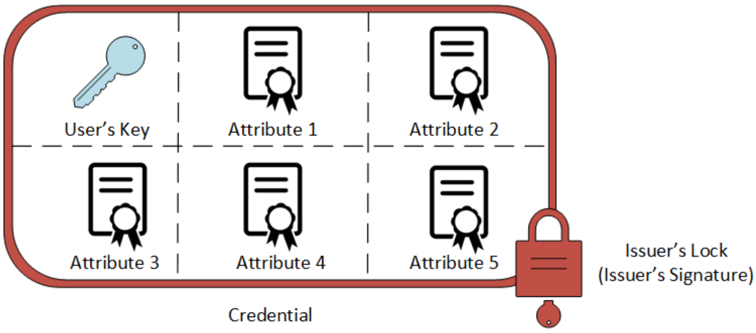


Figure 2: Cryptographic credential construction.

Anonymous credentials hide the attributes, so, seeing a credential, no one can obtain any information about the attributes in it. Furthermore, the credentials allow user to authenticate himself without identification, and provide session unlinkability. Nowadays, there is only several attribute-based credential schemes, e.g. U-Prove [58], Idemix [19] and Hajny-Malina [43].

U-Prove

U-Prove is an anonymous attribute-based credential scheme [58] that belongs to Microsoft company. However, the scheme was first introduced and developed by Credentica company. The underlying cryptographic protocols were designed by Dr. Stefan Brands as a part of his Ph.D. thesis. Scheme security is based on discrete logarithm assumption. U-Prove uses same group as DSA signature scheme. In another words, U-Prove group is a prime order subgroup $\mathbb{Z}_q$ in the multiplicative group of a finite prime field $\mathbb{Z}_p^*$. The scheme uses a variant of the blind Schnorr signature [60] that is the key underlying cryptographic primitive of the scheme. Schnorr signature is used in an attribute issue protocol and guaranties untraceability of credentials by the Issuer. The attribute verify protocol uses the proof of knowledge protocols (cryptographic commitments and Σ-protocols), in particular a variant of the Schnorr protocol [66] is used. A U-prove user can selectively disclose a subset of his attributes, therefore a user is able to control how much information he releases. On the other hand, the scheme does not provide session unlinkability, since all credentials consist of a unique identificator Prover Information (PI) field. PI servers among others as a revocation handler, which allows to revoke dishonest users from the system. It is important to notice that the user real identity remains hidden and there is no way to disclose it. Currently there are only few implementations of the U-Prove protocol on smart cards. The most efficient implementation was provided on MultOS [53]. The attribute proving time depends on the number of stored attributes on the smart card and the number of disclosed attributes within the verification protocol, see Figure 3. However, in case of 5 attributes stored, the proving time is always under 1 s in each scenario.

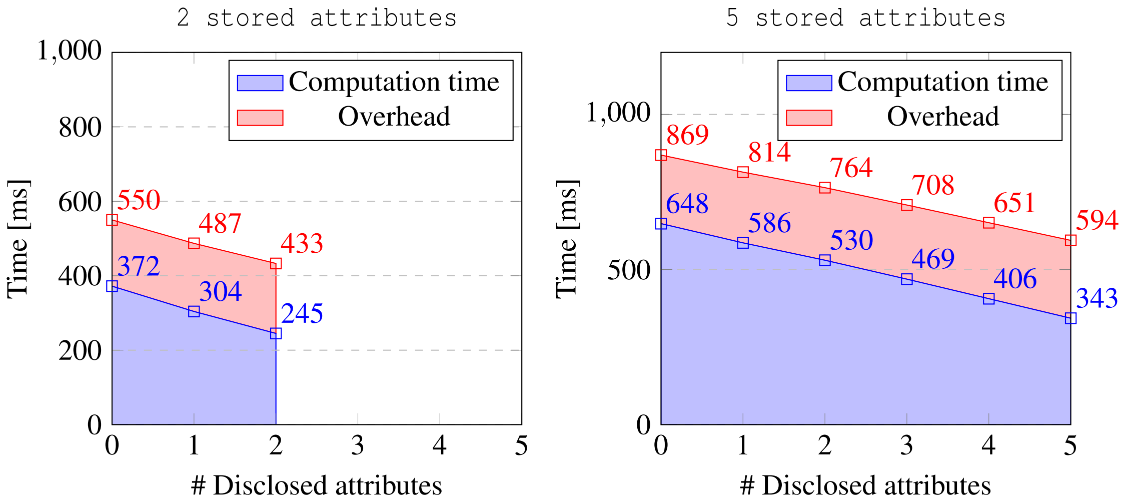


Figure 3: U-Prove attributes proving time for different scenarios.

Idemix

Idemix (Identity Mixer) is an anonymous attribute-based credential scheme [19] developed by IBM Research Zurich. The scheme is based on Camenisch-Lysyanskaya signature [31] that allows the Issuer to sign User's attributes to construct a cryptographic credential within the issue protocol. The User randomizes and sends the credential to the Verifier and then anonymously proves possession of attributes to the Verifier by using zero-knowledge proof of knowledge protocols. The scheme security is held under *strong RSA assumption* in a cyclic group modulo composite $n = pq$, as well as in case of RSA cryptosystem. In contrast to U-Prove, the Idemix provides session unlinkability, that makes it impossible to track Users' movement and behaviour. Since every credential is randomized, there is no efficient revocation mechanism. Hence, the credentials may include time epoch information for limiting its validity or the scheme must be extended by external revocation scheme, e.g. [27]. Currently the most efficient implementation was provided on MultOS card [70], where the proof generation takes up to 1.5 s if 5 attributes were stored, see Figure 4.

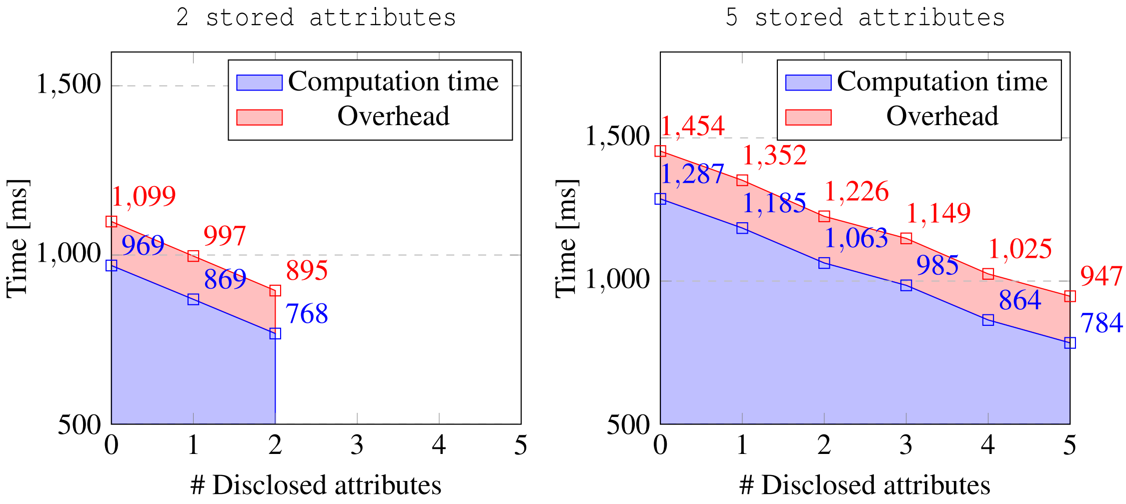


Figure 4: Idemix attributes proving time for different scenarios.

HM12

Hajny-Malina (HM12) [43] is an attribute-based credential scheme with practical revocation developed by the Cryptology Research Group at Brno University of Technology in the Czech Republic. The scheme was first designed by Jan Hajny as a part of his Ph.D. thesis [42]. The scheme security is held under *discrete logarithm assumption* in Okamoto-Uchiyama group $\mathbb{OU}$, i.e. in a multiplicative cyclic group modulo composite number $\mathbb{Z}_n^*$, where $n = r^2 s$ and r, s are primes. The scheme uses the Okamoto-Uchiyama cryptosystem [56] as a key cryptographic primitive. This primitive allows the Manager to decrypt a proof generated by the User and thus disclose User's identity and revoke him from the system. For this reason, OU property acts mainly in `Issue_Att` and `Revoke` protocols, while the `Prove_Att` protocol runs fully over Σ-protocols (namely a Proof of Knowledge Discrete Logarithm Representation (PKDLR)). In contrast to previous schemes, the HM12 scheme provides practical revocation mechanisms, i.e. scheme itself allows to revoke issued credentials on the User's, Issuer's or Verifier initiatives. The scheme also supports revocation of credential unlinkability and User's anonymity. At the same time, there is required to involve more parties to the revocation process. For example, if Issuer, Manager and Verifier cooperate, they can revoke the User's anonymity while the cooperation only of Manager and Verifier allows to revoke session unlinkability and invalid credentials. The scheme is potentially weak against a cryptographic collusion attack, where more Users can in cooperation create a valid but unregistered User [13]. The weakness was solved in the protocol extension [5]. However, if we consider a tamper-resistance device (such as smart card), where the cryptographic keys are stored, we can avoid these collusion attacks.

Currently the most effective implementation was provided on MultOS ML3 smart card [46] in a 1024-bit protocol variant. The verification time takes ca. 2.9 ms for one attribute disclosed. To provide comprehensive measurement of the scheme, we developed a smart card application that allows us to store and disclose up to 5 attributes. Our implementation (1024-bit version) was run on MultOS ML4 card. The time grows linearly with the number of disclosed attributes, see Figure 5. The number of stored attributes has no impact on the final time, since the attributes are not grouped in to the credential. Using the newer ML4 card instead of the older ML3 card, we reduced the attribute proving time by ca. 56%. However, the time complexity can be even more reduced by involving more computationally powerful devices. For example, the paper [4] uses 1392-bit protocol variant implementation on various smart phones to achieve the verification time under 100 ms.

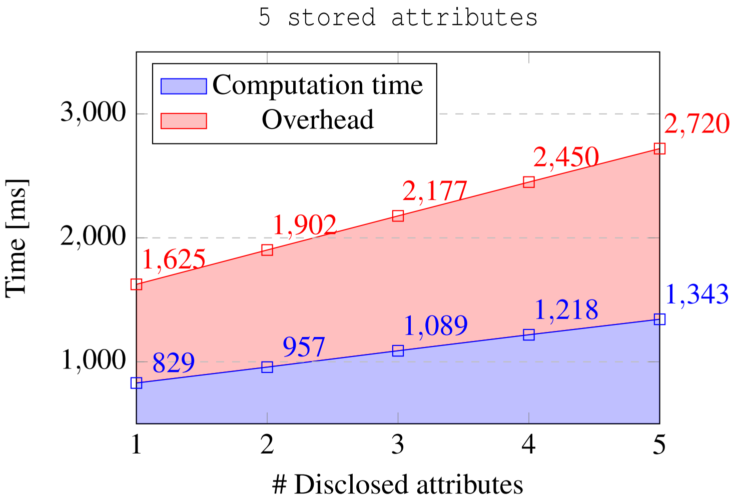


Figure 5: HM12 attributes proving time for different scenarios.

Important to note: the time complexity grows linearly with the number of blacklisted attributes, since each attribute check involves one modular exponentiation.

2.3 Smart Cards

A smart card is a small plastic card, typically of credit-card size, that has an embedded integrated circuit, see Figure 6. The circuit can store and/or process data (microprocessor and/or memory chip is used) and communicate with a terminal via communication interface (i.e. antenna or contact pad). Smart cards evolved from simple memory cards to very efficient "*microcomputers*" with many applications. The security and portability of smart cards provide a fast way to ensure secure transactions, e.g. banking or e-business, and can be used in any system that requires secure authentication. In fact, smart cards are considered tamper-resistant storage devices protecting private keys and other sensitive information. Moreover, they contribute to the achievement of a safe environment for security-critical computation executions, as in the case of authentication, digital signature, and key exchange schemes. Since, our privacy-enhancing protocols, developed and described in this thesis, are primarily intended for card-based authentication and signature schemes, and they use elliptic curve constructions, we provide a short overview of the current state of the art of smart card technologies. Our main interest is focused on hardware cryptographic support of elliptic curve operations.The main contribution in this section has been published in articles [3] and [9].

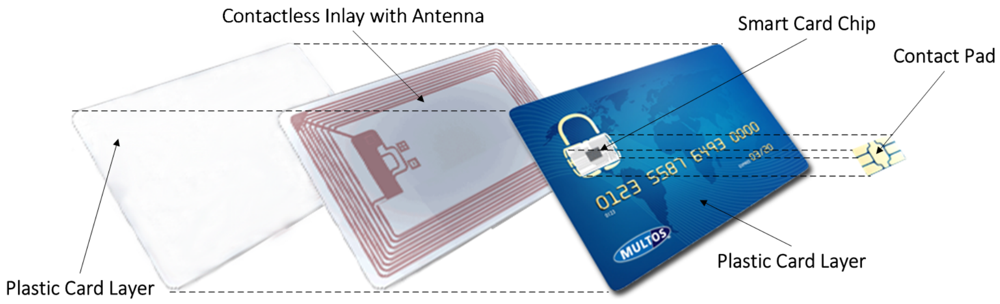


Figure 6: Smart card construction.

Smart cards are a closed platform, i.e., it is not usually possible to upgrade cryptographic libraries on the card. Cryptographic support differs according to the smart card platform: Java Card, MultOS, Basic Card, .NET Card, version of the operating system and the smart card implementation itself. In fact, there is often an inconsistency between the platform specification and the real implementation of smart cards' Application Programming Interface (API) due to the implementer company, e.g. NXP, Gemalto, Giesecke & Devrient, Feitan, Oberthur, Ubivelox, Hitachi, Samsung, MultOS International, ZeitControl GmbH.

Table 2 shows the support of cryptographic functions on different smart card platforms. These types of security functions are: symmetric cryptography (`Symmetric Crypto`), asymmetric cryptography (`AsymmetricCrypto`), hash functions (`Message Digest`), random number generator functions (`RandomData`), modular arithmetic operations (`ModularArithmetic`) and elliptic curve operations (`EllipticCurve`). The table presents the basic overview of supported functions, since the platforms usually offer various operating system versions and smart card implementations.

Advanced cryptographic protocols usually require modular arithmetic operations such as *multiplication* and *exponentiation* with big integers, as well as operations over elliptic curves, including *point addition* and *scalar multiplication*. These operations are provided by MultOS and Basic Card platforms. Java Card offers many standard cryptographic schemes, but the underlying mathematical operations, such as modular arithmetic and elliptic curve operations, are still missing.

Since we are mostly interested in elliptic curve cryptography and related underlying mathematics operations, we provide detail description of supported algorithms on different smart card operating systems in the Table 3.

Table 2: Cryptographic and mathematical support of smart card platforms.

	Java Card	**Basic Card**	**MultOS**	**.NET Card**
Symetric Crypto	DES, TDES, AES (keys up to 256 b), SEED, CBC/ECB modes, CMAC, HMAC	DES, TDES, AES (keys up to 256 b), CBC/CFB/OFB/EAX modes, OMAC	DES, TDES, AES (keys up to 256 b), SEED, CBC/ECB modes	DES, TDES, AES (keys up to 256 b), ECB/CBC modes
Asymetric Crypto	RSA (up to 4096 b), DSA (up to 1024 b), ECDH, ECDSA (up to 512 b)	RSA (up to 4096 b), ECDSA, ECDH, ECNR signature (up to 521 b)	RSA (up to 2048 b), ECDH, ECDSA, ECIES (up to 512 b)	RSA (up to 2048 b)
Message Digest	MD5, RIPEND160, SHA-1, SHA-2, SHA-3 (JC 3.0.5)	SHA-1, SHA-2 (up to 512 b)	SHA-1, SHA-2 (up to 256 b)	MD5, SHA-1, SHA-2 (up to 256 b)
Random Data	Pseudo RND, TRNG (JC 3.0.5)	4B RND function, TRNG	TRNG	Pseudo RNG, TRNG
Modular Arithmetic	**not supported**, exponentiation (RSA encryption), multiplication (only software solution with RSA tunnel: $ab = [(a+b)^2 - a^2 - b^2]/2$)	supported (up to 16 kB)	supported (up to 2048 b)	**not supported**, same as **Java Card**
Elliptic Curve	**not supported**, JC 3.0.5: scalar multiplication (ECDH_PLAIN_XY), point addition (PACE_GM)	supported (up to $\mathbb{F}_{521}$ or $\mathbb{F}_{2^{193}}$)	supported (up to $\mathbb{F}_{512}$)	**not supported**

The basic arithmetic operations on elliptic curves are point addition (ecAdd), scalar multiplication (ecMul) and point inverse (ecInv). We provide the speed of all these operations for Java Card, MultOS and Basic Card. Each operation was averaged over 100 executions on all the aforementioned smart cards. Then, the result was sent to the PC for evaluation. An emphasis is on elliptic curve cryptography benchmarks carried out on the different types of smart cards, since our privacy-enhancing schemes, proposed in Sections 3–6 are based on it. The technical specification of tested smart cards is shown in Table 4.

Figure 7 depicts the ecMul cost for Brainpool curves on Java Card, MultOS and Basic Card. MultOS card are 75% faster than Basic cards (ZC7.6) and 35% faster than the fastest Java cards (J3A081). JC Sm@rtCafe implementations show worse results than JCOP implementations.

Figure 7 shows ecAdd and ecInv costs on MultOS and Basic Card. For MultOS, ecAdd and point doubling require the same time. The ecAdd operation is 20% faster on MultOS cards than on Basic cards.

Table 3: Elliptic curve cryptography support on smart card platforms.

