BRNO UNIVERSITY OF TECHNOLOGY

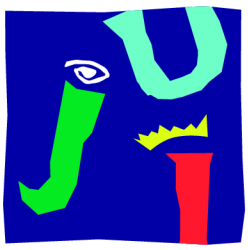
Faculty of Electrical Engineering
and Communication

UNIVERSITAT JAUME I

Institute of New Imaging
Technologies

DOCTORAL THESIS

Brno, September 2023 RAÚL CASANOVA-MARQUÉS



BRNO UNIVERSITY OF TECHNOLOGY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF TELECOMMUNICATIONS

UNIVERSITAT JAUME I

INSTITUTE OF NEW IMAGING TECHNOLOGIES

PRIVACY-ENHANCING TECHNOLOGIES AND PRIVACY-ENHANCING CRYPTOGRAPHY FOR WEARABLES

DOCTORAL THESIS

AUTHOR Raúl Casanova-Marqués

ADVISORS doc. Ing. Jan Hajný, Ph.D.
Dr. Michael Gould Carlson
Dr. Joaquín Torres-Sospedra

BRNO 2023

PRIVACY-ENHANCING TECHNOLOGIES AND PRIVACY-ENHANCING CRYPTOGRAPHY FOR WEARABLES

This thesis has been completed in a joint doctoral degree program at
Brno University of Technology and Universitat Jaume I

DPAD-EIT Electronics and Information Technologies (Double-Degree)

BRNO UNIVERSITY OF TECHNOLOGY

European Joint Doctorate Marie Skłodowska Curie in A Network for Dynamic Wearable Applications with Privacy Constraints (A-WEAR)

UNIVERSITAT JAUME I

.....
Raúl Casanova-Marqués

.....
doc. Ing. Jan Hajný, Ph.D.

.....
Dr. Michael Gould Carlson

.....
Dr. Joaquín Torres-Sospedra

This dissertation is funded by the European Union's Horizon 2020 Research and Innovation programme under the Marie Skłodowska Curie grant agreement No. 813278 (A-WEAR: A network for dynamic WEarable Applications with pRivacy constraints, https://www.a-wear.eu/).

ABSTRACT

The increasing concern surrounding privacy and the safeguarding of digital identities has emphasized the pressing necessity of establishing secure and confidential communication channels. This concern has led to the development of cryptographic mechanisms aimed at facilitating impervious information exchange. Nevertheless, traditional cryptographic approaches are proving insufficient in dynamic and resource-constrained environments, such as wearable devices. As a result, attribute-based credential schemes have emerged as a promising solution, offering fine-grained access control to digital services based on user-specific attributes.

This doctoral thesis examines the efficacy and scalability of attribute-based anonymous credential schemes in ensuring the authenticity and security of users within dynamic architectures of wearable devices. It also explores enhancements to these schemes, with a primary focus on incorporating user revocation while maintaining privacy. Additionally, the thesis presents devised mechanisms to enable attribute-based authentication protocols on smart cards with limited support for elliptic curve cryptography. It addresses specific challenges associated with the usability of smart cards. Moreover, the thesis investigates the integration of anonymous authentication schemes in collaborative indoor positioning systems, aiming to provide privacy and security. Lastly, it explores the implementation of attribute-based authentication schemes in resource-constrained environments, with an emphasis on Internet of Things devices, and evaluates their feasibility within the dynamic architectures of wearable devices.

The first contribution of this thesis introduces a purposefully designed protocol for anonymous authentication on smart cards. This protocol combines attribute-based credentials and user revocation while ensuring computational efficiency. To facilitate effective implementation and evaluation, the thesis employs smart cards equipped with the MULTOS operating system. The second contribution focuses on optimizing the capabilities of smart cards using Java Card technology for the implementation of attribute-based credential schemes. These smart cards are presented as a more accessible alternative for a wider consumer base. To overcome limitations in their application programming interface, the thesis devises strategies to augment the constrained support for elliptic curve cryptography and effectively implement such schemes. The third contribution presents the Privacy-Enhancing Authentication System, a robust solution compatible with smart cards, smartphones, and smartwatches. This system addresses the functional challenges associated with smart cards, including the absence of a graphical interface and limited user control over attribute disclosure. Consequently, it offers a practical and deployable solution for real-world scenarios. Finally, the thesis proposes a groundbreaking scheme to safeguard collaborative indoor positioning systems by addressing both privacy and security concerns. This scheme ensures the preservation of privacy and security by eliminating centralized architectures and employing encryption techniques for positioning information. The thesis includes comprehensive details such as protocol use cases, implementation specifics, execution benchmarks, and a comparative analysis with existing protocols.

KEYWORDS

Cryptographic protocols, attribute-based authentication, attribute-based credentials, privacy protection, anonymity, user revocation, smart cards, wearable architectures, Internet of Things, collaborative indoor positioning systems, elliptic curve cryptography.

RESUMEN

La creciente preocupación por la privacidad y la protección de la identidad digital han subrayado la necesidad crítica de establecer comunicaciones seguras y privadas. Esta preocupación ha impulsado el desarrollo de mecanismos criptográficos que garanticen el intercambio seguro de información. Sin embargo, la criptografía tradicional tiende a ser insuficiente en entornos dinámicos y con recursos limitados, como ocurre en los dispositivos wearables. Como resultado de esta necesidad, los esquemas de credenciales basados en atributos han surgido como una solución prometedora, ya que ofrecen un control de acceso granular a servicios digitales en función de las características del usuario.

Esta tesis doctoral analiza la eficacia y escalabilidad de los esquemas de credenciales anónimas basadas en atributos para garantizar la autenticidad y seguridad de los usuarios en arquitecturas dinámicas de dispositivos wearables. Además, se exploran mejoras en estos esquemas para incluir la revocación del usuario preservando la privacidad. También se diseñan mecanismos para habilitar protocolos de autenticación basados en atributos en tarjetas inteligentes con limitaciones de soporte para la criptografía de curva elíptica. Del mismo modo, se abordan los desafíos específicos de usabilidad asociados con las tarjetas inteligentes. Por otra parte, se investiga la integración de esquemas de autenticación anónima en sistemas colaborativos de posicionamiento en interiores, con el objetivo de proporcionar privacidad y seguridad. Por último, esta tesis explora la implementación de esquemas de autenticación basados en atributos en entornos con recursos limitados, en particular dispositivos de Internet de las cosas, y evalúa su viabilidad en arquitecturas dinámicas de dispositivos wearables.

La primera contribución de esta tesis introduce un protocolo de autenticación anónima específicamente diseñado para las tarjetas inteligentes. Este protocolo combina credenciales basadas en atributos y revocación de usuarios, garantizando al mismo tiempo la eficiencia computacional. Para su implementación y evaluación, se utilizan tarjetas con el sistema operativo MULTOS. La segunda contribución se enfoca en la optimización de las capacidades de las tarjetas que usan la tecnología Java Card para la implementación de esquemas de credenciales basados en atributos. Estas tarjetas inteligentes se presentan como una alternativa más accesible para el consumidor general. Así pues, y debido a las restricciones de su interfaz de programación de aplicaciones, se diseñan estrategias para ampliar el soporte limitado de la criptografía de curva elíptica e implementar eficientemente tales esquemas. La tercera contribución presenta el Privacy-Enhancing Authentication System, una solución robusta compatible con tarjetas, teléfonos y relojes inteligentes. Este sistema afronta los desafíos funcionales relacionados con las tarjetas, como la ausencia de una interfaz gráfica y la falta de control por parte del usuario sobre la divulgación de sus atributos. Como resultado, se ofrece una solución práctica y lista para su despliegue en entornos reales. Por último, se propone un esquema novedoso para proteger los sistemas colaborativos de posicionamiento en interiores al abordar tanto los problemas de privacidad como de seguridad. El esquema se asegura de proteger la privacidad y la seguridad al evitar arquitecturas centralizadas y cifrar la información de posicionamiento. También se incluyen casos de uso del protocolo, detalles de la implementación, resultados de su ejecución y un análisis comparativo con otros protocolos existentes.