	Version	ecAdd	ecMul	ecInv	ECIES	ECDH	ECDSA	$\lvert\mathbb{F}_p\rvert/\lvert\mathbb{F}_{2^m}\rvert$	Space
Java Card	JC 2.2.2	✗	✗	✗	✗	✓	✓	192/193	$\mathscr{A}$
	JC 3.0.1	✗	✗	✗	✗	✓	✓	384/193	$\mathscr{A}$
	JC 3.0.4	✗	✗	✗	✗	✓	✓	521/193	$\mathscr{A}$
	JC 3.0.5	✓!	✓!	✗	✗	✓	✓	521/193	$\mathscr{A}$
	JCOP2.4.1	✓!	✓!	✗	✗	✓	✓	320/–	$\mathscr{A}$
MultOS	4.2	✓	✓	✓	✓	✓	✓	384/–	$\mathscr{A},\mathscr{P}$
	4.3.1 - 4.5.1	✓	✓	✓	✓	✓	✓	512/–	$\mathscr{A},\mathscr{P}$
BasicCard	ZC5, ZC6	✗	✗	✗	✗	✓	✓	–/211	$\mathscr{A},\mathscr{T}$
	ZC7, ZC8	✓	✓	✗	✗	✓	✓	544/211	$\mathscr{A},\mathscr{T}$
.NET	Gemalto .NET 2.0	✗	✗	✗	✗	✗	✗	–/–	–

Note: ✓– algorithm is fully supported, ✓!– algorithm is supported, but there is not direct access, ✗– algorithm is not supported, $\mathbb{F}_p$ – prime finite field, $\mathbb{F}_{2^m}$ – binary finite field, $\mathscr{A}$ – affine space, $\mathscr{P}$ – projective space, $\mathscr{T}$ – twisted curve.

Table 4: Technical specification of tested smart cards.

	J3A081	J3D081	Sm@rtCafe6	Sm@rtCafe5	ZC7.6	ML4	ML3	Gemalto
MCU	P5CD 081	P5CD 081	P5CD 081	P5CDs 080	–	SC23 Z018	SLE78 CLXPM	–
OS	JavaCard 2.2.2	JavaCard 3.0.1	JavaCard 3.0.1	JavaCard 2.2.2	Basic ZC7	MultOS 4.3.1	MultOS 4.3.1	.NET 2.2
ROM	264KB	264KB	264KB	200KB	–	252KB	280KB	80KB
EEPROM	80KB	80KB	80KB	80KB	72KB	18KB	96KB	400KB
RAM	6KB	6KB	6KB	6KB	4.3KB	1.75KB	2KB	16KB

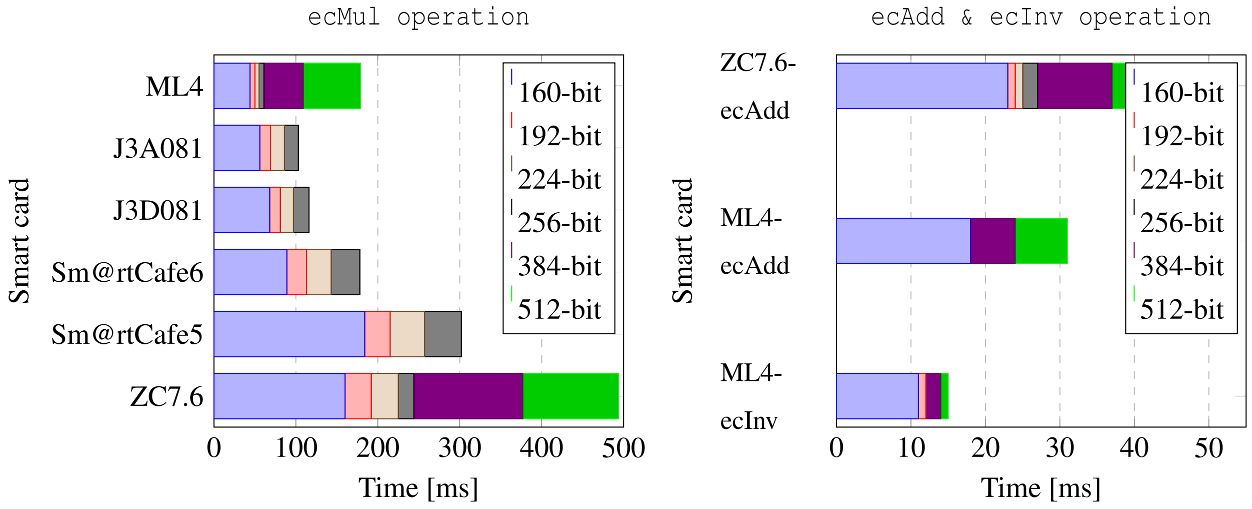


Figure 7: Efficiency of EC operations on different smart card platforms.

3 MULTI-DEVICE AUTHENTICATION WITH STRONG PRIVACY PROTECTION

The content of this section have been published in impact factor journal paper [6]. The cryptographic scheme presented in this section takes a novel approach for the access control based on rather the presence of multiple devices in user's proximity than the direct verification of user identifiers. The novel approach has two key benefits: (1) it significantly improves the privacy protection of users and (2) allows the authentication based on the presence of many low-performance devices.

Our scheme is the first practical proposal with implementation results that combines strong security, all standard privacy-enhancing features and efficiency: (1) **Provable security**: all algorithms are provably secure, based on primitives with rigorous formal proofs, (2) **Multi-device authentication**: the scheme allows user authentication based on the presence of many personal devices, (3) **Anonymity**: the scheme allows authentication based on anonymous proofs of knowledge of private user and/or device identifiers, (4) **Unlinkability**: the scheme prevents creating user behavior profiles based on the authentication sessions linking, (5) **Untraceability**: the scheme prevents any entity from tracing users (or their devices), (6) **Efficiency**: the authentication protocol is fast on constrained user devices (i.e., smart cards) and embedded verification terminals, (7) **Revocation and identification**: the proposed scheme is compatible with major revocation and identification schemes.

3.1 General Architecture

The communication pattern is depicted in Figure 8 and employs the registrar (i.e., a central server that manages users and their equipment), users (i.e., user devices such as smart cards or smart phones), terminals (i.e., embedded devices with RFID readers typically attached next to doors) and tags (i.e., devices that need to be present during authentication and access control, typically safety equipment with programmable RFID sticks, such as the helmet, respirator or harness).

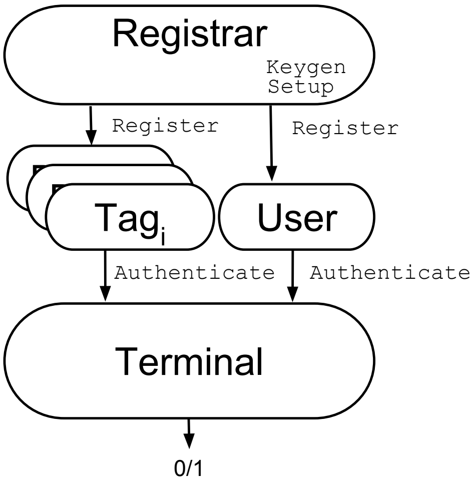


Figure 8: Architecture of multi-device authentication with privacy protection.

3.2 Cryptography Specification

We use the wBB signature scheme to certify the identifiers of tags and users in the `Register` algorithm and interactive proofs of knowledge to prove the knowledge of respective signatures and identifiers in the `Authenticate` protocol.

Setup

$(par) \leftarrow$`Setup`$(1^{\kappa}, n)$: the algorithm inputs the security parameter κ and the maximum number of tag classes n. It generates the bilinear group with parameters $par = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathbf{e}, g_1, \ldots, g_n, g_u \in \mathbb{G}_1, g_2 \in \mathbb{G}_2)$ satisfying $|q| = \kappa$.

Keygen

$(sk_r, pk_r) \leftarrow$`Keygen`(par): the algorithm inputs the public parameters par, selects random registrar's private keys $sk_r = (sk_0, sk_1, \ldots, sk_n, sk_u) \xleftarrow{\$} \mathbb{Z}_q^*$ and computes the public keys $pk_r = (pk_0 \leftarrow g_2^{sk_0}, pk_1 \leftarrow g_2^{sk_1}, \ldots, pk_n \leftarrow g_2^{sk_n}, pk_u \leftarrow g_2^{sk_u})$. It outputs the private keys as registrar's private output and the public key as the public output.

Register

$(\langle ID_i, \sigma_i \rangle_{i=1}^{n}, ID_u, \sigma_u) \leftarrow$`Register`$(par, sk_r, pk_r)$: the algorithm inputs the registrar's keys and public parameters, randomly selects tag and user identifiers $(ID_1, \ldots, ID_n, ID_u) \xleftarrow{\$} \mathbb{Z}_q$ and computes the wBB signatures $(\sigma_1, \ldots, \sigma_n)$ on tag identifiers $(ID_1, \ldots, ID_n)$ and the aggregated user signature σ_u and auxiliary values $\langle \sigma_{u_i}, \sigma_{u_i}^{-ID_i} \rangle_{i=1}^{n}$, $\sigma_{u_u}, \sigma_{u_u}^{-ID_u}$ that allow the construction of efficient proofs of knowledge in the `Authenticate` protocol. The algorithm outputs the tag identifiers and corresponding signatures as a private output to tags. The user identifier, the aggregated signature and auxiliary values are outputted to the user as a private output. Both tags and the user receive the initial *seed* required for the synchronization of the zero-knowledge proofs as a private input. The algorithm is depicted in Figure 9.

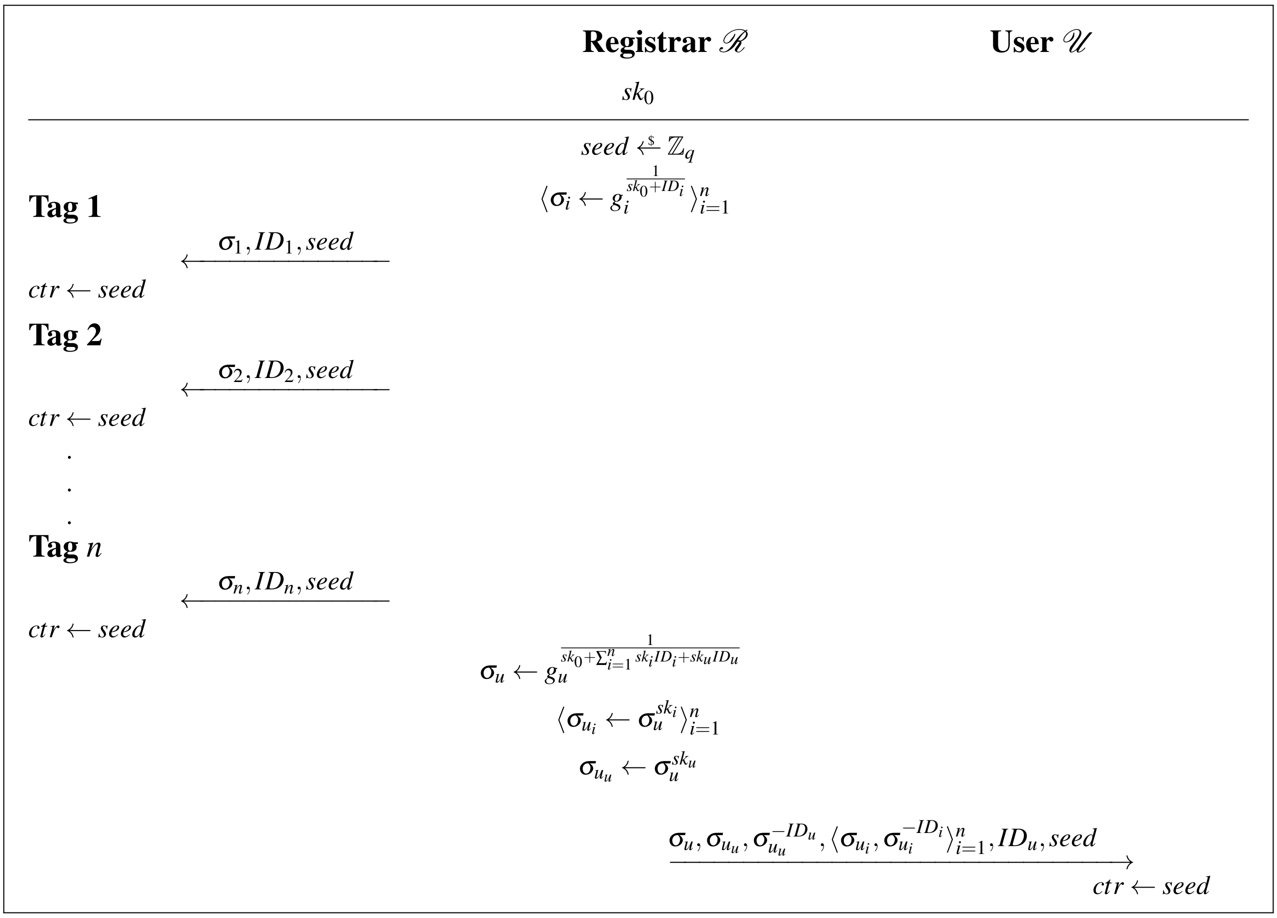

Registrar $\mathscr{R}$
User $\mathscr{U}$
sk_0
$seed \xleftarrow{\$} \mathbb{Z}_q$
$\langle \sigma_i \leftarrow g_i^{\frac{1}{sk_0 + ID_i}} \rangle_{i=1}^{n}$
Tag 1
$\xleftarrow{\sigma_1, ID_1, seed}$
$ctr \leftarrow seed$
Tag 2
$\xleftarrow{\sigma_2, ID_2, seed}$
$ctr \leftarrow seed$
.
.
.
Tag n
$\xleftarrow{\sigma_n, ID_n, seed}$
$ctr \leftarrow seed$
$\sigma_u \leftarrow g_u^{\frac{1}{sk_0 + \sum_{i=1}^{n} sk_i ID_i + sk_u ID_u}}$
$\langle \sigma_{u_i} \leftarrow \sigma_u^{sk_i} \rangle_{i=1}^{n}$
$\sigma_{u_u} \leftarrow \sigma_u^{sk_u}$
$\xrightarrow{\sigma_u, \sigma_{u_u}, \sigma_{u_u}^{-ID_u}, \langle \sigma_{u_i}, \sigma_{u_i}^{-ID_i} \rangle_{i=1}^{n}, ID_u, seed}$
$ctr \leftarrow seed$

Figure 9: `Register` protocol.

Authenticate

$(0/1) \leftarrow$`Authenticate`$(par, \langle ID_i, \sigma_i \rangle_{i=1}^{n}, ID_u, \sigma_u, pk_r)$: the algorithm is distributed among the user, terminal and tags that inputs the identifiers and respective signatures and outputs 1 iff 1) all signatures are valid and created by the registrar, and 2) all identifiers of the user are present and signed. Otherwise it outputs 0. The protocol is a distributed proof of knowledge of wBB signatures where the tags prove that they know their identifiers and corresponding signatures (without actually revealing them) and, at the same time, the user proves that he has an aggregated signature on all his tag identifiers, plus his own identifier. As the user does not know the tag identifiers, all tags must be present and participate on the proof construction. As a result, the user is able to anonymously, untraceably and unlinkably prove his valid registration by the registrar and the presence of all his tags, i.e., the safety equipment. We also provide the full description of `Authenticate` protocol for i^{th} tag in Figure 10.

$$
\begin{array}{lcccr}
\textbf{Tag } i & & \textbf{Terminal } \mathscr{T} & & \textbf{User } \mathscr{U} \\
\hline
ctr{+}{+} & & & & ctr{+}{+} \\
r_i, \rho_{r_i} \xleftarrow{\$} \mathbb{Z}_q & & & & r_u, \rho_{ID_u} \xleftarrow{\$} \mathbb{Z}_q \\
\rho_{ID_i} \leftarrow \mathscr{H}(ctr, g_i) & & & & \rho_{ID_i} \leftarrow \mathscr{H}(ctr, g_i) \\
\sigma'_i \leftarrow \sigma_i^{r_i} & & & & \sigma'_{u_i} \leftarrow \sigma_{u_i}^{r_u} \\
\bar{\sigma}_i \leftarrow \sigma_i'^{-ID_i} g_i^{r_i} & & & & \sigma'_{u_u} \leftarrow \sigma_{u_u}^{r_u} \\
 & & & & \sigma'_u \leftarrow \sigma_u^{r_u} \\
 & & & & \bar{\sigma}_u \leftarrow (\sigma_{u_i}{}^{-ID_i} \sigma_{u_u}{}^{-ID_u} g_u)^{r_u} \\
t_i \leftarrow \sigma_i'^{\rho_{ID_i}} g_i^{\rho_{r_i}} & & & & t_u \leftarrow \sigma'^{\rho_{ID_i}}_{u_i} \sigma'_{u_u}{}^{\rho_{ID_u}} g_u^{\rho_{r_u}} \\
 & \xrightarrow{\bar{\sigma}_i, \sigma'_i, t_i, \mathscr{H}(ctr)} & & \xleftarrow{\bar{\sigma}_u, \sigma'_{u_u}, \sigma'_i, \sigma'_{u_i}, t_u, \mathscr{H}(ctr)} & \\
 & & c \xleftarrow{\$} \mathbb{Z}_q & & \\
 & \xleftarrow{c, \mathscr{H}(ctr)} & & \xrightarrow{c, \mathscr{H}(ctr)} & \\
s_{r_i} \leftarrow \rho_{r_i} + cr_i & & & & s_{r_u} \leftarrow \rho_{r_u} + cr_u \\
s_{ID_i} \leftarrow \rho_{ID_i} - cID_i & & & & s_{ID_u} \leftarrow \rho_{ID_u} - cID_u \\
 & \xrightarrow{s_{ID_i}, s_{r_i}, \mathscr{H}(ctr)} & & \xleftarrow{s_{r_u}, s_{ID_u}, \mathscr{H}(ctr)} & \\
 & & t_i \stackrel{?}{=} g_i^{s_{r_i}} \sigma_i'^{s_{IDi}} \bar{\sigma}_i{}^{-c} & & \\
 & & t_u \stackrel{?}{=} g_u^{s_{ru}} \sigma'^{s_{IDi}}_{u_i} \sigma'^{s_{IDu}}_{u_u} \bar{\sigma}_u{}^{-c} & & \\
 & & \mathbf{e}(\bar{\sigma}_i, g_2) \stackrel{?}{=} \mathbf{e}(\sigma'_i, pk_0) & & \\
 & & \mathbf{e}(\bar{\sigma}_u, g_2) \stackrel{?}{=} \mathbf{e}(\sigma'_u, pk_0) & & \\
 & & \mathbf{e}(\sigma'_{u_i}, g_2) \stackrel{?}{=} \mathbf{e}(\sigma'_u, pk_i) & & \\
 & & \mathbf{e}(\sigma'_{u_u}, g_2) \stackrel{?}{=} \mathbf{e}(\sigma'_u, pk_u) & & \\
 & \xleftarrow{0/1} & & \xrightarrow{0/1} &
\end{array}
$$

Figure 10: `Authenticate` protocol in full notation for i^{th} tag.

Revocation and Identification

All users are theoretically identifiable and traceable by their user IDs. However, these IDs are "hidden" in the signatures as the exponents. Due to the discrete logarithm problem assumption, one cannot easily get the identifiers and do the revocation and identification. However, our scheme is compatible with the major revocation schemes that are already available for cryptographic anonymous credential schemes [20, 25, 27]. In these revocation schemes, the hidden exponent (the user ID) is used as a revocation handle and can be disclosed only by designated authorities. Additionally, valid users remain anonymous while malicious users are identifiable and traceable by a designated authority, such as a court. Such schemes are provably secure, efficient and compatible without any modification, thus we refer to their specification (e.g., the scheme designed directly for smart cards [27]) in case revocation is needed.

3.3 Implementation and Performance Analysis

The `Authenticate` protocol has been implemented as a standard 3-way interactive zero-knowledge proof of knowledge protocol. We use a parallel composition with one challenge and one response for all tags of a user to construct an AND proof for both tag and user signatures. We provide performance measurement of crucial operations on common devices, which are widely used in the access control applications, i.e. a smart card, smart phone, smart watch (as user devices), a custom-built RFID terminal with ARM or Intel CPU and programmable RFID tags (as RFID tags attached to safety equipment). The hardware and software specification of all the devices is presented in Table 5.

Table 5: Specification of tested devices.

	Type	**CPU/MCU**	**OS**	**RAM**
Tag	Smart Card	SC23Z018	MultOS 4.3.1	1.75 KB
User	Smart Card	SC23Z018	MultOS 4.3.1	1.75 KB
User	Phone	Kirin 655	Android 7.0	3 GB
User	Watch	ARM Cortex-A7	Android 7.0	768 MB
Terminal	Pi 3	ARM Cortex-A53	Raspbian 9.3	1 GB
Terminal	PC	Intel i7-7700	Debian 8.6	16 GB

Note: Tag – programmable RFID stick, User – user device, Terminal – terminal, Smart Card – ML4, Phone – HUAWEI P9 Lite 2017, Pi 3 – Raspberry Pi 3 Model B, Watch – HUAWEI Watch 2

The testing scenario is depicted in Figure 11. The user needs to hold a wearable device, such as a smart phone (HUAWEI P9 Lite 2017), a smart card (MultOS Card) or smart watch (HUAWEI Watch 2) and some safety equipment, such as helmets, harnesses, boots, protective suits, each of them with a programmable RFID tag attached. The tag is equipped with a programmable chip SC23Z018 with MultOS 4.3.1 operation system. The proofs are collected and verified by a terminal. We use Raspberry Pi 3 to represent the terminal. In another scenario, PC (Intel i7-7700 CPU, 16 GB RAM) acts as a central authentication server representing the case of a centralized access control system. The system uses RFID communication between tags and a terminal, and NFC or Bluetooth Low Energy (BLE) communication between a terminal and a user device.

The proposed authentication scheme can by used in many types of access control scenarios and for different types of devices. Therefore, we provide the results of each protocol using one RFID tag. Furthermore, we present the crucial EC operations' benchmarks on a wide range of devices in Table 6. The time is measured in milliseconds and the values are an average of 10 measurements,

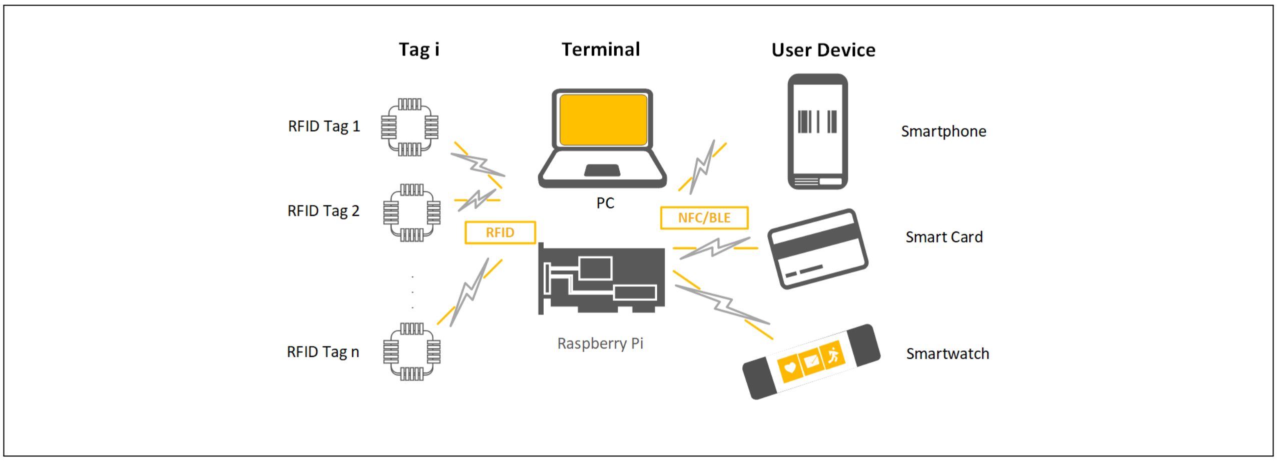


Figure 11: Tested scenario.

as in the previous case. All measurements were performed by using the elliptic curve d159 from the PBC library. For the implementation of EC operations, the PBC library [51] was used on the terminal and jPBC [36] library on Android devices. Native assembler code was used to perform operations on the MultOS smart card. We did not consider Android devices as a terminal device, since the pairing operation requires too much time and therefore it is not usable in practice.

Table 6: Benchmark results of all tested devices.

	Smart Card	**Phone**	**Watch**	**Raspberry Pi 3**	**PC**
	[ms]	**[ms]**	**[ms]**	**[ms]**	**[ms]**
Exponentiation	40	38	207	3.3	0.4
Pairing	-	1050	6571	31	2.4
Tag Proof Generation	277	154	900	18	4
User Proof Generation	441	273	1502	24	5
Verification	-	-	-	271	21

Figure 12 depicts the time required for a proof construction on different devices (MultOS smart card, Android smart phone and smart watch for various number of tags). These devices act as a user device.

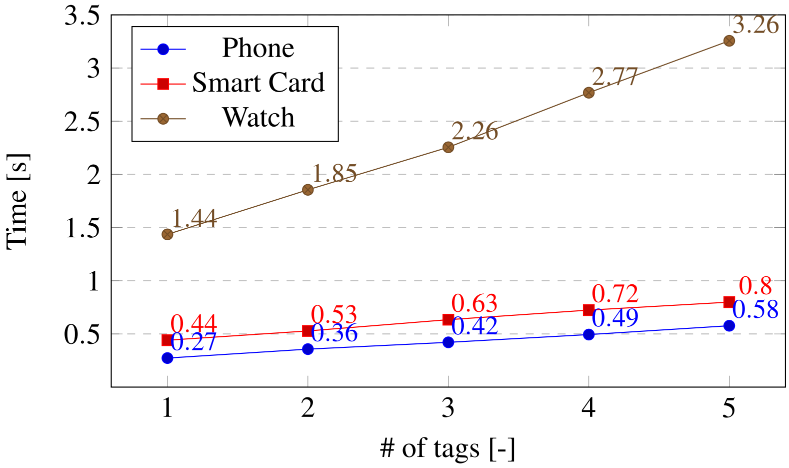


Figure 12: Time dependence of the proof generation on the number of user device.

4 ANONYMOUS DATA COLLECTION SCHEME FROM SHORT GROUP SIGNATURES

The content of this section have been published in conference paper [7]. In this section, we propose a novel cryptographic scheme that we call an anonymous data collection scheme that is instantiated using the wBB signature [23] and the efficient proofs of their knowledge [27]. On a general level, we take the approach of [59], i.e., we let the manager sign all users' private keys. The users then prove the knowledge of such a signature and verifier checks the proof using the manager's public key. Our scheme is unique in the following properties: (1) provides all **privacy-enhancing** features: anonymity, unlinkability, untraceability, (2) the signatures are **small** and **constant**: the size is below 169 B using a strong 224 b curve, (3) the signature generation is **fast**: requires no bilinear pairing and only 5 exponentiations, (4) the signature verification including **revocation** check is efficient: requires only 2 pairings and $\mathcal{O}(|RL|^1)$ exponentiations, and (5) the scheme is built using primitives with formal security proofs.

4.1 General Architecture

Three types of entities interact in our data collection scheme: a manager, a user and a collector. The manager generates cryptographic parameters and keys. It also enrols new users (devices) and revokes invalid ones. The user is represented by its device, such as a smart meter, sensor or some wearable device. It is the source of data that are signed and transferred to the central device (collector). The collector represents the central node that collects all data from users and verifies the group signatures. The privacy-enhanced data collection scheme is presented in Figure 13.

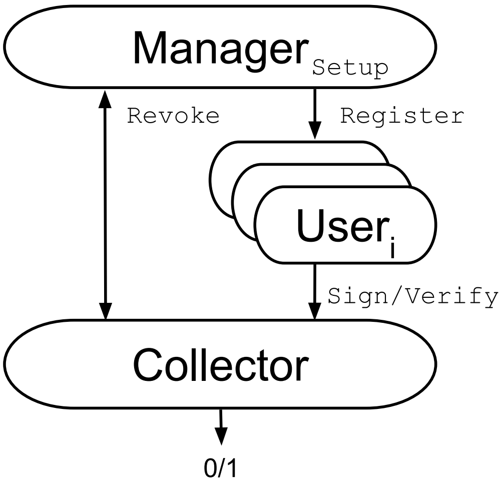


Figure 13: Architecture of the scheme proposed.

4.2 Cryptography Specification

We instantiate the algorithms of the data collection scheme presented in the previous section using the wBB signature [23] and its efficient proof of knowledge [27]. On a high level, we let the user to obtain a wBB signature on his private identifier from the manager. Then, the user proves the knowledge of such a signature anonymously and efficiently using the Schnorr-like zero-knowledge protocol for proving the knowledge of a discrete logarithm [32]. For the conversion from the proof

[1] Revocation List

of knowledge to the signature, we use the Fiat-Shamir heuristics [41]. We present the concrete algorithm and protocol instantiations below.

Setup

$(pk, sk_m, par) \leftarrow$ `Setup`(1^κ): the algorithm inputs the security parameter κ and generates the bilinear group with parameters $par = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathbf{e}, g \in \mathbb{G}_1, g_2 \in \mathbb{G}_2)$ satisfying $|q| = \kappa$. It also generates the manager's private key $sk_m \xleftarrow{\$} \mathbb{Z}_q$ and computes the public key $pk = g_2^{sk_m}$. It outputs the (pk, par) as a public output and the sk_m as the manager's private output.

Register

$(sk_i, rd) \leftarrow$ `Register`(id_i, sk_m): the protocol is distributed between the user and the manager. The manager inputs his private key sk_m and the user inputs his private identifier id_i. The protocol outputs the wBB signature $sk_i = g^{\frac{1}{sk_m + id_i}}$ to the user and updates the manager's revocation database rd by storing id_i.

Sign

$sig(sk_i, m) \leftarrow$ `Sign`(m, id_i, sk_i): the algorithm inputs the user's private identifier id_i, his private key sk_i and the message to be signed. It outputs the signature $sig(sk_i, m)$ that consists of the following elements $(g', sk_i', \bar{sk}_i, \pi)$:

- $g' = g^r$: the generator raised to a randomly chosen randomizer $r \xleftarrow{\$} \mathbb{Z}_q$.
- $sk_i' = sk_i^r$: the users's private key raised to the randomizer.
- $\bar{sk}_i = sk_i'^{-id_i}$: the randomized private key raised to the user identifier.
- $\pi = SPK\{(id_i, r) : \bar{sk}_i = sk_i'^{-id_i} \wedge g' = g^r\}(m)$: proof of knowledge of r and id_i signing the message m.

Verify

$(0/1) \leftarrow$ `Verify`$(sig(sk_i, m), m, pk, bl)$: the algorithm inputs the massage m, its signature $(sig(sk_i, m)$, a blacklist bl and the public key pk. It checks the proof of knowledge signature π and checks that the signature is valid with respect to the manager's public key using the equation $\mathbf{e}(\bar{sk}_i g', g_2) \stackrel{?}{=} \mathbf{e}(sk_i', pk)$. The collector also performs the revocation check $\bar{sk}_i \stackrel{?}{=} sk_i'^{-id_i}$ for all id_i values stored on the blacklist bl. If the revocation check equation holds for any value on the blacklist, the signature is rejected. Otherwise, the signature is accepted if all other checks pass.

Revoke

$bl \leftarrow$ `Revoke`$(rd, sig(sk_i, m))$: the algorithm inputs a signature $sig(sk_i, m)$ and a revocation database rd. It checks $\bar{sk}_i \stackrel{?}{=} sk_i'^{-id_i}$ for all id_is in rd. The id_i that holds in the equation is put on a public blacklist bl.

The `Register`, `Sign` and `Verify` algorithms are presented in CS notation in Figure 14.

4.3 Implementation and Performance Analysis

We implemented the `Sign` and `Verify` protocols, the full description of our algorithms is in Figure 15. Our proposal is particularly suitable for data collections systems, such as smart metering. In these systems, the data are anonymously collected by a central collector from the remote nodes. Furthermore, due to the fast signature generation speed and size efficiency, our scheme can by used in a wide range of other applications, such as e-ticketing and transportation eIDs. For this reason, we performed the measurements on different kinds of devices, both constrained (wearables, embedded devices) and powerful (PC, server) ones.

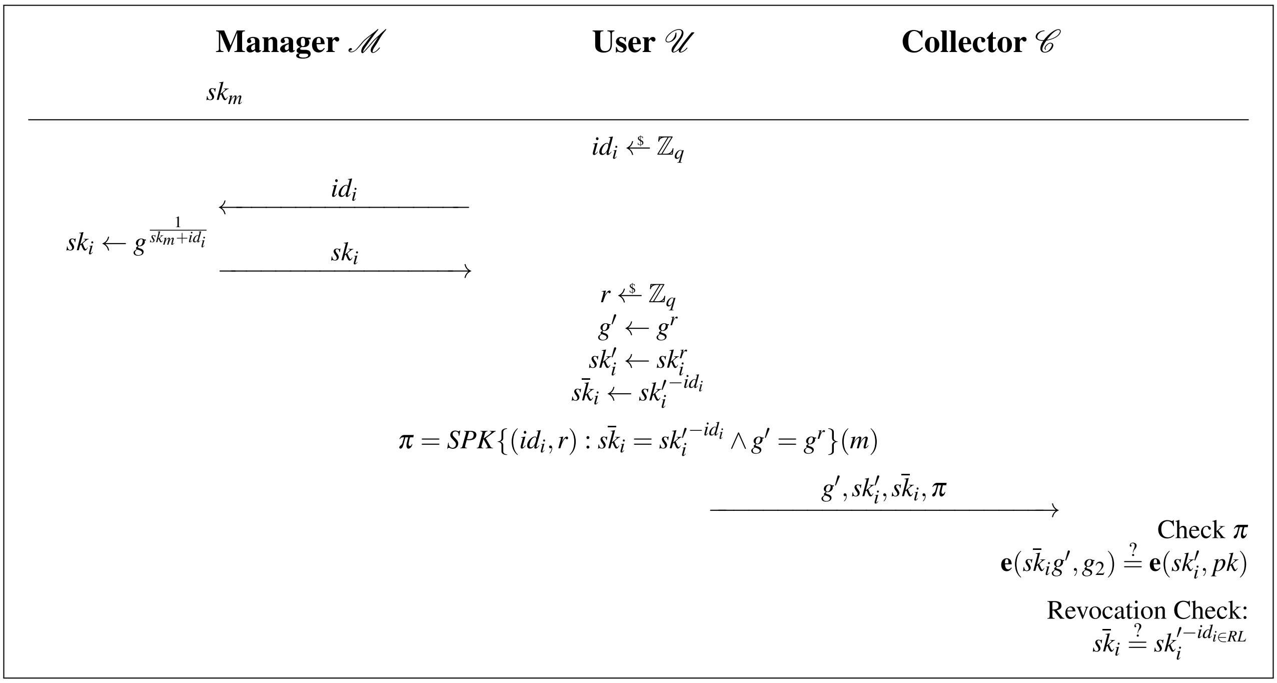

Manager $\mathscr{M}$
User $\mathscr{U}$
Collector $\mathscr{C}$
sk_m
$id_i \xleftarrow{\$} \mathbb{Z}_q$
id_i
$sk_i \leftarrow g^{\frac{1}{sk_m+id_i}}$
sk_i
$r \xleftarrow{\$} \mathbb{Z}_q$
$g' \leftarrow g^r$
$sk_i' \leftarrow sk_i^r$
$\bar{sk}_i \leftarrow sk_i'^{-id_i}$
$\pi = SPK\{(id_i, r) : \bar{sk}_i = sk_i'^{-id_i} \wedge g' = g^r\}(m)$
$g', sk_i', \bar{sk}_i, \pi$
Check π
$\mathbf{e}(\bar{sk}_i g', g_2) \stackrel{?}{=} \mathbf{e}(sk_i', pk)$
Revocation Check:
$\bar{sk}_i \stackrel{?}{=} sk_i'^{-id_{i\in RL}}$

Figure 14: `Register`, `Sign` and `Verify` algorithms.