PALABRAS CLAVE

Protocolos criptográficos, autenticación basada en atributos, credenciales basadas en atributos, protección de la privacidad, anonimato, revocación de usuarios, tarjetas inteligentes, arquitecturas wearables, Internet de las cosas, sistemas colaborativos de posicionamiento en interiores, criptografía de curva elíptica.

Author's Declaration

Author:	Raúl Casanova-Marqués
Author's ID:	223411
Paper type:	Doctoral thesis
Academic year:	2022/23
Topic:	Privacy-enhancing technologies and privacy-enhancing cryptography for wearables

I declare that I have written this paper independently, under the guidance of the advisors, and using exclusively the technical references and other sources of information cited in the paper and listed in the comprehensive bibliography at the end of the paper.

As the author, I furthermore declare that, with respect to the creation of this paper, I have not infringed any copyright or violated anyone's personal and/or ownership rights. In this context, I am fully aware of the consequences of breaking Regulation § 11 of the Copyright Act No. 121/2000 Coll. of the Czech Republic, as amended, and of any breach of rights related to intellectual property or introduced within amendments to relevant Acts such as the Intellectual Property Act or the Criminal Code, Act No. 40/2009 Coll. of the Czech Republic, Section 2, Head VI, Part 4.

Brno .....

.....
author's signature*

*The author signs only in the printed version.

A mi querida familia, por estar siempre a mi lado en este camino.

ACKNOWLEDGEMENT

This research work has been undertaken within the framework of a joint degree program. The program involves the Department of Telecommunications, Faculty of Electrical Engineering and Communication at Brno University of Technology, Czech Republic, and the Institute of New Imaging Technologies at Universitat Jaume I, Spain. It has been an enriching experience, and I would like to express my gratitude to everyone who has contributed to this educational voyage.

I gratefully acknowledge funding from the European Union's Horizon 2020 Research and Innovation programme under the Marie Skłodowska Curie grant agreement No. 813278, A-WEAR. I extend my heartfelt appreciation to the A-WEAR project and all those involved. Their commitment and collaborative endeavors in advancing wearable technology have been invaluable in shaping my research and providing profound insights. I am deeply grateful for the exceptional opportunities and abundant resources offered by this project.

I would also like to extend my deepest gratitude to doc. Ing. Jan Hajný, Ph.D., my supervisor and a distinguished authority in the field of cryptography. His exceptional expertise and guidance have played a pivotal role in shaping my understanding of this complex discipline. To my supervisor, Dr. Michael Gould Carlson, for his guidance and assistance throughout my research journey. A special expression of gratitude is reserved for my supervisor, Dr. Joaquín Torres-Sospedra, whose support during the final phase of my research has been truly invaluable. His mentorship, expert knowledge, and continuous encouragement have been instrumental in shaping the outcomes of my study. I am sincerely grateful for the guidance and timely feedback provided by all my supervisors, as they have greatly contributed to the success of my research endeavor.

I wish to convey my deepest gratitude to my esteemed former professor and cherished friend, Dr. Jordi Castellà-Roca. His exceptional mentorship, enlightening teachings, and stimulating discussions have left an indelible mark on my academic and personal development. I am profoundly grateful for his invaluable intellectual contributions, which have continuously inspired and enriched my journey.

Finally, to my beloved family, I am eternally grateful for your unwavering support and unconditional love throughout my academic journey. Your encouragement and understanding have been the pillars of strength that have propelled me forward during these challenging times. Your belief in my abilities and constant presence in my life have been a constant source of inspiration. I am profoundly thankful for the sacrifices you have made and the countless ways you have stood by my side. This achievement would not have been possible without your love, guidance, and unyielding support.

Contents

List of Figures

List of Tables

1 Introduction

Recently, the proliferation of electronic systems and devices has led to an exponential increase in the amount of digital information being generated and exchanged. While this trend has opened up new opportunities for communication, commerce, and social interaction, it has also raised significant concerns about privacy and digital identity protection. Cryptography, the science of secure communication, has emerged as a crucial tool for addressing these concerns, offering techniques to encode and decode information in ways that can only be accessed by authorized parties.

However, traditional cryptographic methods are not always suitable for the complex and dynamic environments of modern electronic systems, particularly in the case of wearable devices. These devices, which are often resource-constrained and operate in uncontrolled environments, pose significant challenges for cryptographic protocols, such as the need to maintain user authenticity and prevent unauthorized access. To address these challenges, researchers are developing novel cryptographic technologies that provide attribute-based authentication, enabling more granular control over access to digital information based on user characteristics.

1.1 Background and motivation

In the context of protecting user identity, cryptographic algorithms play a crucial role in maintaining the security and privacy of sensitive information by providing essential properties such as confidentiality, integrity, and authenticity. However, in heterogeneous networks such as the *Internet of Things* (IoT), where devices have limited computational resources, it is challenging to implement standard cryptographic algorithms. Asymmetric ciphers such as *Rivest-Shamir-Adleman* (RSA), *Digital Signature Algorithm* (DSA), or *Diffie-Hellman* (DH) may not be supported by *Central Processing Units* (CPUs) and microcontrollers, and implementing them in software may be difficult due to a lack of computational power. Moreover, implementing privacy-preserving features on constrained devices is even more challenging. Fortunately, advanced cryptographic schemes such as *Attribute-based Credentials* (ABCs) [1, 2, 3] can be used to protect user privacy due to their efficient and lightweight design, making them suitable for resource-constrained devices.

Attribute-based Credentials are a cryptographic approach to authentication that preserves the privacy and security of individuals. Instead of traditional credentials, which often require the disclosure of a broad range of personal information, ABCs utilize attributes that describe specific characteristics, such as legal age or citizenship. These attributes are securely stored and can be selectively disclosed by the user

for a particular transaction, giving individuals complete control over their personal information.

Compared to traditional authentication systems, ABCs [4, 5, 6] offer significant advantages, particularly in scenarios where privacy is a critical concern. They can effectively tackle issues such as identity theft, data breaches, and online privacy violations. ABCs have already been implemented in various areas, such as finance and education, demonstrating their versatility and potential for adoption in diverse contexts. Nevertheless, their implementation on offline devices with limited computational resources has been a hard problem for a long time due to a lack of computational resources and unsupported fundamental operations, such as bilinear pairings and arithmetic operations on elliptic curves. In particular, the implementations of core protocols on smart cards were impractical until very recently, according to [7, 8, 3, 9]. Implementations with efficient large-scale revocation are still completely missing on smart cards and only available on online and computationally strong user devices.

The use of smart cards for authentication, access control, and other security-related applications is widespread across various domains, including banking, transportation, healthcare, and government. With the increasing reliance on smart cards for these critical purposes, it is imperative to establish cryptographic protocols that are fast, efficient, and privacy-friendly in protecting the identities of users. Such protocols must prevent unauthorized access to sensitive information stored on smart cards while ensuring that the user's identity remains secure and confidential. Therefore, the development of privacy-enhancing technologies with smart cards is a vital undertaking that plays a critical role in safeguarding the privacy and security of users in the digital age.

On the other hand, as the use of the IoT and industrial networks continues to grow and wearable devices become more powerful, it is becoming increasingly important to prioritize security and privacy protection in real-world applications. One such application is the *Collaborative Indoor Positioning System* (CIPS), which currently compromises the security and privacy of its users.