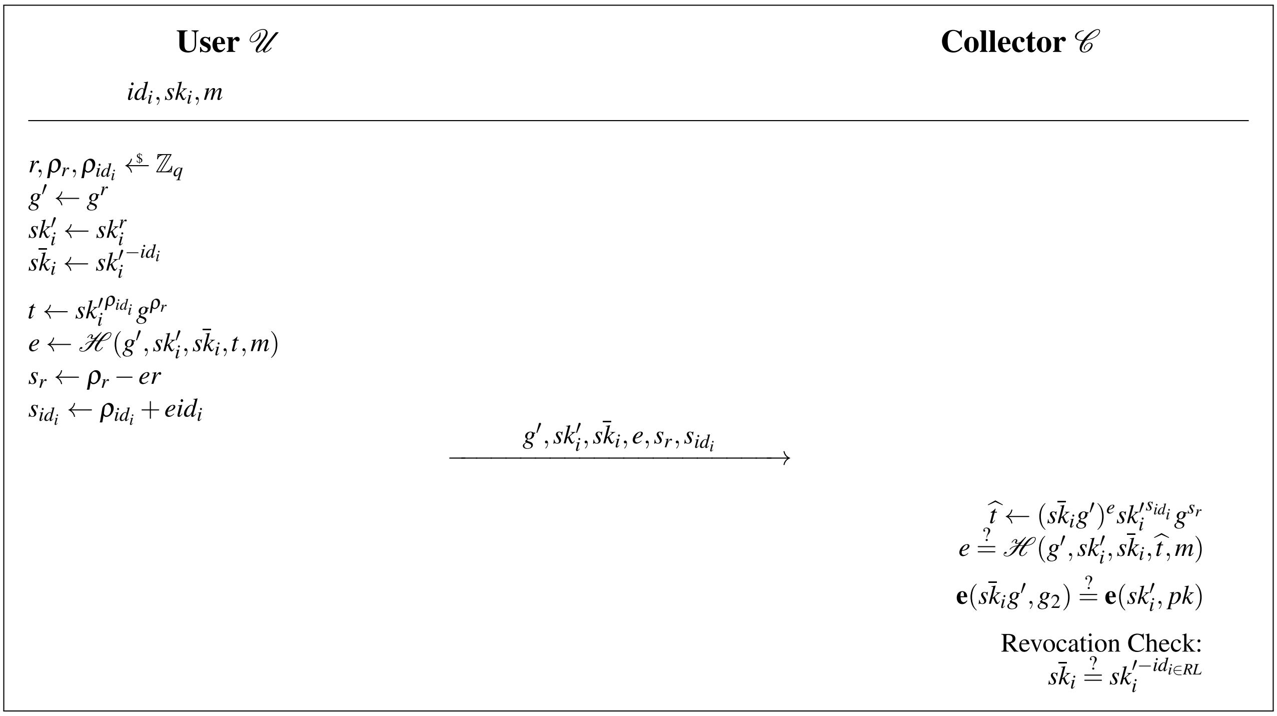

User $\mathscr{U}$
Collector $\mathscr{C}$
id_i, sk_i, m
$r, \rho_r, \rho_{id_i} \xleftarrow{\$} \mathbb{Z}_q$
$g' \leftarrow g^r$
$sk_i' \leftarrow sk_i^r$
$\bar{sk}_i \leftarrow sk_i'^{-id_i}$
$t \leftarrow sk_i'^{\rho_{id_i}} g^{\rho_r}$
$e \leftarrow \mathscr{H}(g', sk_i', \bar{sk}_i, t, m)$
$s_r \leftarrow \rho_r - er$
$s_{id_i} \leftarrow \rho_{id_i} + eid_i$
$g', sk_i', \bar{sk}_i, e, s_r, s_{id_i}$
$\hat{t} \leftarrow (\bar{sk}_i g')^e sk_i'^{s_{id_i}} g^{s_r}$
$e \stackrel{?}{=} \mathscr{H}(g', sk_i', \bar{sk}_i, \hat{t}, m)$
$\mathbf{e}(\bar{sk}_i g', g_2) \stackrel{?}{=} \mathbf{e}(sk_i', pk)$
Revocation Check:
$\bar{sk}_i \stackrel{?}{=} sk_i'^{-id_{i\in RL}}$

Figure 15: Implementation of `Sign` and `Verify` algorithms.

We performed the measurement on all devices mentioned above. The detailed hardware and software specifications are described in Table 7. The performance tests required the implementation of the proposed scheme on different platforms and operation systems. In case of the smart card application, only standard MultOS API and free public development environment (Eclipse IDE for C/C++ Developers, SmartDeck 3.0.1, MUtil 2.8) were used. The application is written in MULTOS assembly code and C language. Smart phones and smart watches run an Android application written in Java language. In particular, we used Android Studio 3.0.1 as the official IDE for Android app development along with Android SDK depending on the specific device, and jPBC-2.0.0 [36]

Table 7: Specification of tested devices.

Device	CPU/MCU	OS	RAM
Smart Card	SC23Z018	MultOS 4.3.1	1.75 KB
Phone 1	Kirin 655	Android 7.0	3 GB
Phone 2	Krait 400	Android 5.1	2 GB
Watch 1	ARM Cortex-A7	Android 6.0	512 MB
Watch 2	ARM Cortex-A7	Android 7.0	768 MB
Raspberry Pi 3	ARM Cortex-A53	Raspbian 9.3	1 GB
Raspberry Pi 2	ARM Cortex-A7	Raspbian 9.3	1 GB
Raspberry Pi	ARM1176JZF-S	Raspbian 9.3	512 MB
PC	Intel i7-7700	Debian 8.6	16 GB
Server	Intel Xeon 2.27	Debian 8.6	32 GB

Note: Smart Card – ML4, Phone 1 – HUAWEI P9 Lite 2017, Phone 2 – SONY Experia Z1 Compact, Raspberry Pi 3 – Raspberry Pi 3 Model B, Raspberry Pi 2 – Raspberry Pi 2 Model B, Raspberry Pi – Raspberry Pi Model B+, Watch 1 - Sony SmartWatch 3 SWR50, Watch 2 – HUAWEI Watch 2

library which allows performing operations over elliptic curves (point addition, scalar multiplication and bilinear pairing). The rest of the devices run OS Linux and, therefore, the scheme was implemented in C, where PBC-0.5.14 [51] library was used for the elliptic curve operations. The scheme was developed in NetBeans IDE 8.2 development environment. The code was remotely build and executed on the targeted device, i.e., Raspberry Pi/2/3, PC and server.

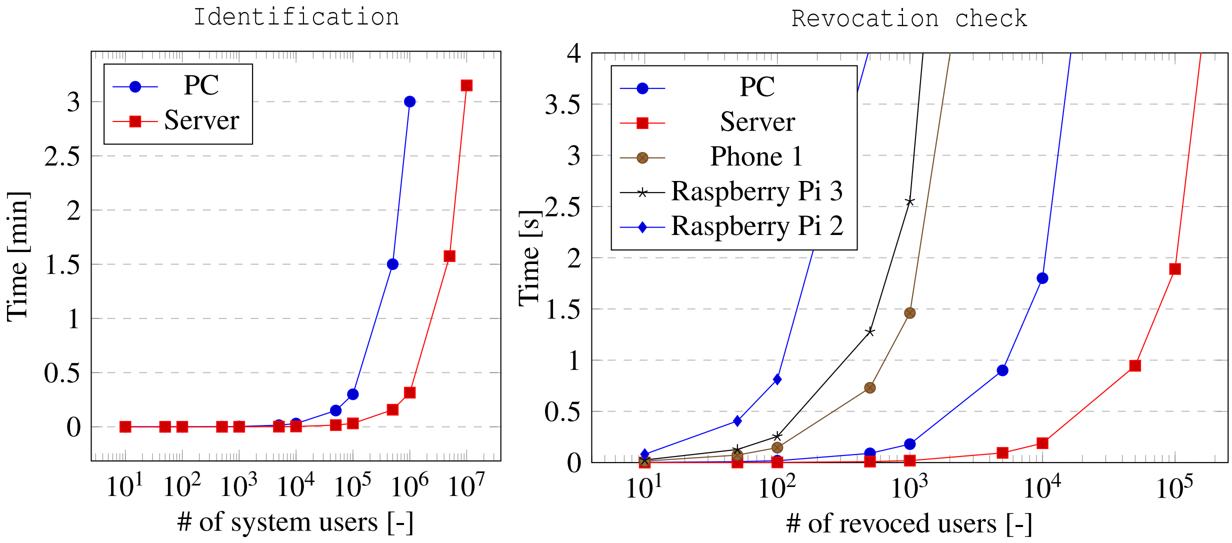
The `Sign` and `Verify` algorithms were implemented using pairing-friendly elliptic curves. Since our scheme requires asymmetric bilinear pairing, we considered the elliptic curves of D types from the PBC library, namely d159, d201, and d224. The performance tests were run 10 times on each device, and the arithmetic mean of the measured values was calculated. The computation time of `Sign` and `Verify` algorithms is provided in Table 8. At the first sight, the effectiveness of `Sign` protocol is obvious. Using the 224 b elliptic curve, which is of 112 b security strength, the `Sign` protocol takes only 442 ms on a smart card. On the other hand, the Android devices are slow in EC operations, in particular in bilinear pairing.

The Figure 16 shows the time needed to complete the malicious user identification and revocation check procedure. In case of the de-anonymisation procedure, the number of scalar multiplications is equal to the number of users. We stress, that the de-anonymisation procedure is expected to be performed on powerful devices and can be parallelized on their processors and cores (CPU/Cores). For instance, our PC (1/4), and server (2/8) are able go through the list of thousands of users and find the identity of a user in less than 4 min, see Figure 16. In the revocation check procedure, the PC (1/4) and server (2/8) are able to search the blacklist in less than 0.5 s.

Furthermore, we also provide the comparison of our scheme with the state of the art pairing and non-pairing based group signature schemes. We considered the efficient group signature schemes identified in [8]. Table 9 shows the comparison of our scheme with these pairing and non-pairing based group signature schemes.

Table 8: Performance of `Sign` and `Verify` protocols for different elliptic curves on user devices.

	Signing time [ms]			Verification time [s]		
Device/Curve	**d159**	**d201**	**d224**	**d159**	**d201**	**d224**
Smart Card	362	415	442	–	–	–
Phone 1	180	253	336	2.1	2.5	3.1
Phone 2	665	705	943	10.9	11.6	12.7
Watch 1	1252	2215	2889	26.2	31.0	38.0
Watch 2	1019	1139	1637	13.6	15.8	19.2
Raspberry Pi 3	18	24	30	0.082	0.115	0.138
Raspberry Pi 2	32	42	53	0.144	0.197	0.236
Raspberry Pi	67	89	110	0.266	0.372	0.434
PC	3	4	5	0.007	0.009	0.011



Identification

Revocation check

Figure 16: Time needed to identify a malicious user and to check the blacklist.

Table 9: Comparison with current short group signature schemes.

Scheme	**Sign Cost**	**Verify Cost**	**Signature Size**
BBS [24]	$9E_{\mathbb{G}_1}$+$3E_{\mathbb{G}_T}$	1**e**+$8E_{\mathbb{G}_1}$+$2E_{\mathbb{G}_2}$+$3E_{\mathbb{G}_T}$	$3\mathbb{G}_1$+$6\mathbb{Z}_p$ (1545 b)
DP [38]	$8E_{\mathbb{G}_1}$+$3E_{\mathbb{G}_T}$	1**e**+$7E_{\mathbb{G}_1}$+$2E_{\mathbb{G}_2}$+$3E_{\mathbb{G}_T}$	$4\mathbb{G}_1$+$5\mathbb{Z}_p$ (1559 b)
HLCCN [47]	$7E_{\mathbb{G}_1}$+$5E_{\mathbb{G}_T}$	1**e**+$5E_{\mathbb{G}_1}$+$2E_{\mathbb{G}_2}$+$4E_{\mathbb{G}_T}$	$3\mathbb{G}_1$+$5\mathbb{Z}_p$ (1375 b)
ACJT [15]	$12E_{\mathbb{G}_n^*}$	$10E_{\mathbb{G}_n^*}$	$7\mathbb{G}_n^*$+$1\mathbb{Z}_c$ (7328 b)
CG [28]	$10E_{\mathbb{G}_n^*}$	$10E_{\mathbb{G}_n^*}$	$8\mathbb{G}_n^*$+$1\mathbb{Z}_q$ (8352 b)
IMSTY [49]	$7E_{\mathbb{G}_n^*}$	$7E_{\mathbb{G}_n^*}$	$5\mathbb{G}_n^*$+$5\mathbb{Z}_p$+$1\mathbb{Z}_c$ (6155 b)
HM GS [45]	$9E_{\mathbb{G}_n^*}$	$10E_{\mathbb{G}_n^*}$	$7\mathbb{G}_n^*$+$1\mathbb{Z}_q$ (7328 b)
Our Scheme	$5E_{\mathbb{G}_1}$	2**e**+$3E_{\mathbb{G}_1}$	$3\mathbb{G}_1$+$3\mathbb{Z}_p$ (1035 b)

Note: $E_{\mathbb{G}_1}$ – EC scalar multiplication in $\mathbb{G}_1$, similarly $E_{\mathbb{G}_2}$ and $E_{\mathbb{G}_T}$, **e** – bilinear pairing.

5 ANONYMOUS CREDENTIALS WITH PRACTICAL REVOCATION

The content of this section have been published in conference paper [2]. In this section, we present a novel elliptic curve variant of the HM12 attribute-based credential scheme [43]. The elliptic curve variant (we call it ecHM12) meets all requirements for an ABC scheme as well as the original scheme HM12. In particular, we preserved (1) **privacy-enhancing** features: anonymity, untraceability, unlinkability, (2) **non-transferability**, (3) **selective disclosure of attributes**, (4) computationally **efficient revocation** and malicious user identification. Furthermore, by involving elliptic curves to the scheme, we achieve (5) **higher computational efficiency** compared with the standard HM12 scheme, especially during the `Prove_Att` phase. The ecHM12 scheme also requires (6) **smaller bandwidth**, since data communication transfer is 85% smaller compared to the original scheme HM12.

5.1 General Architecture

Four types of entities interact in our ABC scheme: a user, a manager and a Verifier. The user gets issued attributes from the issuer and anonymously proves their possession to the verifier. The issuer is responsible for issuing user attributes. The manager validates user credentials (collection of attributes issued by the issuer), can revoke a (dishonest) user, and in collaboration with the issuer, can identify the (dishonest) users. The verifier verifies possession of required attributes provided by users. Each entity communicates in the system through specific cryptographic protocols. All the protocols and involved entities are depicted in the Figure 17.

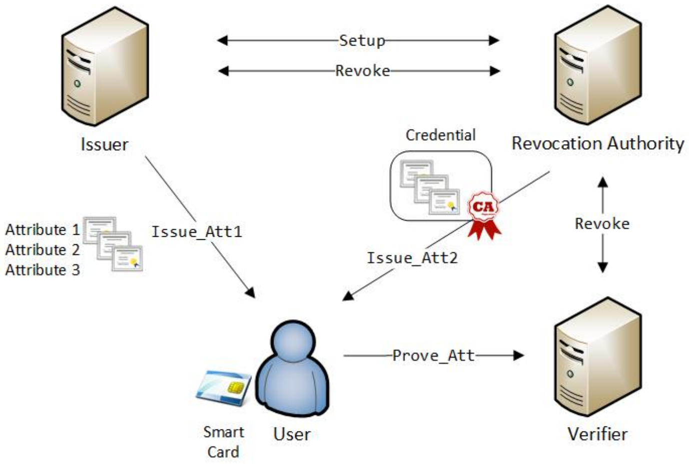


Figure 17: Architecture of proposed ecHM12 scheme.

5.2 Cryptography Specification

In this section we provide a detailed description of each protocol which runs within the proposed scheme.