CIPSs are a powerful tool that uses sensors and devices for locating individuals and objects within indoor environments through inter-user communication. However, CIPSs face several security and privacy challenges, including potential issues related to user tracking, unauthorized access to sensitive location data, and data manipulation. There is also the risk of malicious attacks on the system, such as spoofing or jamming [10], which can compromise the accuracy and reliability of location data. Robust security measures must be implemented to protect data from collection to transmission and processing. Inaccurate or corrupted location data can have serious consequences, especially in safety-critical applications such as emergency response

or industrial settings. User identity protection is also essential in the context of inter-user communication to safeguard against unauthorized disclosure or manipulation. In this light, ABCs could also potentially be applied in the context of CIPSs to enhance privacy and security. The use of encryption and digital signatures can provide assurance of data integrity and prevent unauthorized modifications of location data.

The reliability and integrity of CIPSs hinge on establishing identity protection measures and preventing potential issues related to user tracking. It is imperative to develop cryptographic protocols that not only authenticate users but also provide confidentiality to these systems. Such measures will foster confidence and trust in the system, enabling it to serve as a valuable tool for enhanced multi-user collaboration. By addressing these critical security and privacy concerns, CIPSs can unlock their full potential while safeguarding sensitive data and user identities.

1.2 Research questions, objectives, and challenges

The advancement of wearable technology has unlocked novel opportunities for user authentication and access control in various applications. However, guaranteeing the privacy and security of user identities in such dynamic and resource-constrained environments poses considerable challenges. As a solution, attribute-based anonymous credential schemes have emerged, but their adequacy for ensuring user authenticity in wearable architectures is not well understood. Moreover, the performance of wearable devices while executing anonymous credential protocols remains obscure, and the most effective ways to revoke invalid users in such schemes demand further examination. To enhance the comprehension of attribute-based anonymous credential schemes and their application in wearable environments, this thesis endeavors to explore these research questions:

- *How can anonymous credential schemes be adapted to support user revocation while maintaining privacy?*
- *What strategies can be employed to enable attribute-based authentication protocols on smart cards with limited support for elliptic curve cryptography?*
- *What are the usability challenges associated with using anonymous credentials in various applications, and how can they be addressed?*
- *How can anonymous credential schemes be integrated into collaborative indoor positioning systems to enhance privacy and security?*
- *How can anonymous credential schemes be implemented in resource-constrained environments, such as* IoT devices?

- *Are attribute-based authentication schemes suitable for ensuring user authenticity in dynamic wearable architectures?*

This thesis aims to contribute valuable insights into the effectiveness and scalability of attribute-based anonymous credential schemes in ensuring user authenticity and security in dynamic wearable architectures. To achieve this objective, the thesis addresses the aforementioned research questions. Furthermore, the primary objectives of this study are to design and develop novel cryptographic algorithms that offer efficient and effective protection of user privacy and digital identity in electronic systems. To attain this goal, the study addresses several challenges, including inefficient revocation of invalid users, missing identification of malicious users, and low performance on constrained devices like wearables. Finally, the developed algorithms undergo testing and benchmarking on existing wearable hardware devices, including smart cards, smartwatches, and smartphones.

1.3 Methodology, scope, and limitations

To achieve the objectives outlined above, this research adopts a methodology that combines both theoretical and empirical approaches. Theoretical research is conducted to investigate the state-of-the-art in cryptographic techniques and related fields, such as wireless networking and wearable devices. This involves conducting a comprehensive literature review of relevant studies, papers, and reports. Empirical research is conducted to evaluate the proposed solutions in a real-world scenario. This involves the development of prototypes or proof-of-concepts that are tested on existing wearable hardware devices. The performance and security of the developed solutions are evaluated by conducting experiments and simulations and comparing the results with existing solutions. Finally, the research uses both quantitative and qualitative research methods to analyze and interpret the data collected from the experiments and simulations.

The scope of this research is to design and evaluate novel cryptographic technologies for protecting the privacy and digital identity of electronic users in dynamic wearable architectures. Specifically, the research focuses on developing solutions for attribute-based authentication in electronic systems, addressing the issues of inefficient revocation of invalid users, missing identification of malicious users, and low performance on constrained devices such as wearables. The research also involves testing and benchmarking the developed algorithms on existing wearable hardware devices, such as smart cards, smartwatches, and smartphones. While this research aims to make a significant contribution to the field of cybersecurity, it is important to acknowledge that it has certain limitations. For example, the research is con-

ducted within the context of a specific use case or scenario and may not cover all possible situations where attribute-based authentication and wearable devices are used. Additionally, the research does not address issues related to user adoption or user experience but focuses solely on the technical aspects of the proposed solutions. Finally, the research may be limited by the resources, tools, and expertise available to the researcher and may not be able to address all possible aspects of the proposed solutions.

1.4 Contribution and outline

The contributions and structure of this dissertation are described in detail below. An important aspect to highlight is that each chapter is accompanied by a comprehensive review of the state of the art relevant to its respective topic. This deliberate approach provides a deeper understanding and contextualization of the research presented, enriching the scholarly significance of each chapter's exploration.

Chapter 2 provides a concise overview of the cryptographic preliminaries that were utilized during the development of the thesis. This includes an introduction to the notation used throughout the design of the cryptographic protocols as well as a detailed description of the DH key exchange protocol, the *weak Boneh-Boyen* (wBB) signature algorithm, and the Sigma protocols. By presenting these fundamental concepts, the chapter aims to provide the reader with a solid understanding of the building blocks that form the basis of the cryptographic technologies proposed in the thesis.

Chapter 3 introduces the *Revocable Keyed-Verification Anonymous Credential* (RKVAC) protocol, which allows for user authentication using anonymous credentials and supports efficient revocation, even on smart cards. The chapter presents a complete cryptographic specification of a novel scheme that integrates efficient attribute-proving protocols with revocation protocols, along with a full implementation of the revocable attribute-based credentials scheme on off-the-shelf smart cards with MULTOS operating systems. Extensive evaluations of the protocol are provided, demonstrating the efficiency of the cryptographic design and implementation. The chapter concludes with a discussion of the efficacy of the proposed protocol and a summary of the major outcomes. This chapter is based on publication [11], and the contributions of the PhD candidate are outlined in the chapter's summary.

Chapter 4 focuses on the implementation of the *Keyed-Verification Anonymous Credential* (KVAC) and RKVAC protocols for smart cards based on the Java Card technology. The restricted *Application Programming Interface* (API) of the Java Card platform poses a challenge to developers who wish to implement non-standard

protocols. To address this issue, we present a comprehensive Java Card implementation of ABC schemes. Since software implementation of algebraic operations would significantly slow down the protocol, we propose a solution that involves hardware acceleration and the transformation of fundamental mathematical operations to use the Java Card API to the greatest extent possible. By exploiting the restricted cryptographic API of the Java Card platform, we demonstrate how to accelerate the execution of modular arithmetic and elliptic curve operations. Furthermore, we apply various optimization and acceleration techniques to further reduce execution times. This chapter is based on publication [12].

Chapter 5 assesses the potential of ABC schemes for real-world applications, evaluating their maturity and readiness for deployment. The chapter introduces the *Privacy-Enhancing Authentication System* (PEAS), which utilizes ABCs technology to provide a range of privacy-preserving features, including anonymity, unlinkability, and untraceability. The system is designed to be flexible and easily deployable on devices with varying computational power. We present the results of our experiments, including benchmarking and piloting, to demonstrate the effectiveness and efficiency of our approach. This chapter is based on publications [13] and [14], and the contributions of the PhD candidate are outlined in the chapter's summary.