Setup

$(par, K_{\mathscr{M}}, K_{\mathscr{I}}) \leftarrow \texttt{Setup}(1^{\kappa})$: the `Setup` protocol mostly matches the original HM12 scheme, only in the final step the scheme is switched to the elliptic curve variant. The main purpose of this protocol is to establish system parameters *par* and to generate $K_{\mathscr{I}}$ and $K_{\mathscr{M}}$ keys. The input parameter κ defines the security strength of the cryptographic scheme, similarly to the scheme HM12. Additionally, κ includes elliptic curve domain parameters bitlengths. The Issuer defines a group $\mathbb{H}$ modulo big prime number p and generators h_1, h_2 of order q, with $q|p-1$. $\mathbb{H}$ is the subgroup of the group $\mathbb{Z}_p^*$ as in the DSA signature scheme. In addition, the Issuer generates the key pair $sk_{\mathscr{I}} = K_{\mathscr{I}}$ and $pk_{\mathscr{I}}$ for signing purpose using a defined signature scheme, e.g. RSA. The Manager needs to:

- define the Okamoto-Uchiyama group $\mathbb{OU}_n$ by specifying the modulus $n = r^2 s$, where r and s are secure primes (i.e. $r = 2r' + 1$, $s = 2s' + 1$, where r' and s' are primes),
- find a generator $g_1 \xleftarrow{\$} \mathbb{Z}_n^*$ of $ord(g_1 \bmod r^2) = r(r-1)$ in $\mathbb{Z}_{r^2}^*$ and $ord(g_1) = rr's'$ in $\mathbb{Z}_n^*$,
- choose an elliptic curve over finite field $E(\mathbb{F}_p)$ with the domain parameters (a, b, p, q, G, h), where p is big prime number specifying the field $\mathbb{F}_p$, $a, b \in \mathbb{F}_p$ are static coefficients of the E, G is curve point generator $G = (x_G, y_G)$ of order q, and h is the cofactor defined as $h = \#E(\mathbb{F}_p)/q$,
- randomly choose Master's secrets $\langle s_{1,i} \rangle_{i \in \mathscr{A}} \xleftarrow{\$} \mathbb{Z}_q$ for available attributes, and $s_2, s_3 \xleftarrow{\$} \mathbb{Z}_q$, such that $\langle GCD(s_{1,i}, q) = 1 \rangle_{i \in \mathscr{A}}$, $GCD(s_2, q) = 1$, $GCD(s_3, q) = 1$.
- compute second generator $g_2 \leftarrow g_1^{s_2} \bmod n$ in the $\mathbb{OU}_n$,
- set first curve generator $G_1 \leftarrow G$, generate all attributes $\langle eca_i \leftarrow s_{1,i} \bullet G_1 \rangle_{i \in \mathscr{A}}$, and finally, get second and third curve generators $G_2 \leftarrow s_2 \bullet G_1$, $G_3 \leftarrow s_3 \bullet G_1$ in $E(\mathbb{F}_p)$.

The system parameters $par = (g_1, g_2, \mathbb{OU}_n, h_1, h_2, \mathbb{H}, G_1, G_2, G_3, E(\mathbb{F}_p))$ together with the set of attributes $\langle eca_i \rangle_{i \in \mathscr{A}}$ are made public, while the values r, s and $\langle s_{1,i} \rangle_{i \in \mathscr{A}}, s_2, s_3$ represent the Manager's secret key $K_{\mathscr{M}}$ and are securely stored by the Manager, and the secret key $K_{\mathscr{I}}$ is securely stored by the Issuer.

Issue_Att

$(K_U) \leftarrow$ `Issue_Att`$(par, K_{\mathscr{M}}, K_{\mathscr{I}})$: the protocol follows the HM12 idea. The issue phase is split into two parts `Issuer_Att1` and `Issuer_Att2` protocols, see Figure 18. The goal is to compute the User's key $K_U = \{w_1, w_2, \langle w_{3,i} \rangle_{i \in \mathscr{A}}\}$.

`Issuer_Att1` is run between the User and the Issuer. The User generates a cryptographic commitment $\bar{H} = h_1^{w_1} h_2^{w_2} \bmod p$ in $\mathbb{H}$, where the User's keys w_1, w_2 are committed values. Then, the User signs the commitment with his private key $sk_{\mathscr{U}}$ and sends it and the signature with the proof of construction PK_{U1} to the Issuer. The Issuer checks the signature and the proof and signs User's commitments by his private key $K_{\mathscr{I}}$. Commitments are stored by the Issuer for identification and revocation purposes. Any secure signature scheme, e.g. RSA, DSA, can be used.

`Issuer_Att2` is run between the User and the Manager. The User computes another commitment $\bar{A} = g_1^{w_1} g_2^{w_2} \bmod n$ in $\mathbb{OU}_n$ and sends $\bar{A}$, $\bar{H}$, the signature of $\bar{H}$ (generated by the Issuer) and the proof of discrete logarithm equivalence PK_{U2} to the Manager. Now, the Manager is able to compute the User's partial keys $\langle w_{3,i} \rangle_{i \in \mathscr{A}}$ for all attributes $\langle eca_i \rangle_{i \in \mathscr{A}}$, such that the following equalities hold:

$$\begin{aligned} eca_i &= w_1 \bullet G_1 + w_2 \bullet G_2 + w_{3,i} \bullet G_3 \\ &= w_1 \bullet G_1 + w_2 \cdot s_2 \bullet G_1 + w_{3,i} \cdot s_3 \bullet G_1 \\ &= (w_1 + w_2 \cdot s_2 + w_{3,i} \cdot s_3) \bullet G_1 \\ &= s_{1,i} \bullet G_1 = eca_i \end{aligned} \tag{1}$$

The values $\bar{A}, \bar{H}$ and $\langle w_{3,i} \rangle_{i \in \mathscr{A}}$ are stored in the Manager's database and sent to the User. The User securely stores his key $K_U = \{w_1, w_2, \langle w_{3,i} \rangle_{i \in \mathscr{A}}\}$, e.g. on a smart card.

Prove_Att

$(0/1) \leftarrow$ `Prove_Att`(par, K_U): this protocol is run fully over $E(\mathbb{F}_p)$. The protocol is depicted in Figure 19. The User proves the ownership of attributes $\langle eca_i \rangle_{i \in \mathscr{D}}$ to the Verifier using PK protocols. The

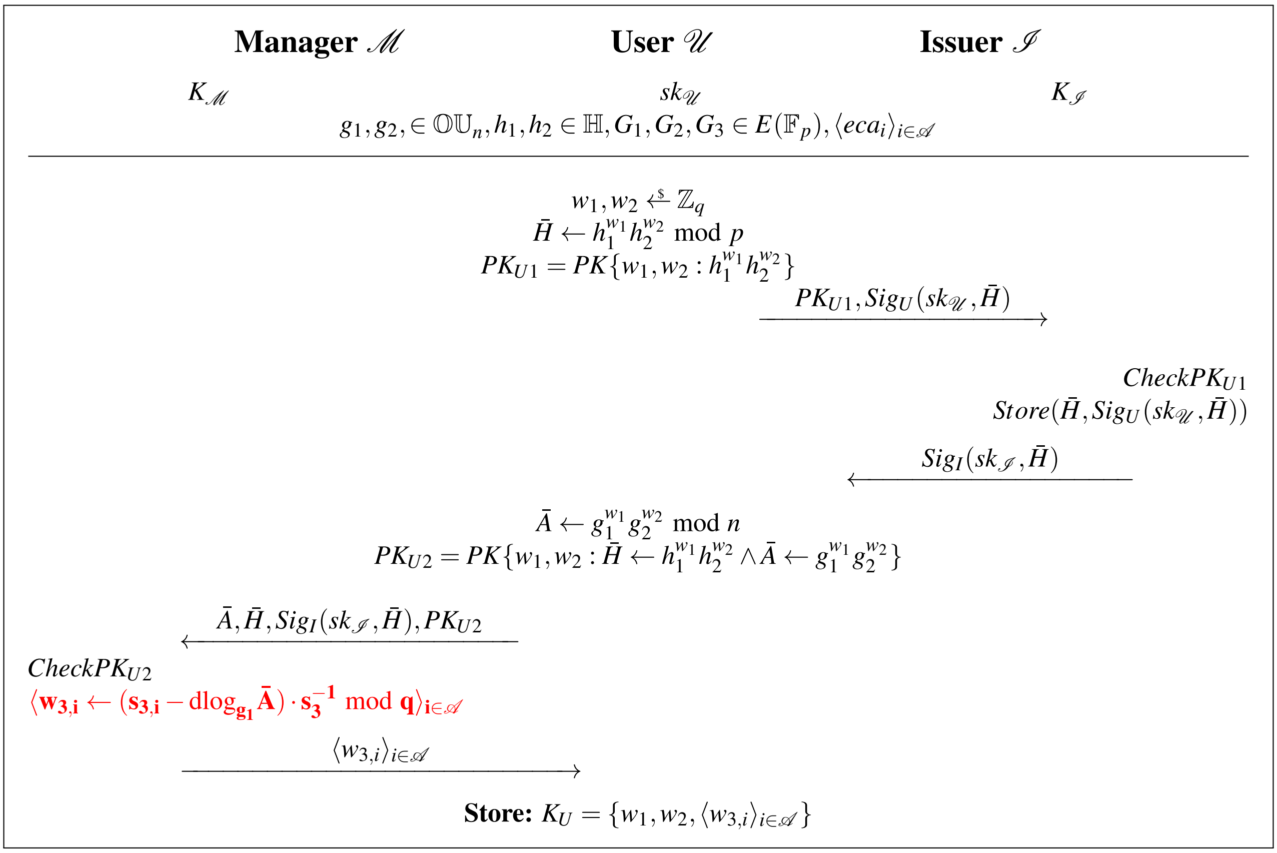

Manager $\mathscr{M}$
User $\mathscr{U}$
Issuer $\mathscr{I}$
$K_{\mathscr{M}}$
$sk_{\mathscr{U}}$
$K_{\mathscr{I}}$
$g_1, g_2, \in \mathbb{OU}_n, h_1, h_2 \in \mathbb{H}, G_1, G_2, G_3 \in E(\mathbb{F}_p), \langle eca_i \rangle_{i \in \mathscr{A}}$
$w_1, w_2 \xleftarrow{\$} \mathbb{Z}_q$
$\bar{H} \leftarrow h_1^{w_1} h_2^{w_2} \bmod p$
$PK_{U1} = PK\{w_1, w_2 : h_1^{w_1} h_2^{w_2}\}$
$PK_{U1}, Sig_U(sk_{\mathscr{U}}, \bar{H})$
$CheckPK_{U1}$
$Store(\bar{H}, Sig_U(sk_{\mathscr{U}}, \bar{H}))$
$Sig_I(sk_{\mathscr{I}}, \bar{H})$
$\bar{A} \leftarrow g_1^{w_1} g_2^{w_2} \bmod n$
$PK_{U2} = PK\{w_1, w_2 : \bar{H} \leftarrow h_1^{w_1} h_2^{w_2} \wedge \bar{A} \leftarrow g_1^{w_1} g_2^{w_2}\}$
$\bar{A}, \bar{H}, Sig_I(sk_{\mathscr{I}}, \bar{H}), PK_{U2}$
$CheckPK_{U2}$
$\langle \mathbf{w_{3,i}} \leftarrow (\mathbf{s_{3,i}} - \mathrm{dlog}_{\mathbf{g_1}} \bar{\mathbf{A}}) \cdot \mathbf{s_3^{-1}} \bmod \mathbf{q} \rangle_{\mathbf{i} \in \mathscr{A}}$
$\langle w_{3,i} \rangle_{i \in \mathscr{A}}$
Store: $K_U = \{w_1, w_2, \langle w_{3,i} \rangle_{i \in \mathscr{A}}\}$

Figure 18: `Issue_Att` protocol of the ecHM12 scheme.

unlinkability is provided by using the random number K_S, which is re-generated in every session. Moreover, the protocol provides revocation features by committing the value K_S in the commitment C_2 and the committed values $\langle w_{3,i} \rangle_{i \in \mathscr{D}}$ (revocable key parts of the User's key) in the commitments $\langle C_{1,i} \rangle_{i \in \mathscr{D}}$. The commitments $\langle C_{1,i} \rangle_{i \in \mathscr{D}}$ and C_2 permit to check if the User is in the blacklist or not, and to remove him from the system by involving the Manager in the revocation process. The verification time depends on the number of disclosed attributes by the User and on the number of all revoked Users.

`Revoke`

$(BL) \leftarrow$ `Revoke`$(par, proof, K_{\mathscr{M}})$: the original HM12 scheme uses the OU trapdoor to solve the discrete logarithm problem. In ecHM12 scheme, this trapdoor cannot be used. However, revocation of a dishonest user is still possible. The protocol input parameters are system parameters *par* and *proof* generated by the User within the `Prove_Att` protocol. The revocation part of the *proof* consists of commitments $\langle C_{1,i} \rangle_{i \in \mathscr{D}}$ and C_2. The Manager computes Equation 2 for all user keys $w_{3,DATABASE}$ in Manager's database until a match is found.

$$\langle w_{3,DATABASE} \bullet C_2 \overset{?}{=} C_{1,i} \rangle_{i \in \mathscr{R}} \tag{2}$$

If a match is found, the commitment that belongs to this particular User is revoked by publishing $\langle w_{3,i} \rangle_{i \in \mathscr{R}}$ (where $\mathscr{R}$ donates a subset of revoked attributes) on a blacklist (BL). The revocation complexity is linear in the number of Users instead of constant as in the HM12 scheme. Yet revocation remains practical, see Section 5.3 for implementation details. On the other hand, the protocol `Prove_Att` is faster than in the HM12 scheme.

User $\mathscr{U}$ **Verifier $\mathscr{V}$**

K_U $\quad G_1, G_2, G_3 \in E(\mathbb{F}_p), \langle eca_i \rangle_{i \in \mathscr{D}}$

$$\xleftarrow{\quad nonce \quad} \qquad nonce \xleftarrow{\$} \mathbb{Z}_q$$

$K_S, r_1, r_2, r_3, r_S \xleftarrow{\$} \mathbb{Z}_q$

$a \leftarrow \Sigma_{i \in \mathscr{D}} eca_i, \quad \bar{a} \leftarrow r_1 \bullet G_1 + r_2 \bullet G_2 + r_3 \bullet G_3$

$A \leftarrow K_S \bullet a, \quad \bar{A} \leftarrow r_S \bullet a$

$\langle C_{1,i} \leftarrow (K_S \cdot w_{3,i}) \bullet G_3 \rangle_{i \in \mathscr{D}}, \quad C_2 \leftarrow K_S \bullet G_3$

$\bar{C}_1 \leftarrow r_3 \bullet G_3, \quad \bar{C}_2 \leftarrow r_S \bullet G_3$

$e \leftarrow \mathscr{H}(nonce, a, \bar{C}_1, \bar{a}, A, \bar{A}, C_2, \langle C_{1,i} \rangle_{i \in \mathscr{D}}, \bar{C}_2)$

$z_1 \leftarrow (r_1 - eK_S|\mathscr{D}|w_1) \bmod q$

$z_2 \leftarrow (r_2 - eK_S|\mathscr{D}|w_2) \bmod q$

$z_3 \leftarrow (r_3 - eK_S \Sigma_{i \in \mathscr{D}} w_{3,i}) \bmod q$

$z_S \leftarrow (r_S - eK_S) \bmod q$

$$\xrightarrow{A, \langle C_{1,i} \rangle_{i \in \mathscr{D}}, C_2, e, z_1, z_2, z_3, z_S}$$

Check BL: $\langle C_2 \bullet w_{3,blacklisted} \overset{?}{=} C_{1,i} \rangle_{i \in \mathscr{D}}$

$\bar{a} \leftarrow e \bullet A + z_1 \bullet G_1 + z_2 \bullet G_2 + z_3 \bullet G_3$

$a \leftarrow \Sigma_{i \in \mathscr{D}} eca_i, \quad \bar{A} \leftarrow e \bullet A + z_S \bullet a$

$\bar{C}_1 \leftarrow e \bullet \Sigma_{i \in \mathscr{D}} C_{1,i} + z_3 \bullet G_3, \quad \bar{C}_2 \leftarrow e \bullet C_2 + z_S \bullet G_3$

$e \overset{?}{=} \mathscr{H}(nonce, a, \bar{C}_1, \bar{a}, A, \bar{A}, C_2, \langle C_{1,i} \rangle_{i \in \mathscr{D}}, \bar{C}_2)$

Figure 19: `Prove_Att` protocol of the ecHM12 scheme.

5.3 Implementation and Performance Analysis

We provide full protocol implementation (proof of concept implementation). The protocol was implemented on MultOS ML4 smart card in 192-bit version (i.e. NIST curve P-192 was used). The implementation supports up to 5 attributes issued. The performance test results for increasing number of disclosed attributes are depicted in Figure 20. The time for one attribute possession proving takes around 1 s (including communication overhead) and ca. 2 s to prove possession of all 5 attributes.

Moreover, we provide comparison of our scheme implementation with implementation of the origin scheme HM12 (1024-bit version), see Section 2.2. The results show a significant time reduction (by 20% within on-card computation time and almost by 40% in total, i.e. including communication between the card and the reader), since we use more efficient elliptic curve operations and transmit a smaller amount of data. Important to note, that our implementation holds significantly higher security level (1776-bit group equivalent according to [68] instead of 1024-bit group of the HM12 implementation). With increasing security strength of the protocols we can expect much bigger difference in attribute proving time and bandwidth usage.

The elliptic curve construction reduces data transmission cost, since we need to transfer only

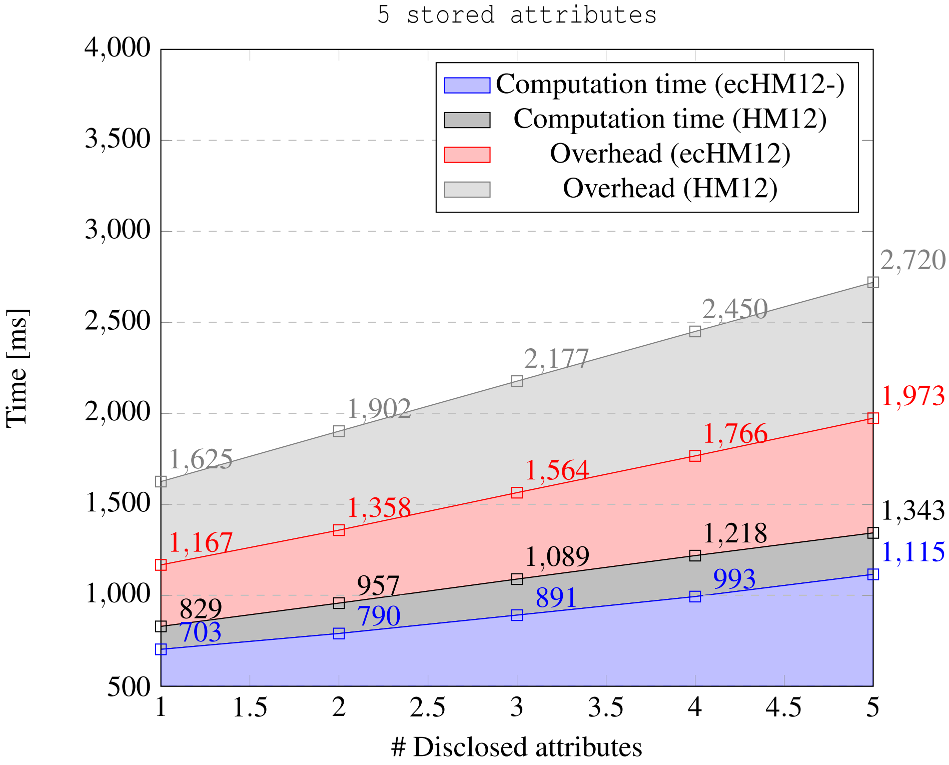


Figure 20: Scheme ecHM12 attributes proving time for different scenarios.