Chapter 6 presents a novel decentralized privacy-preserving authentication mechanism for CIPS. The proposed scheme is based on *Bluetooth Low Energy* (BLE) technology and offers several key benefits, including anonymized location data sharing, decentralized authentication, and offline revocation. The chapter includes a security analysis that demonstrates the robustness and effectiveness of the proposed scheme. We evaluate our protocol in various environments with different restrictions, demonstrating its versatility and effectiveness in real-world scenarios. To simplify the implementation of our protocol on unsupported platforms, we define a standardized BLE advertising packet format. We also provide comprehensive comparative results of our protocol's execution on commonly used devices, showcasing its superior performance and efficiency. Finally, the chapter provides a thorough comparative analysis with other existing location-based schemes, providing insights into the benefits and drawbacks of each approach. This chapter is based on publications [15] and [16].

Chapter 7 serves as the conclusion of this research work, summarizing the main findings and conclusions of the thesis. The research questions are answered based on the results obtained in the previous chapters. We discuss the contributions of this work to the field of privacy-preserving authentication and positioning systems and outline potential future research directions. We also reflect on the limitations and challenges of our proposed solutions and suggest areas for improvement.

2 Cryptographic preliminaries

This chapter serves as a retrospective examination of the foundational components that underpin our cryptographic schemes. Specifically, we examine the Diffie-Hellman key exchange protocol [17], the weak Boneh-Boyen signature scheme [18], and the Sigma protocols [19], which together provide the essential building blocks for constructing robust and secure cryptographic schemes.

2.1 Notation

The symbol ":" means "such that" and $|x|$ is the bit-length of x. The symbol $\mathcal{H}$ denotes a secure hash function. We write $a \in_R A$ when a is sampled uniformly at random from A. Let $\texttt{GroupSetup}\ (1^\kappa)$ be an efficient algorithm that generates a group $\mathbb{G} = \langle g \rangle$ of prime order q, such that $|q| = \kappa$. Let $\mathbf{e}$ denote a bilinear map $\mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$.

2.2 Diffie-Hellman key exchange

The Diffie-Hellman [17] key exchange protocol is one of the oldest and most widely used cryptographic protocols. The protocol allows two parties who have never interacted before to agree on a shared secret through an unsecure channel and anonymously. Figure 2.1 illustrates the operation of the DH key exchange protocol.

Alice		*Bob*
	$params = (q, \mathbb{G}, g)$	
$a \in_R \mathbb{Z}_q$		$b \in_R \mathbb{Z}_q$
$A = g^a$		$B = g^b$
	$A \longrightarrow$	
	$\longleftarrow B$	
$K = B^a = g^{ba}$		$K = A^b = g^{ab}$

Fig. 2.1: Definition of the Diffie-Hellman key exchange protocol

The basic idea behind the protocol is that the two parties, Alice and Bob, each generate a public-private key pair and exchange their public keys over the unsecure channel. They then use their private keys and the other party's public key to derive a shared secret that can be used as a symmetric encryption key. The security of

the protocol relies on the difficulty of computing discrete logarithms in a finite field. The following is a brief description of the DH key exchange protocol:

1. Alice and Bob agree on a finite cyclic group $params = (q, \mathbb{G}, g)$.
2. Alice randomly selects her private value $a \in_R \mathbb{Z}_q$ and computes the public value $A = g^a$.
3. Bob randomly selects his private value $b \in_R \mathbb{Z}_q$ and computes the public value $B = g^b$.
4. Alice and Bob exchange their respective public values, A and B.
5. Alice computes the shared key $K = B^a$, i.e., $K = g^{ba}$.
6. Bob computes the shared key $K = A^b$, i.e., $K = g^{ab}$.

2.3 Weak Boneh-Boyen signature

The weak Boneh-Boyen [18] signature scheme has emerged as a prominent short pairing-based signature scheme in the field of cryptography. Notably, the wBB scheme is specifically designed to address the challenge of fast message signing while ensuring the highest level of security. The wBB scheme is also unique in that it permits the combination of zero-knowledge proofs with signed messages, enabling the anonymous and unlinkable proof of knowledge of signed messages and signatures.

The wBB signature scheme is based on the *q-Strong Diffie-Hellman* (q-SDH) assumption [20], which is widely considered to be a strong assumption in the standard model. This provides the utmost level of security for the wBB scheme, ensuring that it is resistant to attacks from malicious actors. The combination of fast message signing and robust security makes the wBB scheme an attractive option for use in various applications such as secure communication, electronic voting systems, and digital cash. The scheme works by using a bilinear map that enables the computation of the pairing between two elements in different groups. Specifically, the scheme uses groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of prime order p, where $\mathbb{G}_1 \neq \mathbb{G}_2$, and a group $\mathbb{G}_T$ of prime order p that are equipped with bilinear pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$.

Figure 2.2 provides a visual representation of the wBB signature scheme. The following is a detailed and technical explanation of how the scheme operates. This will enable readers to gain a thorough understanding of its inner workings and appreciate the security benefits it provides for digital communication.

- $(params, pk, sk) \leftarrow \texttt{KeyGen}\ (1^\kappa)$: on the input of the system security parameter κ, the algorithm generates a bilinear group $params = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathbf{e}, g_1, g_2)$, computes $pk = g_2^{sk}$, where $sk \in_R \mathbb{Z}_q$, and outputs $(params, pk)$ as the public key and sk as the private key.

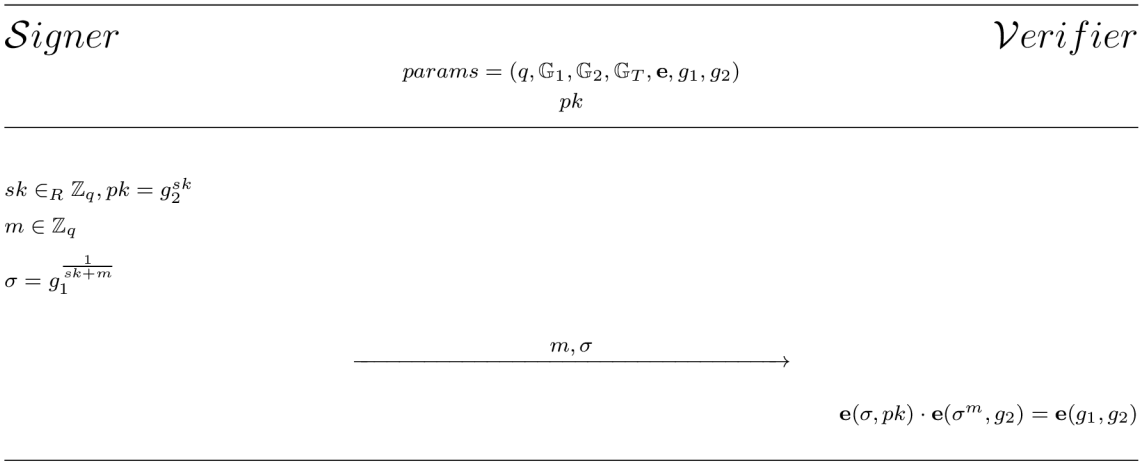


Fig. 2.2: Definition of the weak Boneh Boyen signature scheme

- $(\sigma) \leftarrow$ `Sign` $(params, sk, m)$: on the input of the message $m \in \mathbb{Z}_q$, the system security parameters $params$ and the secret key sk, the algorithm outputs the signature of the message $\sigma = g_1^{\frac{1}{sk+m}}$.
- $(1/0) \leftarrow$ `Verify` $(params, pk, m, \sigma)$: on the input of the system security parameters $params$, the public key pk, a signature σ, and a message m, the algorithm returns 1 if and only if $\mathbf{e}(\sigma, pk) \cdot \mathbf{e}(\sigma^m, g_2) = \mathbf{e}(g_1, g_2)$ holds, i.e., the signature is valid, and 0 otherwise.

2.4 Sigma protocols

Σ-protocols [19] are a class of cryptographic protocols that allow the prover to demonstrate to the verifier that they know a secret value without revealing any information about that value. We employ the Σ-protocols outlined in [21] to demonstrate the knowledge of a discrete logarithm, specifically the protocol $PK\{x : y = g^x\}$. Σ-protocols consist of three phases: commitment, challenge, and response. Each phase uses a specific cryptographic function that satisfies certain mathematical properties, such as being computationally hard to invert or finding collisions.

- *Commitment*: In this phase, the prover selects a random value and generates a commitment to the verifier. The commitment is typically a function of the secret value and some randomness.
- *Challenge*: In this phase, the verifier challenges the prover by sending a random value. The challenge serves to prevent the prover from guessing the secret value without actually knowing it. The challenge is also designed to prevent the prover from cheating by using a precomputed response.
- *Response*: In this phase, the prover generates a response to the verifier. The response is a function of the secret value, the randomness used in the commit-

ment phase, and the challenge. The verifier uses the response to verify that the prover knows the secret value without actually revealing the secret value itself.

One of the key advantages of Σ-protocols is that they can be converted into full zero-knowledge protocols, as shown by Cramer et al. [22]. This means that we can use Σ-protocols to prove knowledge of a secret without revealing any additional information beyond what is necessary. This is important in many cryptographic applications, where it is crucial to protect the confidentiality of sensitive information. Figure 2.3 illustrates an interactive zero-knowledge proof demonstration, where a prover and verifier engage in multiple rounds of communication to prove the validity of a statement without revealing any additional information beyond the validity itself.

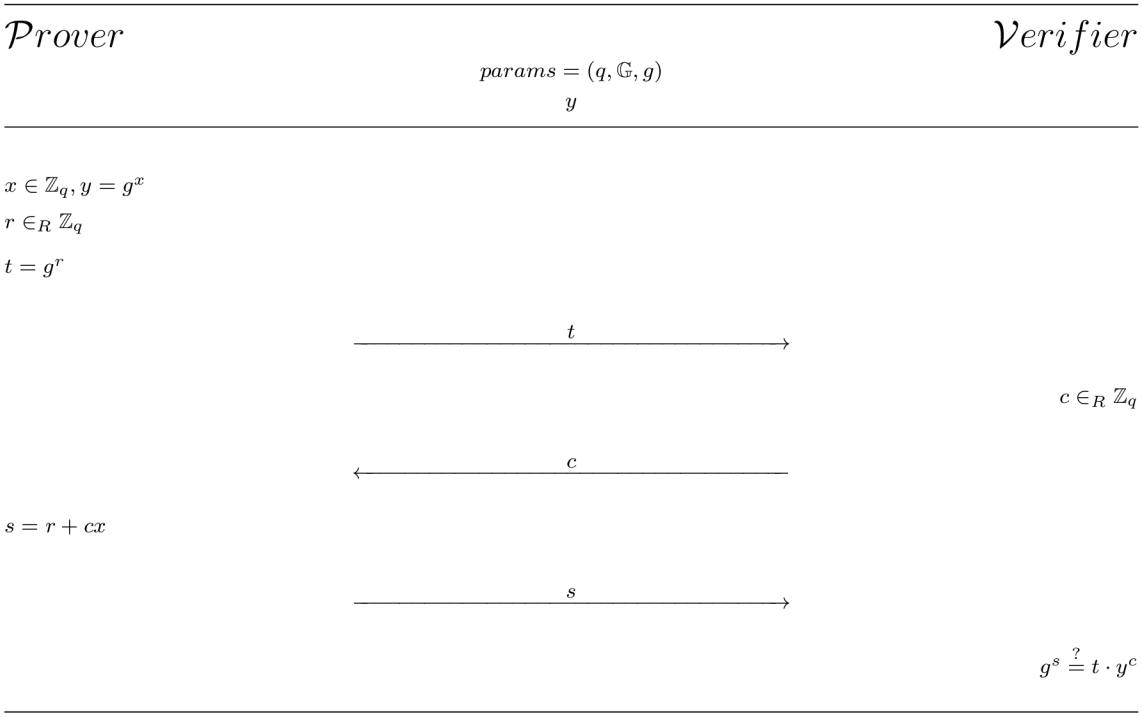


Fig. 2.3: Definition of an interactive zero-knowledge proof

In Figure 2.3, x represents the secret value that the prover wants to prove knowledge of without revealing it explicitly; y is a public value deriving from x that serves as a commitment from the prover to the verifier; r is a randomly chosen value by the prover during the proof; t is a commitment value calculated by the prover based on the random value r; c represents the challenge value generated by the verifier during the interactive proof. It is a random value selected by the verifier and sent to the prover. Finally, s is the response value computed by the prover based on the challenge value c and the secret value x.

The key idea behind this interactive zero-knowledge proof is that if the prover knows the secret value x, they can generate the appropriate responses to convince the verifier without revealing x explicitly. However, if the prover does not know x, it would be computationally infeasible for them to generate valid responses that satisfy the verifier's checks.

By relying on the techniques introduced by Fiat and Shamir [23], we can achieve the non-interactive execution of Σ-protocols with strong computational security, ensuring their resilience against potential attacks. The advantage of this approach is that it eliminates the need for repeated interactions between the prover and the verifier, resulting in a more efficient and practical protocol. In addition, the non-interactive nature of the protocol reduces the likelihood of human error during the interaction process. For the purposes of this thesis, we will only consider non-interactive zero-knowledge proof protocols. To illustrate this approach, we present a non-interactive zero-knowledge proof protocol in Figure 2.4, which enables a prover to demonstrate their knowledge of a secret value to a verifier without any further interaction.

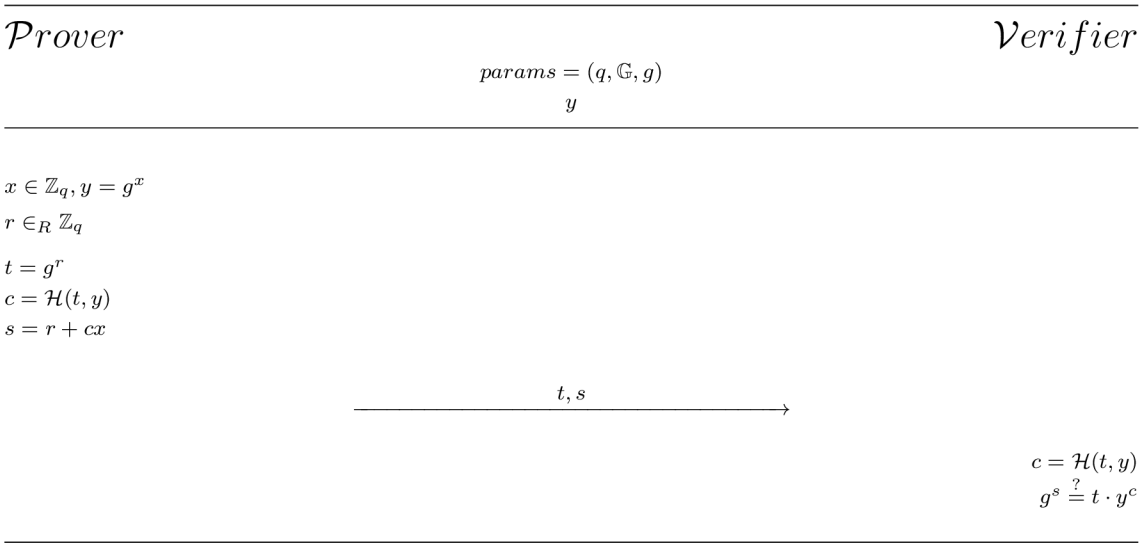


Fig. 2.4: Definition of a non-interactive zero-knowledge proof

The main difference between Figures 2.3 and 2.4 is that the interactive scheme involves multiple rounds of communication between the prover and verifier to prove knowledge of the secret. In contrast, the non-interactive scheme allows the prover to generate a proof independently. This proof can then be verified by the verifier without any further interaction.