264 B in case of $E(\mathbb{F}_{192})$ instead of 1558 B in the original scheme (1392-bit version) in the `Prove_Att` protocol. On the Verifier side, the time needed for checking blacklist is also more efficient in the ecHM12 scheme than in the HM12 scheme because of the involved operations: ecHM12 uses scalar multiplication and HM12 uses the slower modular exponentiation.

For the ecHM12 scheme, the revocation mechanism complexity is linear instead of constant as in the HM12 scheme. However, we expect Manager to be computationally strong and, consequently, the slow-down does not really affect the protocol complexity. We use oldish mid-range server, namely the 2009 IBM x3550 M2 with two Intel Xeon 2.27 GHz processors with 8 cores each and 32 GB RAM, to represent the Manager. The elliptic curve scalar multiplication over $E(\mathbb{F}_{224})$ took negligible 0.0189 ms, i.e. with $100,000$ users in the system, the revocation time will be ca. 1.9 s at maximum.

6 FAST KEYED-VERIFICATION ANONYMOUS CREDENTIALS

The content of this section have been published in conference paper [1]. In this section, we propose a novel cryptographic scheme for anonymous attribute-based credentials that is designed primarily for smart cards. The scheme is based on our original algebraic MAC that makes its proving protocol very efficient. The computational complexity of our proving protocol is the lowest from related schemes (only $u+2$ scalar multiplications to present an attribute ownership proof) and we need only basic arithmetic operations that are already provided by existing smart cards' APIs. We present the results of the full implementation of our proving protocol that is faster by at least 44 % than the state of the art implementation. By reaching the time of 366 ms including overhead, which is required for proving personal attributes on a 192-bit EC security level, we argue that the anonymous credentials are finally secure and practical even for time-critical and large-scale applications like eIDs, e-ticketing and mass transportation.

In contrast to traditional anonymous attribute-based credential schemes, the verifier needs to know the secret keys to be able to verify user's attributes in Keyed-Verification Anonymous Credential (KVAC) schemes. This feature is particularly convenient for scenarios where attribute issuers and attribute verifiers are the same entities. The mass transportation settings is an example of such a scenario because the transportation authority both issues and checks the tickets and passes. Our KVAC scheme supports (1) all the standard **privacy-enhancing** features of ABC schemes, such as anonymity, unlinkability, untraceability, (2) **selective disclosure of attributes**, it is (3) **compatible with** major credential schemes [33, 58] and **standard revocation schemes** [30, 27], and it is (4) the **fastest** scheme from the current state of the art of ABC schemes.

6.1 General Architecture

The communication pattern is presented in Figure 21 and employs: a user, a issuer and a verifier. The user gets issued attributes from an Issuer and anonymously proves their possession to the verifier. The issuer is responsible for issuing attributes to a user. A issuer signs the user attributes with it (issuer) secret key. The verifier verifies a possession of required attributes by the user. The verifier requires to have a issuer secret key, however, this necessity do not create any security risk, since we assume that the issuer and the verifier are the same entity.

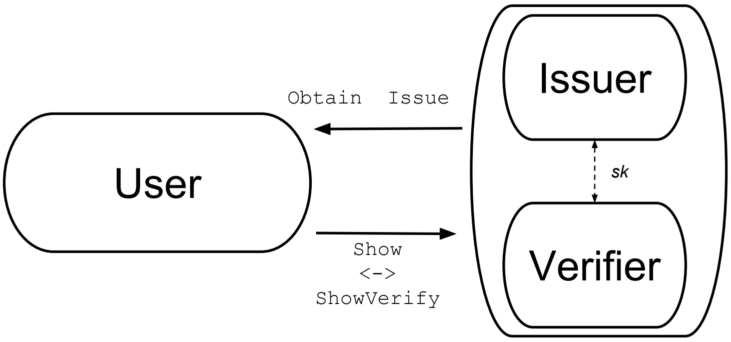


Figure 21: Architecture of keyed-verification anonymous credentials.

6.2 Cryptography Specification

Our scheme is parametrized by n, the amount of attributes in a credential. We describe our scheme using *selective disclosure* as attribute predicates, i.e., a predicate ϕ can be seen as a set $\mathscr{D} \subseteq \{1,\ldots,n\}$

containing the indices of the disclosed attributes and the attribute values of the disclosed attributes $\langle m_i \rangle_{i\in\mathscr{D}}$. One novel trick allows us to strongly improve the efficiency of our scheme. Instead of computing a standard noninteractive Schnorr-type proof of knowledge, we use the fact that the verifier knows the secret key. This allows us to omit elements that the verifier can compute by itself and saves the prover a lot of work.

We note that our `Issue` algorithm does not support the efficient issuance of committed attributes. This feature is useful in applications where a user needs to transfer his attributes among credentials or needs to get issued attributes that are only private to him. However, we consider these scenarios rare in targeted applications such as e-ticketing, mass transportation and loyalty cards. In those cases, the personal attributes (i.e., ticket type, pass validity period, registration number) are known to issuer or might even be chosen by the issuer. However, if the issuance of undisclosed attributes is necessary, it can be done by employing Paillier encryption [57], as is shown in [17].

Setup

$(par) \leftarrow \texttt{Setup}(1^\kappa)$: protocol outputs $par = (\mathbb{G}, g, q) \leftarrow \texttt{GroupSetup}(1^\kappa)$.

$(sk, ipar) \leftarrow \texttt{CredKeygen}(par)$: this protocol runs $(sk, ipar) \leftarrow \mathsf{MAC}_{\mathsf{wBB}}.\texttt{KeyGen}(par)$ and outputs sk and $ipar$.

Issue_Att

$(cred) \leftarrow \texttt{Issue}(sk, (m_1, \dots, m_n))$: runs $(\sigma, \langle \sigma_{x_i} \rangle_{i=0}^n) \leftarrow \mathsf{MAC}_{\mathsf{wBB}}.\texttt{MAC}(sk, (m_1, \dots, m_n))$. Next, provides a proof that allows a user to verify the validity of the credential: $\pi \leftarrow SPK\{(x_0, \dots, x_n) : \bigwedge_{i=0}^n \sigma_{x_i} = \sigma^{x_i} \wedge X_i = g^{x_i}\}$. The algorithm outputs credential $cred \leftarrow (\sigma, \langle \sigma_{x_i} \rangle_{i=0}^n, \pi)$.

$(0/1) \leftarrow \texttt{Obtain}(ipar, cred, (m_1, \dots, m_n))$: parses $ipar$ as $(X_0, \dots, X_n)$ and parses $cred$ as $(\sigma, \langle \sigma_{x_i} \rangle_{i=0}^n, \pi)$. The algorithm checks that $\sigma_{x_0} \cdot \prod_{i=1}^n \sigma_{x_i}^{m_i} = g$ and verifies π with respect to $ipar$ and σ.

Prove_Att

$(proof) \leftarrow \texttt{Show}(ipar, cred, (m_1, \dots, m_n), (\mathscr{D}, \langle m_i \rangle_{i\in\mathscr{D}}))$: in credential presentation, we want to let the user prove posession of a valid credential with the desired attributes. On a high level, we want to prove knowledge of a weak Boneh-Boyen signature, so we can apply the efficient proof due to Arfaoui et al. [14] and Camenisch et al. [27], by extending it to support a vector of messages: Take a random $r \xleftarrow{\$} \mathbb{Z}_q$ and let $\hat{\sigma} \leftarrow \sigma^r$ and $\hat{\sigma}_{x_i} \leftarrow \sigma_{x_i}^r$ for $i = 0, \dots, n$, and prove

$$proof = SPK\{(\langle m_i \rangle_{i\notin\mathscr{D}}, r) : \hat{\sigma}_{x_0} \prod_{i\in\mathscr{D}} \hat{\sigma}_{x_i}^{m_i} = g^r \prod_{i\notin\mathscr{D}} \hat{\sigma}_{x_i}^{-m_i}\}.$$

The verifier simply checks that the $\hat{\sigma}_{x_i}$ values are correctly formed and verifies the proof.

While this approach is secure and conceptually simple, it is not very efficient. We now present how we can construct a similar proof in a much more efficient manner. The key observation is that the user does not have to compute anything that the verifier, who is in possession of the issuer secret key sk, can compute. This means we can omit the computation of the $\hat{\sigma}_{x_i}$ values and define `Show` as follows. Randomize the credential by taking a random $r \leftarrow \mathbb{Z}_q^*$ and setting $\hat{\sigma} \leftarrow \sigma^r$. Take $\rho_r, \rho_{m_{i\notin\mathscr{D}}} \xleftarrow{\$} \mathbb{Z}_q$ and compute

$$t = \prod_{i\notin\mathscr{D}} \sigma_{x_i}^{\rho_{m_i} \cdot r} g^{\rho_r}, c = \mathscr{H}(t, \hat{\sigma}, par, ipar), s_r = \rho_r + cr, \langle s_{m_i} = \rho_{m_i} - cm_i \rangle_{i\notin\mathscr{D}}.$$

Send $(\hat{\sigma}, c, s_r, \langle s_{m_i} \rangle_{i\notin\mathscr{D}})$ to the verifier.

$(0/1) \leftarrow \texttt{ShowVerify}(sk, (\mathscr{D}, \langle m_i \rangle_{i \in \mathscr{D}}), proof)$: the verifier running ShowVerify will receive $proof = (\hat{\sigma}, c, s_r, \langle s_{m_i} \rangle_{i \notin \mathscr{D}})$ from the user. It computes

$$t \leftarrow g^{s_r} \cdot \hat{\sigma}^{-c \cdot x_0 + \sum_{i \notin \mathscr{D}}(x_i \cdot s_{m_i}) - \sum_{i \in \mathscr{D}}(x_i \cdot m_i \cdot c)}$$

and checks that $c = \mathscr{H}(t, \hat{\sigma}, par, ipar)$. Output 1 if valid and 0 otherwise. The Show and ShowVerify algorithms are depicted in Figure 22.

User $\mathscr{U}$		**Verifier $\mathscr{V}$**
$\langle m_i \rangle_{i=1}^{n}, \sigma, \langle \sigma_{x_i} \rangle_{i=0}^{n}, \mathscr{D}$	$\mathbb{G}, g, q$	$\langle x_i \rangle_{i=0}^{n}, \mathscr{D}, \langle m_i \rangle_{i \in \mathscr{D}}$
$r, \rho_r, \rho_{m_{i \notin \mathscr{D}}} \xleftarrow{\$} \mathbb{Z}_q$		
$\hat{\sigma} \leftarrow \sigma^r$		
$t \leftarrow \prod_{i \notin \mathscr{D}} \sigma_{x_i}^{\rho_{m_i} \cdot r} g^{\rho_r}$		
$c \leftarrow \mathscr{H}(t, \hat{\sigma}, par, ipar)$		
$s_r \leftarrow \rho_r + cr$		
$\langle s_{m_i} \leftarrow \rho_{m_i} - cm_i \rangle_{i \notin \mathscr{D}}$		
	$\xrightarrow{\hat{\sigma}, c, s_r, \langle s_{m_i} \rangle_{i \notin \mathscr{D}}}$	
		$t \leftarrow g^{s_r} \cdot \hat{\sigma}^{-c \cdot x_0 + \sum_{i \notin \mathscr{D}}(x_i \cdot s_{m_i}) - \sum_{i \in \mathscr{D}}(x_i \cdot m_i \cdot c)}$
		Check $c \overset{?}{=} \mathscr{H}(t, \hat{\sigma}, par, ipar)$

Figure 22: Definition of the Show and ShowVerify algorithms of our KVAC scheme.

6.3 Implementation and Performance Analysis

The Show and ShowVerify algorithms of our scheme were implemented using a standard NIST P-192 curve [11]. We stress that this selection of parameters reflects contemporary recommendations regarding security levels, unlike other implementations of anonymous credentials that use mostly small modular groups. Only standard MultOS API and free public development environment (Eclipse IDE for C/C++ Developers, SmartDeck 3.0.1, MUtil 2.8) were used. For terminal application, Java BigInteger class and BouncyCastle API were used. We compare our results (blue and red) with the state of the art results of Vullers and Alpár (VA) [70] (black and white) for different numbers of attributes stored and disclosed in Figure 23. We note that our implementation uses significantly higher security parameters (1024-bit vs. 1776-bit DSA group equivalent according to [68]).

The algorithm time (blue) tells the time necessary to compute all algorithms on the card. The overhead time (red) adds time necessary to do all the supporting actions, mainly establishing the communication with a reader connected to PC and transferring APDUs. All results are arithmetic means of 10 measurements in milliseconds[2]. Compared to VA's implementation of Idemix, our implementation of all proving protocol algorithms on the card is at least 44% faster in all cases, see Figure 23 for details. In the case of only 2 attributes stored on the card, our scheme is by 72 % faster than VA's implementation. The card needs only 211 ms to compute the ownership proof for disclosed attributes. The total time of around 360 ms necessary for the whole proof generation on the card including communication with and computations on a terminal (standard PC, Core i7 2.4

[2] Unlike microcontrollers and CPUs, smart card SDKs do not provide public tools for the measurement of clock cycles. Furthermore, the conversion between the number of cycles per an operation and it's execution time is difficult due to cards' variable clock speed. Therefore, the performance is usually measured in milliseconds [44, 53, 70, 37].

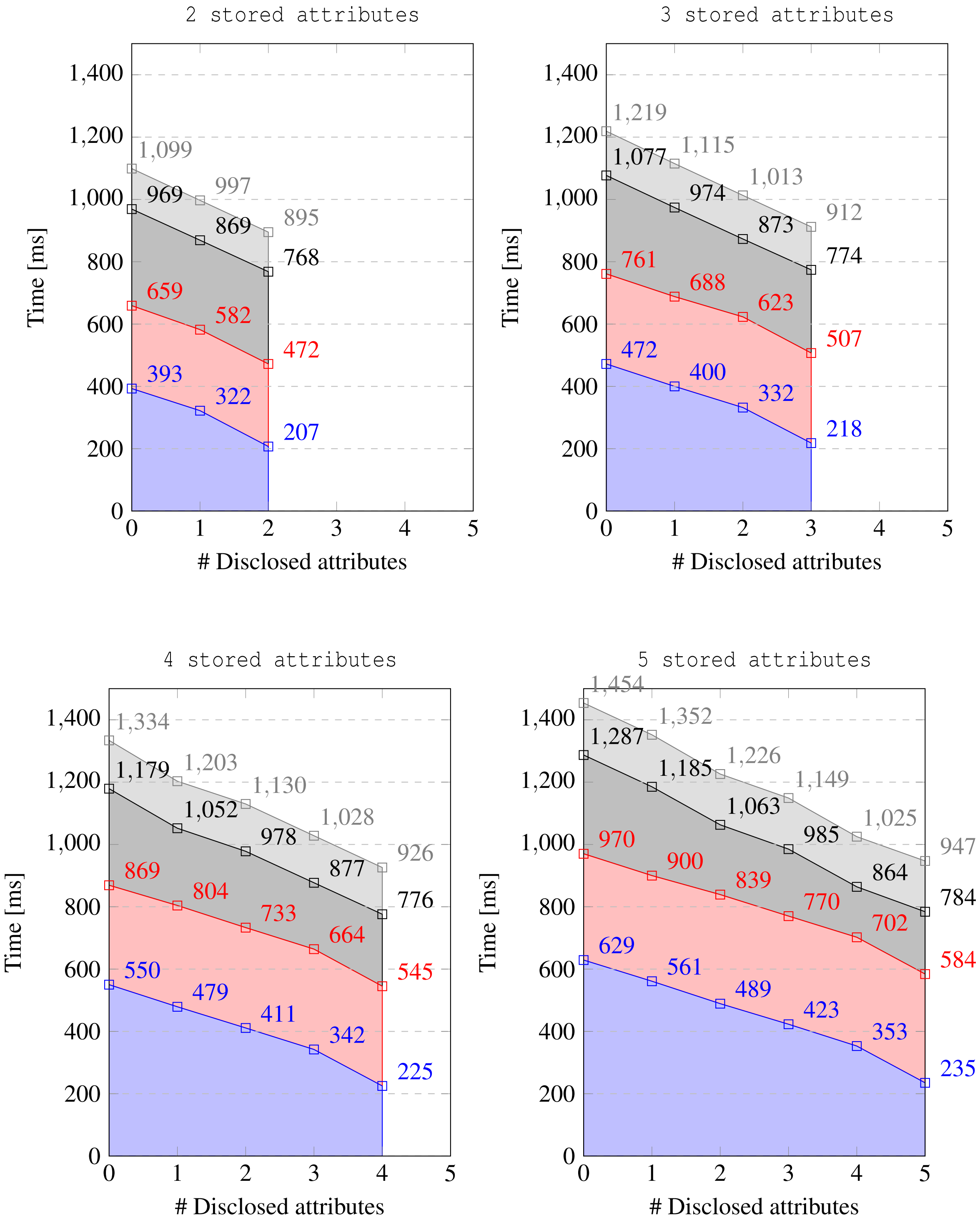


Figure 23: Speed of our proving protocol compared to Vullers and Alpár (VA) implementation [70]. Blue - our algorithm time, red - our overhead, black - VA algorithm time and grey - VA overhead.