3 Revocable attribute-based credentials on smart cards

The practical identification and revocation of misbehaving users are crucial components of ABC schemes. Unfortunately, the cryptographic protocols employed for verifying personal attributes and revoking invalid users have been developed independently, resulting in considerable difficulties with their integration. Despite the existence of efficient attribute verification [9] and generic ABC revocation schemes [3], the intricate nature of these protocols poses significant complexity challenges to their combination. This chapter presents the RKVAC protocol as a solution to the challenge of integrating cryptographic protocols for attribute verification and revocation while preserving computational feasibility for smart cards. The structure of this chapter is as follows:

- An introduction to ABCs, with a focus on their implementation on smart cards.
- A detailed explanation of the cryptographic scheme and a comprehensive security and privacy analysis.
- Implementation details and benchmark results.
- A summary of the key findings.

3.1 Introduction

Anonymous ABCs enable individuals to prove their personal attributes, such as age, citizenship, or negative SARS-CoV-2 test results, without disclosing their identities. ABCs provide a means for users to maintain control over their personal information while also engaging in activities that require the disclosure of certain attributes.

The ability to demonstrate attributes while preserving anonymity and privacy is crucial in situations where user privacy is paramount, such as in healthcare, e-commerce, and government services. By leveraging ABCs, organizations can furnish users with a mechanism for demonstrating their attributes while preserving control over their personal information. This makes ABCs a formidable tool for safeguarding user privacy and enhancing trust in digital systems.

ABCs incorporate privacy-enhancing features that further strengthen the protection of user privacy. These features include unlinkability, which prevents the exposure of sensitive information; untraceability, which prevents the user's identity from being inferred; and selective attribute disclosure, which allows users to disclose only necessary attributes. These features make ABCs an important area of research and development for promoting privacy and security in the digital age.

Although ABC schemes have been extensively studied in the literature, with seminal works by Chaum [4], Brands [5], and Camenisch and Lysyanskaya [6], implementing them practically on resource-constrained offline devices has remained a significant challenge due to the high computational complexity involved. Earlier research has shown that the implementation of core ABC protocols on smart cards was considered impractical until recently [8, 7, 3, 9]. Despite recent advances, however, efficient large-scale revocation in smart card implementations remains an open problem, with current solutions limited to online and computationally powerful devices.

3.2 State of the art

Attribute-based Credential schemes have been extensively studied in the literature, and several implementations of these schemes on programmable smart cards are available, such as those proposed by Mostowski and Vullers [7], Vullers and Alpár [8], de la Piedra et al. [24], and Camenisch et al. [9]. Nonetheless, one crucial feature that is lacking in these schemes is revocation, which is an essential feature for removing misbehaving or invalid users from the system. The topic of revocation has been extensively researched, as evidenced by numerous papers, including those by Camenisch and Lysyanskaya [25], Nguyen [26], Tsang et al. [27], Camenisch et al. [28], Tsang et al. [29], and Hajny et al. [30]. However, none of these papers proposed practical protocols that could be used in large-scale smart card applications due to numerous issues. These issues included unsupported operations (e.g., bilinear pairing on constrained user devices), periodic smart card content updates, online communication needs, loss of unlinkability, only user-driven revocation, or missing security proofs.

Lueks et al. [31] proposed a revocation scheme with low computational cost based on the *Identity Mixer* (Idemix) implementation by Vullers and Alpár [8], which was part of the *I Reveal My Attributes* (IRMA) Project[1]. However, this scheme had several disadvantages, such as limited unlinkability within one epoch and the need for revocation list re-computation for each verifier, which was impractical.

To address these issues, Verheul [32] extended the scheme proposed by Lueks et al. [31]. Nevertheless, this extension required bilinear pairings, which were currently unsupported by smart cards. Furthermore, while Camenisch et al. [3] proposed an efficient revocation scheme for smart cards, the integration of the revocation protocols with any ABC scheme was not yet described or implemented.

[1] See `https://irma.app`

More recently, Camenisch et al. [9] proposed the KVAC protocol based on algebraic *Message Authentication Code* (MAC) and *Boneh-Boyen* (BB) signatures. The protocol was specifically designed to address the challenges of implementing ABCs on smart cards. The design and implementation considered the unique features of smart cards, including their limited processing power, memory, and energy resources, while still providing efficient and secure ABC issuance and verification. Nevertheless, the scheme lacked revocation entirely, and it remained an open problem to develop efficient large-scale revocation schemes for smart card implementations that were both practical and secure.

3.3 Cryptographic scheme

In this section, we delve into a detailed analysis of the entities involved in the RKVAC protocol, as well as the cryptographic design of the algorithms that comprise the protocol.

To improve the clarity and legibility of this chapter, we included a table of all the symbols used in our cryptographic protocol. Table 3.1 defines each symbol and its associated meaning, enabling a thorough comprehension of the protocol's components.

3.3.1 Entities

The RKVAC protocol operates within a precisely defined system model consisting of several entities with distinct roles and responsibilities. To facilitate a more comprehensive understanding of the protocol's mechanisms, we hereby expound upon the entities involved in the system model depicted in Figure 3.1:

- *Revocation Authority*: plays a crucial role by assigning and issuing a unique revocation handler, denoted as the private attribute m_r, to each user during the `Issue` phase. This attribute enables the revocation authority to revoke users when necessary, thereby ensuring the security and integrity of the system.
- *Issuer*: assumes the critical responsibility of assigning and issuing personal attributes m_i to users in a cryptographic credential. This process is carried out using the `Issue` algorithm, which aggregates the user's attributes into a single credential, digitally signed by the issuer's secret key sk_I to ensure its authenticity and integrity. The system's security and functionality rely heavily on the issuer's performance, making the selection of a trusted and reliable issuer of utmost importance.
- *User*: acquires the cryptographic credential containing the personal attributes assigned by the issuer through the `Issue` algorithm. Subsequently, the user

Tab. 3.1: Table of symbols

Symbol	Definition
$q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathbf{e}, g_1, g_2$	parameters for the selected pairing-friendly elliptic curve.
$j, \alpha_1, \ldots, \alpha_j, h_1, \ldots, h_j$	parameters for the revocation authority.
$k, e_1, \ldots, e_k, \sigma_{e_1}, \ldots, \sigma_{e_k}$	user randomizers, and signed randomizers.
RL, RH, RD	revocation list, list of revocation handlers, and revocation database.
m_{ID}, m_r	user identifier and attribute for revocation.
$m_1, \ldots, m_n$	attributes with the user's information.
sk_{RA}, pk_{RA}	key pair of the revocation authority (private and public keys).
σ_{RA}	signature of the revocation authority.
sk_I, sk_V	private keys of the issuer and verifier (identical keys).
$\sigma, \sigma_{x_1}, \ldots, \sigma_n, \sigma_{x_r}$	cryptographic credential issued to the user.
$\hat{\sigma}, \hat{\sigma}_{e_I}, \hat{\sigma}_{e_{II}}, \bar{\sigma}_{e_I}, \bar{\sigma}_{e_{II}}$	cryptographic credential randomized by the user.
$\mathcal{D}$	keys to the attributes to be disclosed.
ρ	random number used to randomize the cryptographic credential.
$\rho_v, \rho_i, \rho_{m_r}, \rho_{m_{z \notin \mathcal{D}}}, \rho_{e_I}, \rho_{e_{II}}$	random numbers used to compute the protocol commitments and responses.
$t_{verify}, t_{revoke}, t_{sig}, t_{sigI}, t_{sigII}$	cryptographic commitments computed by the user.
e	challenge used in the cryptographic protocol.
$s_{m_{z \notin \mathcal{D}}}, s_v, s_{m_r}, s_i, s_{e_I}, s_{e_{II}}$	responses obtained during the execution of the cryptographic protocol.
i	unique per-session value.
C	unique one-time pseudonym.
ψ	alias of $\hat{\sigma}, \hat{\sigma}_{e_I}, \hat{\sigma}_{e_{II}}, \bar{\sigma}_{e_I}, \bar{\sigma}_{e_{II}}$.
π	alias of $e, s_{m_{z \notin \mathcal{D}}}, s_v, s_{m_r}, s_i, s_{e_I}, s_{e_{II}}$.
$t'_{verify}, t'_{revoke}, t'_{sig}, t'_{sigI}, t'_{sigII}$	cryptographic commitments reconstructed by the verifier.

leverages the `Show` algorithm to anonymously prove the possession of necessary attributes to the verifier. Furthermore, to establish the security and privacy of the system, the user must generate a one-time per-session pseudonym, denoted as C, which is linked to the credential through the revocation handler m_r to ensure non-repudiation. Lastly, the user must present evidence of the pseudonym's non-revocation status, providing proof of its legitimacy and enabling secure communication within the system.

- *Verifier*: validates the possession of the necessary attributes and the revocation status of the revocation handler m_r through the use of the `Verify` algorithm. This process requires the verifier's secret key sk_V and the revocation authority's public key pk_{RA} to ensure the authenticity and integrity of the verification. By confirming the validity of the attributes and the revo-

cation status, the verifier can grant or deny access to the requested service, maintaining the security and privacy of the system.

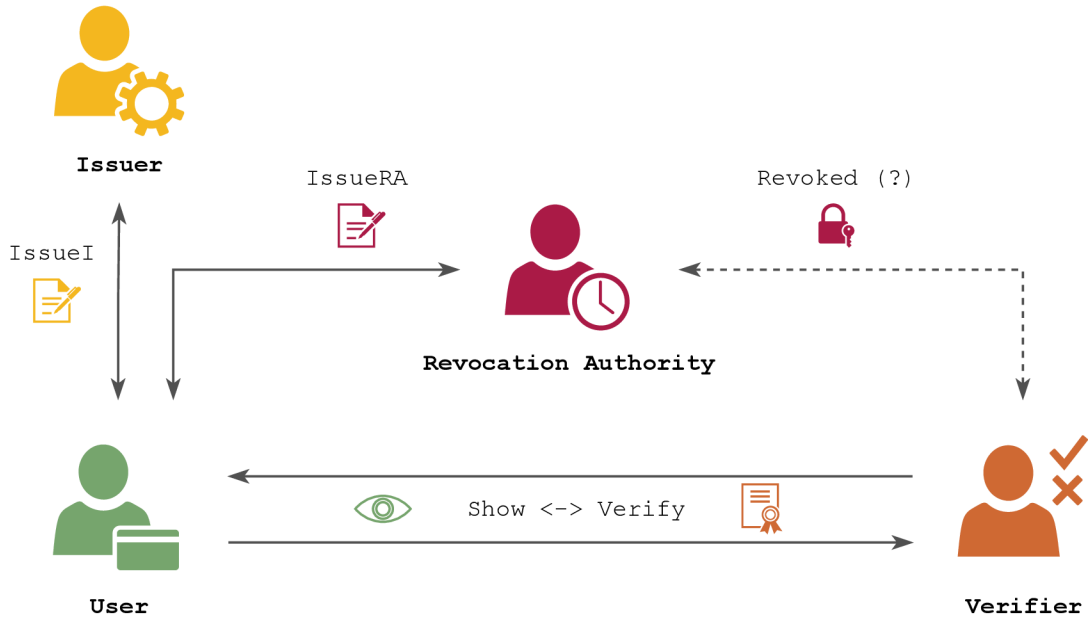


Fig. 3.1: Entities and algorithms constituting the RKVAC protocol

3.3.2 Protocol specification

The technical specifications of the algorithms and protocols are explicitly presented below, encompassing a thorough description of their input and output parameters.

- $(params) \leftarrow$ `Setup` (1^{κ}): the algorithm receives the security parameter κ as input and generates the public system parameters. These parameters are a bilinear group with parameters $params = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathbf{e}, g_1, g_2)$ that satisfy $|q| = \kappa$.
- $(params_{RA}, sk_{RA}, pk_{RA}) \leftarrow$ `SetupRA` $(params, max_{sessions})$: the algorithm inputs the public system parameters and the maximum number of unlinkable sessions per user within one epoch $max_{sessions}$. First, the algorithm randomly selects the private key $sk_{RA} \in_R \mathbb{Z}_q$ and computes the revocation authority's public key $pk_{RA} = g_2^{sk_{RA}}$, based on the system parameters $params$. Subsequently, the revocation authority establishes integers j and k such that $max_{sessions} = k^j$. The maximum number of unlinkable sessions per user within one epoch $max_{sessions}$ is set to 100 in our implementation, which means that $j = 2$ and $k = 10$. Furthermore, the revocation authority randomly selects integers $(\alpha_1, \ldots, \alpha_j) \in_R \mathbb{Z}_q$ and computes $\{h_z = g_1^{\alpha_z} \mid z \in \mathbb{Z} \wedge 1 \leq z \leq j\}$. Finally, the authority initializes an empty revocation list RL and a set containing

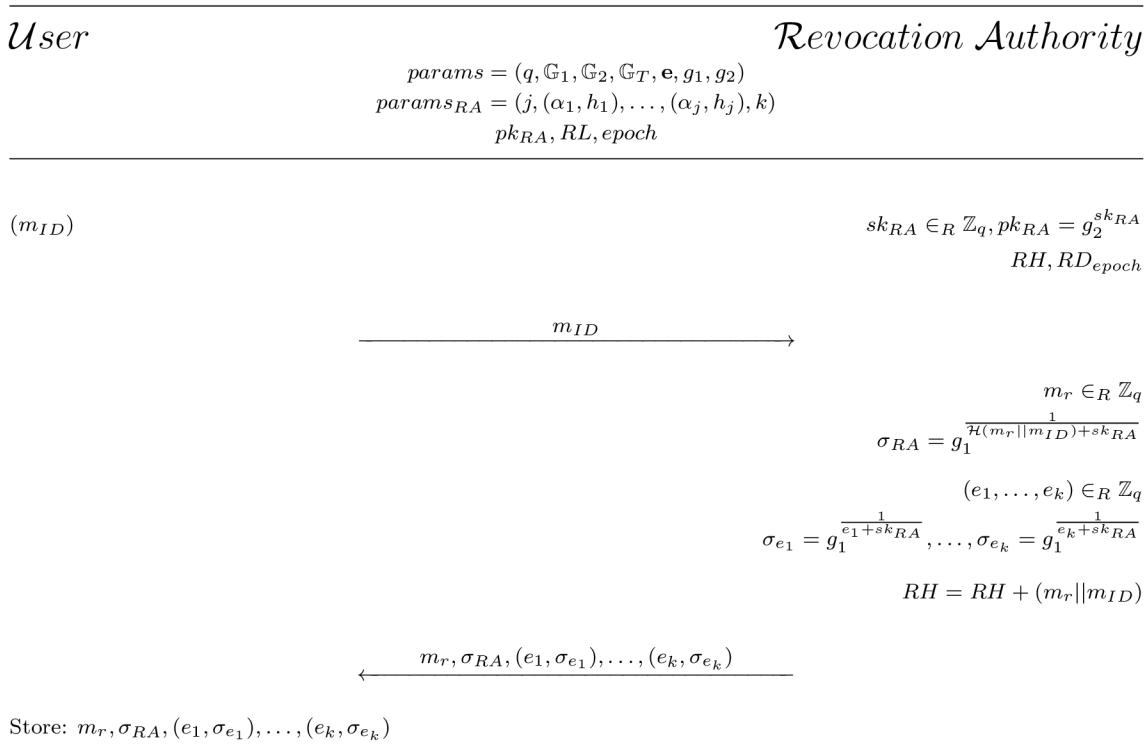


Fig. 3.2: Definition of the `Issue` algorithm, carried out by the revocation authority to emit revocation handlers

revocation handlers RH. The algorithm outputs the revocation authority parameters $params_{RA}$ and the key pair (sk_{RA}, pk_{RA}). The revocation authority runs the `SetupRA` algorithm.