GHz, 8 GB RAM) makes the implementation suitable also for time-critical applications like public transportation and ticketing. We also evaluated our scheme using an embedded device (Raspberry Pi 3) instead of the PC as a terminal. Even in that case the total time including overhead was below 450 ms. Based on our benchmarks, we expect that increasing security parameters to the 256-bit EC level would cost acceptable 15 % - 20 % in performance. Our implementation is artificially limited to 10 attributes per a user, but the smart card's available memory resources (approx. 1.75 KB RAM and 7.5 KB usable EEPROM) would allow storing upto 50 attributes on a single card.

Furthermore, we also provide the comparison of our scheme with the state of the art ABC schemes, see Table 10. Our proving algorithm, the part of the protocol we envision being executed on a smart card, only requires $u+2$ exponentiations, where u is the number of undisclosed attributes. Idemix takes place in the RSA group, meaning that the exponentiations are much more expensive than exponentiations in a prime order group. U-Prove lacks the unlinkability property. Compared to $\mathsf{MAC_{BB}}$, our scheme requires only 2 exponentiations without hidden attributes, whereas $\mathsf{MAC_{BB}}$ requires 12, showing that especially for a small number of undisclosed attributes, our scheme is significantly faster than $\mathsf{MAC_{BB}}$.

Table 10: Comparison of presentation protocols of credential schemes.

	Exp. prime	**Exp. RSA**	**Unlinkability**	**MAC**	**Security**
U-Prove [58]	$u+1$	0	✗	✗	-
Idemix [33]	0	$u+3$	✓	✗	sRSA [65]
Ringers et al. [64]	$n+u+9$	0	✓	✗	whLRSW [71]
$\mathsf{MAC_{DDH}}$ [34]	$6u+12$	0	✓	✓	DDH [21]
$\mathsf{MAC_{GGM}}$ [34]	$5u+4$	0	✓	✓	GGM [67]
$\mathsf{MAC_{BB}}$ [16]	$u+12$	0	✓	✓	q-sDH [22]
Our work	$u+2$	0	✓	✓	sDDHI, SDH [29, 26]

7 CONCLUSION

The main goal of this doctoral thesis was to find novel privacy-preserving cryptographic solutions for current ICT application scenarios, especially for access control and data collection systems. The main emphasis was put on the support of new privacy-preserving features, such as anonymity, untraceability and unlinkability. Furthermore, the revocation and identification must remain possible and the developed schemes must be practical in wide applications, i.e. the implementation mus be efficient even on constrained devices, such as smart cards. Following these requirements, the thesis presents four novel lightweight privacy-preserving cryptographic proposals, that are provable secure and practical in many current ICT application scenarios.

The first proposed scheme, presented in Section 3, is provably secure and provides the full set of privacy-enhancing features, that is anonymity, untraceability and unlinkability of users. Furthermore, our scheme supports distributed multi-device authentication with multiple RFID user devices. This feature is particularly important in applications for controlling an access to dangerous areas where the presence of protective equipment is checked during each access control session. Besides the full cryptographic specification, we also show the results of our implementation on devices commonly used in access control applications, i.e. smart cards and embedded verification terminals. By avoiding costly operations on user devices, such as bilinear pairings, we were able to achieve times comparable with existing systems (around 500 ms), while providing significantly higher security, privacy protection and features for RFID multi-device authentication.

In Section 4, we provide the full cryptographic specification of our novel scheme for secure privacy-friendly data collection that is designed for computationally restricted user devices and supports all the security, privacy-protection and inspection features. Using the scheme, data can be anonymously collected from almost all types of devices, including simple sensors and smart meters. On the other side, malicious users can be efficiently identified and revoked. Furthermore, we provide the practical results of our implementation of the scheme on embedded devices, smart phones, smart cards, smart watches, computers and servers so that the efficiency can be thoroughly evaluated on various platforms.

Section 5 presents our novel anonymous attribute-based credential scheme. We modify the original scheme of Hajny and Malina [43] in a way that the scheme becomes more efficient due the use of elliptic curve construction. The scheme provides anonymity, untraceability, unlinkability, selective disclosure of attributes, non-transferability, revocation and malicious user identification as the original scheme. However, by involving elliptic curves, we achieved faster verification phase (by 30%) and smaller communication cost between the user and the verifier (by 85%) compared to the original scheme, with equivalent or greater security level.

The last proposed scheme is presented in Section 6. The section introduces our novel keyed-verification credential system designed for lightweight devices (primarily smart cards) and provides security and efficiency proofs. By using a novel algebraic MAC based on Boneh-Boyen signatures, we achieve the most efficient proving protocol compared to existing schemes. In order to demonstrate the practicality of our scheme, we present an implementation on a standard, off-the-shelf, MultOS smart card. While using significantly higher security parameters than most existing implementations, we achieve performance that is more than 44 % better than the current state of the art implementations.

BIBLIOGRAPHY

Author's Selected Publications

[1] Camenish, Jan, Drijvers, Manu, Dzurenda, Petr, and Hajny, Jan. *Fast Keyed-Verification Anonymous Credentials on Standard Smart Cards*. In 34th International Conference on ICT Systems Security and Privacy Protection - IFIP SEC 2019, Lecture Notes in Computer Science (LNCS). Springer, 2019. Lisbon, Portugal, (Accepted).

[2] Dzurenda, Petr, Hajny, Jan, Malina, Lukas, and Ricci, Sara. *Anonymous Credentials with Practical Revocation using Elliptic Curves*. In SECRYPT 2017 Proceedings, pp. 534–539. 2017. ISBN: 978-989-758-259- 2.

[3] Dzurenda, Petr, Ricci, Sara, Hajny, Jan, and Malina, Lukas. *Performance Analysis and Comparison of Different Elliptic Curves on Smart Cards*. In International Conference on Privacy, Security and Trust (PST), pp. 1–10. 2017. Calgary, Canada, ISBN: 978-1-5386-2487-6.

[4] Hajny, Jan, Dzurenda, Petr, and Malina, Lukas. *Privacy-PAC: Privacy-Enhanced Physical Access Control*. In Proceedings of the ACM CCS, WPES '14, pp. 93–96. New York, NY, USA: ACM, 2014. ISBN 978-1-4503-3148-7. doi:10.1145/2665943. 2665969.

URL `http://doi.acm.org/10.1145/2665943.2665969`

[5] Hajny, Jan, Dzurenda, Petr, and Malina, Lukas. *Attribute-based credentials with cryptographic collusion prevention*. Security and Communication Networks, 8(18):3836–3846, 2015.

[6] Hajny, Jan, Dzurenda, Petr, and Malina, Lukas. *Multidevice Authentication with Strong Privacy Protection*. Wireless Communications and Mobile Computing, pp. 1–12, 2018. Vol. 2018, no. 3295148, ISBN: 1530-8669.

[7] Hajny, Jan, Dzurenda, Petr, Malina, Lukas, and Ricci, Sara. *Anonymous Data Collection Scheme from Short Group Signatures*. In SECRYPT 2018 Proceedings, pp. 1–10. 2018. ISBN: 978-989-758-319-3.

[8] Malina, Lukas, Dzurenda, Petr, and Hajny, Jan. *Evaluation of anonymous digital signatures for privacy-enhancing mobile applications*. International Journal of Security and Networks (online), 13(1):27–41, 2018.

[9] Malina, Lukas, Dzurenda, Petr, Hajny, Jan, and Martinasek, Zdenek. *Assessment of Cryptography Support and Security on Programmable Smart Cards*. In 2018 41st International Conference on Telecommunications and Signal Processing (TSP), pp. 1–5. IEEE, 2018.

[10] Malina, Lukas, Hajny, Jan, Dzurenda, Petr, and Zeman, Vaclav. *Privacy-preserving security solution for cloud services*. Journal of applied research and technology, 13(1):20–31, 2015.

Other Publications

[11] 186-4, FIPS PUB. *Federal Information Processing Standards Publication: Digital Signature Standard (DSS)*. 2013.
URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.362.5590`

[12] 20008-2:2013, ISO/IEC. *Information technology - security techniques - anonymous digital signatures - part 2: Mechanisms using a group public key*. 2013. International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC).

[13] ALPÁR, Gergely, HOEPMAN, Jaap H., and LUEKS, Wouter. *An attack against fixed value discrete logarithm representations*. Cryptology ePrint Archive, Report 2013/120, 2013. `http://eprint.iacr.org/`.

[14] ARFAOUI, Ghada, LALANDE, Jean-Franccois, TRAORÉ, Jacques, DESMOULINS, Nicolas, BERTHOMÉ, Pascal, and GHAROUT, Saïd. *A practical set-membership proof for privacy-preserving NFC mobile ticketing*. Proceedings on Privacy Enhancing Technologies, 2015(2):25–45, 2015.

[15] ATENIESE, Giuseppe, CAMENISCH, Jan, JOYE, Marc, and TSUDIK, Gene. *A practical and provably secure coalition-resistant group signature scheme*. In Annual International Cryptology Conference, pp. 255–270. Springer, 2000.

[16] BARKI, Amira, BRUNET, Sollen, DESMOULINS, Nicolas, and TRAORÉ, Jacques. *Improved Algebraic MACs and Practical Keyed-Verification Anonymous Credentials*. In Proceedings of the 2016 Selected Areas in Cryptography - SAC 2016. 2016.

[17] BELENKIY, Mira, CAMENISCH, Jan, CHASE, Melissa, KOHLWEISS, Markulf, LYSYANSKAYA, Anna, and SHACHAM, Hovav. Randomizable Proofs and Delegatable Anonymous Credentials, pp. 108–125. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. ISBN 978-3-642-03356-8. doi:10.1007/978-3-642-03356-8_7.
URL `https://doi.org/10.1007/978-3-642-03356-8_7`

[18] BERNARD, Marr. *How Is Big Data Used In Practice? 10 Use Cases Everyone Must Read*, 2018.

[19] BICHSEL, Patrik, BINDING, Carl, CAMENISCH, Jan, GROSS, Thomas, HEYDT-BENJAMIN, Tom, SOMMER, Dieter, and ZAVERUCHA, Greg. *Specification of the Identity Mixer Cryptographic Library version 2.3.0**. Tech. rep., IBM, 2010.

[20] BICHSEL, Patrik, CAMENISCH, Jan, GROSS, Thomas, and SHOUP, Victor. *Anonymous credentials on a standard java card*. In Proceedings of the 16th ACM conference on Computer and communications security, CCS '09, pp. 600–610. New York, NY, USA: ACM, 2009. ISBN 978-1-60558-894-0.

[21] BONEH, Dan. *The Decision Diffie-Hellman problem*. In Algorithmic Number Theory, pp. 48–63. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998. ISBN 978-3-540-69113-6.

[22] BONEH, Dan and BOYEN, Xavier. *Short Signatures Without Random Oracles*. In Advances in Cryptology - EUROCRYPT 2004, pp. 56–73. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. ISBN 978-3-540-24676-3.

[23] BONEH, Dan and BOYEN, Xavier. *Short Signatures Without Random Oracles and the SDH Assumption in Bilinear Groups*. Journal of Cryptology, 21(2):149–177, 2008. ISSN 1432-1378. doi:10.1007/s00145-007-9005-7.
URL `http://dx.doi.org/10.1007/s00145-007-9005-7`

[24] BONEH, Dan, BOYEN, Xavier, and SHACHAM, Hovav. *Short group signatures*. In Advances in Cryptology - CRYPTO'04. 2004. ISBN 3-540-22668-0.

[25] BRICKELL, Ernie, CAMENISCH, Jan, and CHEN, Liqun. *Direct Anonymous Attestation*. In Proceedings of the 11th ACM Conference on Computer and Communications Security, CCS '04, pp. 132–145. New York, NY, USA: ACM, 2004. ISBN 1-58113-961-6. doi:10.1145/1030083.1030103.
URL `http://doi.acm.org/10.1145/1030083.1030103`

[26] Brown, Daniel R. L. and Gallant, Robert P. *The Static Diffie-Hellman Problem*. IACR Cryptology ePrint Archive, 2004:306, 2004.
URL http://eprint.iacr.org/2004/306

[27] Camenisch, Jan, Drijvers, Manu, and Hajny, Jan. *Scalable Revocation Scheme for Anonymous Credentials Based on N-times Unlinkable Proofs*. In Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society, WPES '16, pp. 123–133. New York, NY, USA: ACM, 2016. ISBN 978-1-4503-4569-9. doi:10.1145/2994620.2994625.
URL http://doi.acm.org/10.1145/2994620.2994625

[28] Camenisch, Jan and Groth, Jens. *Group signatures: Better efficiency and new theoretical aspects*. In International Conference on Security in Communication Networks, pp. 120–133. Springer, 2004.

[29] Camenisch, Jan, Hohenberger, Susan, Kohlweiss, Markulf, Lysyanskaya, Anna, and Meyerovich, Mira. *How to win the clonewars: efficient periodic n-times anonymous authentication*. In Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS 2006, Alexandria, VA, USA, Ioctober 30 - November 3, 2006, pp. 201–210. 2006. doi:10.1145/1180405.1180431.
URL http://doi.acm.org/10.1145/1180405.1180431

[30] Camenisch, Jan and Lysyanskaya, Anna. An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation, pp. 93–118. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001. ISBN 978-3-540-44987-4. doi:10.1007/3-540-44987-6_7.
URL http://dx.doi.org/10.1007/3-540-44987-6_7

[31] Camenisch, Jan and Lysyanskaya, Anna. *A signature scheme with efficient protocols*. In Proceedings of the 3rd international conference on Security in communication networks, SCN'02, pp. 268–289. Berlin, Heidelberg: Springer-Verlag, 2003. ISBN 3-540-00420-3.

[32] Camenisch, Jan and Stadler, Markus. Efficient group signature schemes for large groups, pp. 410–424. Springer Berlin Heidelberg, 1997. ISBN 978-3-540-69528-8. doi:10.1007/BFb0052252.
URL http://dx.doi.org/10.1007/BFb0052252

[33] Camenisch, Jan and Van Herreweghen, Els. *Design and implementation of the idemix anonymous credential system*. In Proceedings of the 9th ACM conference on Computer and communications security, CCS '02, pp. 21–30. New York, NY, USA: ACM, 2002. ISBN 1-58113-612-9.

[34] Chase, Melissa, Meiklejohn, Sarah, and Zaverucha, Greg. *Algebraic MACs and Keyed-Verification Anonymous Credentials*. In Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, CCS '14, pp. 1205–1216. New York, NY, USA: ACM, 2014. ISBN 978-1-4503-2957-6. doi:10.1145/2660267.2660328.
URL http://doi.acm.org/10.1145/2660267.2660328

[35] Chaum, David and Van Heyst, Eugène. *Group signatures*. In Workshop on the Theory and Application of of Cryptographic Techniques, EUROCRYPT'91, pp. 257–265. Berlin, Heidelberg: Springer-Verlag, 1991. ISBN 3-540-54620-0.

[36] De Caro, Angelo and Iovino, Vincenzo. *jPBC: Java pairing based cryptography*. In Proceedings of the 16th IEEE Symposium on Computers and Communications, ISCC 2011, pp. 850–855. Kerkyra, Corfu, Greece, June 28 - July 1: IEEE, 2011. http://gas.dia.unisa.it/projects/jpbc/.
URL \url{http://gas.dia.unisa.it/projects/jpbc/}

[37] de la Piedra, Antonio, Hoepman, Jaap-Henk, and Vullers, Pim. Towards a Full-Featured Implementation of Attribute Based Credentials on Smart Cards, pp. 270–289. Cham: Springer International Publishing, 2014. ISBN 978-3-319-12280-9. doi:10.1007/978-3-319-12280-9_18.
URL http://dx.doi.org/10.1007/978-3-319-12280-9_18

[38] DELERABLÉE, Cécile and POINTCHEVAL, David. *Dynamic fully anonymous short group signatures*. In Progress in Cryptology-VIETCRYPT 2006, pp. 193–210. Springer, 2006.

[39] *COMMUNICATION FROM THE COMMISSION TO THE EUROPEAN PARLIAMENT, THE COUNCIL, THE EUROPEAN ECONOMIC AND SOCIAL COMMITTEE AND THE COMMITTEE OF THE REGIONS A European strategy on Cooperative Intelligent Transport Systems, a milestone towards cooperative, connected and automated mobility*. Official Journal of the European Union, pp. 1–12, 2016.

[40] *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*. Official Journal of the European Union, L119:1–88, 2016.
URL `http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=OJ:L:2016:119:TOC`

[41] FIAT, Amos and SHAMIR, Adi. *How To Prove Yourself: Practical Solutions to Identification and Signature Problems*. In Andrew M. Odlyzko, editor, Advances in Cryptology — CRYPTO' 86, pp. 186–194. Berlin, Heidelberg: Springer Berlin Heidelberg, 1987. ISBN 978-3-540-47721-1.

[42] HAJNĂ˝, Jan. Authentication Protocols and Privacy Protection. Doctoral thesis, Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Telecommunications, Brno, 2012.

[43] HAJNY, Jan and MALINA, Lukas. *Unlinkable Attribute-Based Credentials with Practical Revocation on Smart-Cards*. In Stefan Mangard, editor, Smart Card Research and Advanced Applications - CARDIS 2012, Lecture Notes in Computer Science, pp. 62–76. Springer Berlin Heidelberg, 2013. ISBN 978-3-642-37287-2.

[44] HAJNY, Jan, MALINA, Lukas, MARTINASEK, Zdenek, and TETHAL, Ondrej. Performance Evaluation of Primitives for Privacy-Enhancing Cryptography on Current Smart-Cards and Smart-Phones, pp. 17–33. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014. ISBN 978-3-642-54568-9. doi:10.1007/978-3-642-54568-9_2.
URL `https://doi.org/10.1007/978-3-642-54568-9_2`

[45] HAJNY, Jan, MALINA, Lukas, MARTINASEK, Zdenek, and ZEMAN, Vaclav. *Privacy-preserving SVANETs: Privacy-preserving simple vehicular ad-hoc networks*. In Security and Cryptography (SECRYPT), 2013 International Conference on, pp. 1–8. IEEE, 2013.

[46] HAJNY, Jan, MALINA, Lukas, and TETHAL, Ondrej. *Privacy-Friendly Access Control Based on Personal Attributes*. In The 9th International Workshop on Security, vol. 8639 of *Lecture Notes in Computer Science*, pp. 1–16. Springer, 2014. ISBN -.

[47] HWANG, Jung Yeon, LEE, Sokjoon, CHUNG, Byung-Ho, CHO, Hyun Sook, and NYANG, DaeHun. *Short group signatures with controllable linkability*. In Lightweight Security & Privacy: Devices, Protocols and Applications (LightSec), 2011 Workshop on, pp. 44–52. IEEE, 2011.

[48] ISERN-DEYÀ, Andreu Pere, HUGUET-ROTGER, Llorencc, PAYERAS-CAPELLÀ, Magdalena M., and MUT-PUIGSERVER, Macià. *On the practicability of using group signatures on mobile devices: implementation and performance analysis on the android platform*. International Journal of Information Security, 14(4):335–345, 2015.

[49] ISSHIKI, Toshiyuki, MORI, Kengo, SAKO, Kazue, TERANISHI, Isamu, and YONEZAWA, Shoko. *Using group signatures for identity management and its implementation*. In Proceedings of the second ACM workshop on Digital identity management, pp. 73–78. ACM, 2006.

[50] LIVINGSTONE, Sonia, Ă“LAFSSON, Kjartan, and STAKSRUD, Elisabeth. *Social networking, age and privacy*. The European Commission Safer Internet Programme, LSE Research Online, 2011. `http://eprints.lse.ac.uk/id/eprint/35849`.

[51] LYNN, Ben. *The pairing-based cryptography (PBC) library*. `https://crypto.stanford.edu/pbc/`, 2018.

[52] Manulis, Mark, Fleischhacker, Nils, Gunther, Felix, Kiefer, Franziskus, and Poettering, Bertram. *Group signatures: Authentication with privacy*. pp. 1–267, 2012.
URL `https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Studies/GruPA/GruPA.pdf`

[53] Mostowski, Wojciech and Vullers, Pim. *Efficient U-Prove implementation for anonymous credentials on smart cards*. In International Conference on Security and Privacy in Communication Systems, pp. 243–260. Berlin, Heidelberg: Springer, 2011.

[54] Naumann, Ingo and Hogben, Gilles. *ENISA: Privacy Features of eID Cards*. Network Security Newsletter, 2008:9–13, 2008. ISSN 1353-4858.

[55] of the interior of the Czech Republic, Ministry. *eObÄŤanka: ZaÄŤÃ˘tek digitalizace v ÄŚR*, 2018.
URL `https://portal.gov.cz/eobcanka`

[56] Okamoto, Tatsuaki and Uchiyama, Shigenori. *A new public-key cryptosystem as secure as factoring*. In International Conference on the Theory and Applications of Cryptographic Techniques, pp. 308–318. Springer, 1998.

[57] Paillier, Pascal. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes, pp. 223–238. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999. ISBN 978-3-540-48910-8. doi:10.1007/3-540-48910-X_16.
URL `https://doi.org/10.1007/3-540-48910-X_16`

[58] Paquin, Christian and Zaverucha, Greg. *U-Prove Cryptographic Specification V1.1 (Revision 3)*. In Microsoft, pp. 1–23. 2013.

[59] Pointcheval, David and Sanders, Olivier. Short Randomizable Signatures, pp. 111–126. Cham: Springer International Publishing, 2016. ISBN 978-3-319-29485-8. doi:10.1007/978-3-319-29485-8_7.
URL `http://dx.doi.org/10.1007/978-3-319-29485-8_7`

[60] Pointcheval, David and Stern, Jacques. *Provably secure blind signature schemes*. In International Conference on the Theory and Application of Cryptology and Information Security, pp. 252–265. Springer, 1996.

[61] Poller, Andreas, Waldmann, Ulrich, Vowé, Sven, and Türpe, Sven. *Electronic identity cards for user authentication-promise and practice*. IEEE Security & Privacy, 10(1):46–54, 2012.

[62] Potzmader, Klaus, Winter, Johannes, Hein, Daniel, Hanser, Christian, Teufl, Peter, and Chen, Liqun. *Group signatures on mobile devices: Practical experiences*. In International Conference on Trust and Trustworthy Computing, pp. 47–64. Springer, 2013.

[63] Research, Juniper. *SMART GRIDS TO SAVE CITY DWELLERS $14BN IN ENERGY COSTS BY 2022*, 8th January 2018. Hampshire, UK.

[64] Ringers, Sietse, Verheul, Eric R., and Hoepman, Jaap-Henk. *An efficient self-blindable attribute-based credential scheme*. IACR Cryptology ePrint Archive, 2017:115, 2017.

[65] Rivest, Ronald L. and Kaliski, Burt. RSA Problem, pp. 532–536. Boston, MA: Springer US, 2005. ISBN 978-0-387-23483-0. doi:10.1007/0-387-23483-7_363.
URL `https://doi.org/10.1007/0-387-23483-7_363`

[66] Schnorr, Claus. *Efficient identification and signatures for smart cards*. In Advances in cryptology—CRYPTO'89 proceedings, pp. 239–252. Springer, 1990.

[67] Shoup, Victor. Lower Bounds for Discrete Logarithms and Related Problems, pp. 256–266. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997. ISBN 978-3-540-69053-5. doi:10.1007/3-540-69053-0_18.
URL `http://dx.doi.org/10.1007/3-540-69053-0_18`

[68] Smart, Nigel. *ECRYPT II Yearly Report on Algorithms and Keysizes*. Tech. rep., Katholieke Universiteit Leuven, 2012. ECRYPT II European Network of Excellence in Cryptology II, ICT-2007-216676.

[69] THE WHITE HOUSE. *National Strategy for Trusted Identities in Cyberspace, Enhancing Online Choice, Efficiency, Security, and Privacy*. Washington, April 14, 2011, Retrieved September 9, 2017. `https://www.nist.gov/sites/default/files/documents/2016/12/08/nsticstrategy.pdf`.

[70] VULLERS, Pim and ALPAR, Gergely. *Efficient Selective Disclosure on Smart Cards Using Idemix*. In Policies and Research in Identity Management, vol. 396 of *IFIP Advances in Information and Communication Technology*, pp. 53–67. Springer Berlin Heidelberg, 2013. ISBN 978-3-642-37281-0.

[71] WEI, Victor K. and YUEN, Tsz Hon. *More short signatures without random oracles*. Cryptology ePrint Archive, Report 2005/463, 2005. `https://eprint.iacr.org/2005/463`.

Ing. Petr **Dzurenda**

Researcher · Academic staff

Brno University of Technology,
Faculty of Electrical Engineering and Communication, Department of Telecommunications,
Technicka 12, 616 00, Brno,
Czech Republic

☎ (+420) 720 591 809 | ✉ dzurenda@feec.vutbr.cz | ⌂ https://www.vutbr.cz/lide/petr-dzurenda-106420

Professional experience

2014 - Present **Scientific Worker** *at Centre of Sensor, Information and Communication Systems (SIX).*

2014 - Present **Researcher** *at Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Telecommunications.*

2014 - Present **Academic staff** *at Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Telecommunications.*

Education

2013 - Present **Doctoral program (Ph.D.)** *at Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Telecommunications.*
Thesis title: Cryptographic protection of digital identity

2010 - 2013 **Master program (Ing.)** *at Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Telecommunications.*
Thesis title: The security risks of authentication methods

2007 - 2010 **Bachelor program (Bc.)** *at Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Telecommunications.*
Thesis title: Data transmition security with general linear block codes

Internship abroad

10/2016 - 01/2017 **Universitat Rovira i Virgili,** *Department of Computer Engineering and Mathematics, Tarragona, Spain. Research in: Elliptic Curve Cryptography, Performance of different elliptic curves on Smart Cards. Duration: four month*

09/2015 - 12/2015 **Universitat Rovira i Virgili,** *Department of Computer Engineering and Mathematics, Tarragona, Spain. Research in: Privacy Enhancing Technologies, Privacy in Low Emission Zones and Automatic Fare Collection systems. Duration: four month*

Publication Activities (Scopus, Web of Science)

IF journals	*5*
Technical journals	*7*
International conf.	*18*
Total publications	***30***
H-index (WoS)	***1***
H-index (Scopus)	***3***
H-index (Google Scholar)	***4***

Research projects

2018 - 2020 **Strategic Programs for Advanced Research and Technology in Europe - SPARTA (H2020-SU-ICT-03-2018):** *Horizon 2020: Research and Innovation action (RIA)*

2019 - 2021 **Legal and technical means of privacy protection in cyberspace (TL02000398):** *Technology Agency of the Czech Republic (TACR)*

2017 - 2022 **Creation of a double-degree doctoral study program Electronics and Information Technology and creation of a doctoral study program Information Security - OP3V (CZ.02.2.69/0.0/0.0/16_018/0002575):** *Ministry of Education, Youth and Sports (MSMT)*

2017 - 2020 **Automated management and monitoring of protective equipment (FV20354):** *Ministry of Industry and Trade (MPO)*

2015 - 2019 **Interdisciplinary Research of Wireless Technologies - INWITE (LO1401):** *Ministry of Education, Youth and Sports (MSMT)*

2016 - 2018 **Secure Access-Control for Critical Infrastructures (VI20162018003):** *Ministry of the interior of the Czech Republic (MVCR)*

2014 - 2017 **Secure Systems for Electronic Services User Verification (TA04010476):** *Technology Agency of the Czech Republic (TACR)*

2014 - 2016 **Research into cryptographic primitives for secure authentication and digital identity protection (GP14-25298P):** *Czech Science Foundation (GACR)*

Expert Reviewer

IF Journal **IEEE Access** *(3.557 Impact Factor), ISSN: 1941-0026*

Certificates

2018 **Post-quantum Cryptography Course:** *Basque center for applied mathematics - bcam, Bilbao, Spain*

2017 **Palo Alto Networks ACE:** *Accredited Configuration Engineer (ACE) Exam*

2017 **Hillstone Networks:** *Hillstone Certified Security Professional*

2013 **Cisco Networking Academy:** *CCNA Exploration: Network Fundamentals*

2013 **Cisco Networking Academy:** *CCNA Exploration: Routing Protocols and Concepts*

Teaching activities

2019 - Present **ICT Security 1** *- TIC1, Assistant Lecturer, (Laboratory exercise and Lectures, BUT)*

2017 - Present **ICT Security 1** *- TIC1, Assistant Lecturer, (Laboratory exercise, BUT)*

2016 **Applied Cryptography** *- TAKR, Assistant Lecturer, (Laboratory exercise, BUT)*

2014 - 2016 **Cryptography** *- MKRI, Assistant Lecturer, (Laboratory exercise, BUT)*

2014 **Advanced Data Transmission Technology** *- MVDP, Assistant Lecturer, (Laboratory exercise, BUT)*

Skills

Professional skills *Cryptography, Privacy-Enhancing Technologies, IoT Security, Smart Cards, Elliptic Curves, Post-Quantum Cryptography, Cyber-Security*

Programming *C/C++ (Intermediate), C# (Basic), Java (Advanced), MultOS Card (Advanced), Java Card (Advanced), Basic Card (Advanced), Android (Intermediate)*

Languages *Czech (Native), English (B2), Spanish (A2), Italian (A1)*

Selected publications

IF journals

[1] HAJNÝ, J.; DZURENDA, P.; MALINA, L. Multidevice Authentication with Strong Privacy Protection. WIRELESS COMMUNICATIONS & MOBILE COMPUTING, 2018, vol. 2018, no. 3295148, p. 1-12. ISSN: 1530-8669.

[2] MALINA, L.; DZURENDA, P.; HAJNÝ, J.; MARTINÁSEK, Z. Secure and Efficient Two-factor Zero-knowledge Authentication Solution for Access Control Systems. COMPUTERS & SECURITY, 2018, vol. 77, no. 2018, p. 500-513. ISSN: 0167-4048.

[3] ČLUPEK, V.; ZEMAN, V.; DZURENDA, P. Light-weight Mutual Authentication with Non-repudiation. Radioengineering, 2018, vol. 27, no. 1, p. 143-150. ISSN: 1210-2512.

[4] FUJDIAK, R.; DZURENDA, P.; MLÝNEK, P.; MIŠUREC, J.; ORGOŇ, M.; BEZZATEEV, S. Anomalous Behaviour of Cryptographic Elliptic Curves over Finite Field. Elektronika Ir Elektrotechnika, 2017, vol. 23, no. 5, p. 82-88. ISSN: 1392-1215.

[5] HAJNÝ, J.; DZURENDA, P.; MALINA, L. Attribute- based credentials with cryptographic collusion prevention. Security and Communication Networks, 2015, vol. 8, no. 18, p. 3836-3846. ISSN: 1939-0114.

International conferences

[1] HAJNÝ, J.; DZURENDA, P.; MALINA, L.; RICCI, S. Anonymous Data Collection Scheme from Short Group Signatures. In SECRYPT 2018 Proceedings. 2018. p. 1-10. ISBN: 978-989-758-319-3.

[2] MALINA, L.; DZURENDA, P.; HAJNÝ, J. Evaluation of anonymous digital signatures for privacy-enhancing mobile applications. International Journal of Security and Networks (online), 2018, vol. 13, no. 1, p. 27-41. ISSN: 1747-8405.

[3] DZURENDA, P.; HAJNÝ, J.; MALINA, L.; RICCI, S. Anonymous Credentials with Practical Revocation using Elliptic Curves. In SECRYPT 2017 Proceedings. SCITEPRESS, 2017. p. 534-539. ISBN: 978-989-758-259-2.

[4] DZURENDA, P.; RICCI SARA; HAJNÝ, J.; MALINA, L. Performance Analysis and Comparison of Different Elliptic Curves on Smart Cards. In In 2017 the 15th International Conference on Privacy, Security and Trust (PST). 2017. p. 1-10. ISBN: 978-1-5386-2487-6.

[5] HAJNÝ, J.; DZURENDA, P.; MALINA, L. Privacy-Enhanced Data Collection Scheme for Smart- Metering. In Proceedings of the International Conference on Information Security and Cryptology. Lecture Notes in Computer Science. 2015. p. 1-18. ISSN: 0302-9743.

[6] HAJNÝ, J.; MALINA, L.; DZURENDA, P. Privacy-PAC: Privacy- Enhanced Physical Access Control. In WPES 2014 Proceedings. USA: 2014. p. 1-4. ISBN: 978-1-4503-3148-7.

Reference

doc. Ing. Jan Hajný, Ph.D.

Head of the Advanced Cybersecurity Group *at Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Telecommunications in Czech Republic.*
+420 54114 6961, hajny@feec.vutbr.cz

Dr. Jordi Castella-Roca

Head of the Department *at Universitat Rovira i Virgili, Department of Computer Engineering and Mathematics in Spain (Catalonia).*
+34 977 558270, jordi.castella@urv.cat

ABSTRACT

The doctoral thesis deals with privacy-preserving cryptographic schemes in access control and data collection areas. Currently, card-based physical access control systems are used by most people on a daily basis, for example, at work, in public transportation and at hotels. However, these systems have often very poor cryptographic protection. For instance, user identifiers and keys can be easily eavesdropped and counterfeited. Furthermore, privacy-preserving features are almost missing and, therefore, user's movement and behavior can by easily tracked. Service providers (and even eavesdroppers) can profile users, know what they do, where they go, and what they are interested in. In order to improve this state, we propose four novel cryptographic schemes based on efficient zero-knowledge proofs and elliptic curve cryptography. In particular, the thesis presents three novel privacy-friendly authentication schemes for access control and one for data collection application scenarios. The first scheme supports distributed multi-device authentication with multiple Radio-Frequency IDentification (RFID) user's devices. This feature is particularly important in applications for controlling access to dangerous areas where the presence of protective equipment is checked during each access control session. The other two presented schemes use attribute-based approach to protect user's privacy, i.e. these schemes allow users to anonymously prove the ownership of their attributes, such as age, citizenship, and gender. While one of our scheme brings efficient revocation and identification mechanisms, the other one provides the fastest authentication phase among the current state of the art solutions. The last (fourth) proposed scheme is a novel short group signature scheme for data collection scenarios. Data collection schemes are used for secure and reliable data transfer from multiple remote nodes to a central unit. With the increasing importance of smart meters in energy distribution, smart house installations and various sensor networks, the need for secure data collection schemes becomes very urgent. Such schemes must provide standard security features, such as confidentiality and authenticity of transferred data, as well as novel features, such as strong protection of user's privacy and identification of malicious users. The proposed schemes are provably secure and provide the full set of privacy-enhancing features, namely anonymity, untraceability and unlinkability of users. Besides the full cryptographic specification and security analysis, we also show the results of our implementations on devices commonly used in access control and data collection applications.