- $(sk_I, sk_V) \leftarrow$ `SetupIV` $(params)$: the algorithm randomly selects the private keys $sk_I \leftarrow (x_0, x_1, \ldots, x_n, x_r) \in_R \mathbb{Z}_q$ based on the system parameters $params$. In our design, we presuppose that the issuer and verifier are one and the same entity; henceforth, their keys are identical $sk_I = sk_V$. The algorithm outputs the issuer and verifier keys (sk_I, sk_V). The issuer runs the `SetupIV` algorithm.
- $(m_r, \sigma_{RA}, (e_1, \sigma_{e_1}), \ldots, (e_k, \sigma_{e_k})) \leftarrow$ `IssueRA` $(params, params_{RA}, m_{ID})$: the algorithm inputs the user identifier m_{ID}. The algorithm is run between the user and the revocation authority and is shown in Figure 3.2. First, the revocation authority proceeds to generate the revocation attribute randomly $m_r \in_R \mathbb{Z}_q$ and subsequently sign the revocation attribute m_r and the user identifier m_{ID} with the secret keys as $\sigma_{RA} = g_1^{\frac{1}{\mathcal{H}(m_r||m_{ID})+sk_{RA}}}$. Afterward, the algorithm generates randomizers $(e_1, \ldots, e_k) \in_R \mathbb{Z}_q$ and signs each one $\{\sigma_{e_z} = g_1^{\frac{1}{e_z+sk_{RA}}} \mid z \in \mathbb{Z} \wedge 1 \leq z \leq k\}$. The algorithm outputs the revocation attribute m_r, the revocation authority's signature σ_{RA}, as well as the corresponding randomizers, each accompanied by their respective signatures

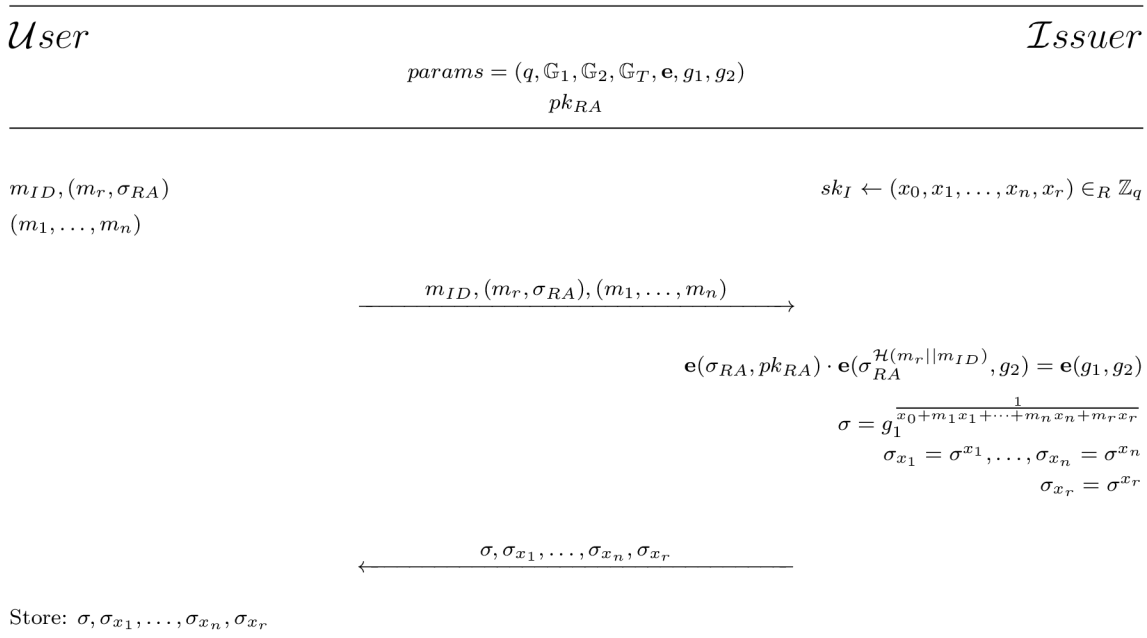


Fig. 3.3: Definition of the `Issue` algorithm, carried out by the issuer to emit personal attributes

$(e_1, \sigma_{e_1}), \ldots, (e_k, \sigma_{e_k})$.

- $(\sigma, \sigma_{x_1}, \ldots, \sigma_{x_n}, \sigma_{x_r}) \leftarrow$ `IssueI` $(params, pk_{RA}, m_{ID}, m_r, \sigma_{RA}, m_1, \ldots, m_n)$: the algorithm inputs the public key of the revocation authority pk_{RA}, the user identifier m_{ID}, the revocation attribute m_r, the signature of the revocation authority σ_{RA}, and the user attributes $m_1, \ldots, m_n$. The algorithm is run between the user and the issuer and is shown in Figure 3.3. The algorithm begins with the issuer verifying the authenticity of the revocation authority's signature. Once verification is complete, the issuer proceeds to sign the user attributes with the secret keys as $\sigma = g_1^{\frac{1}{x_0+m_1x_1+\cdots+m_nx_n+m_rx_r}}$. Thereafter, the issuer calculates the auxiliary values $\sigma_{x_1}, \ldots, \sigma_{x_n}, \sigma_{x_r}$. The algorithm outputs the cryptographic credential σ and auxiliary values $(\sigma_{x_1}, \ldots, \sigma_{x_n}, \sigma_{x_r})$ to the user.
- $(C, \psi, \pi, m_{z \in \mathcal{D}}) \leftarrow$ `Show` $(params, params_{RA}, nonce, epoch)$: the algorithm receives the *timestamp*, the *nonce*, and the keys to the attributes to be disclosed $\mathcal{D}$ from the verifier as inputs. The algorithm outputs the pseudonym C, the randomized user credentials ψ, the cryptographic proof of possession of the attributes π, and the disclosed attributes $m_{z \in \mathcal{D}}$. Users execute the `Show` algorithm. Figure 3.4 depicts a comprehensive explanation of the `Show` algorithm. The user begins by selecting a unique per-session value i and calculating their transaction pseudonym C based on the received *epoch*. Next, the user proceeds

Fig. 3.4: Definition of the `Show` and `Verify` algorithms

by randomizing their credentials. The user then calculates the commitments t_{verify}, t_{revoke}, t_{sig}, t_{sigI}, and t_{sigII}. Lastly, the user computes a proof of knowledge for all the attributes in the credential that are not disclosed.

- $(0/1) \leftarrow$ `Verify` $(params, params_{RA}, pk_{RA}, C, \psi, \pi, m_{z \in \mathcal{D}})$: the algorithm inputs the pseudonym C, the randomized user credentials ψ, the cryptographic

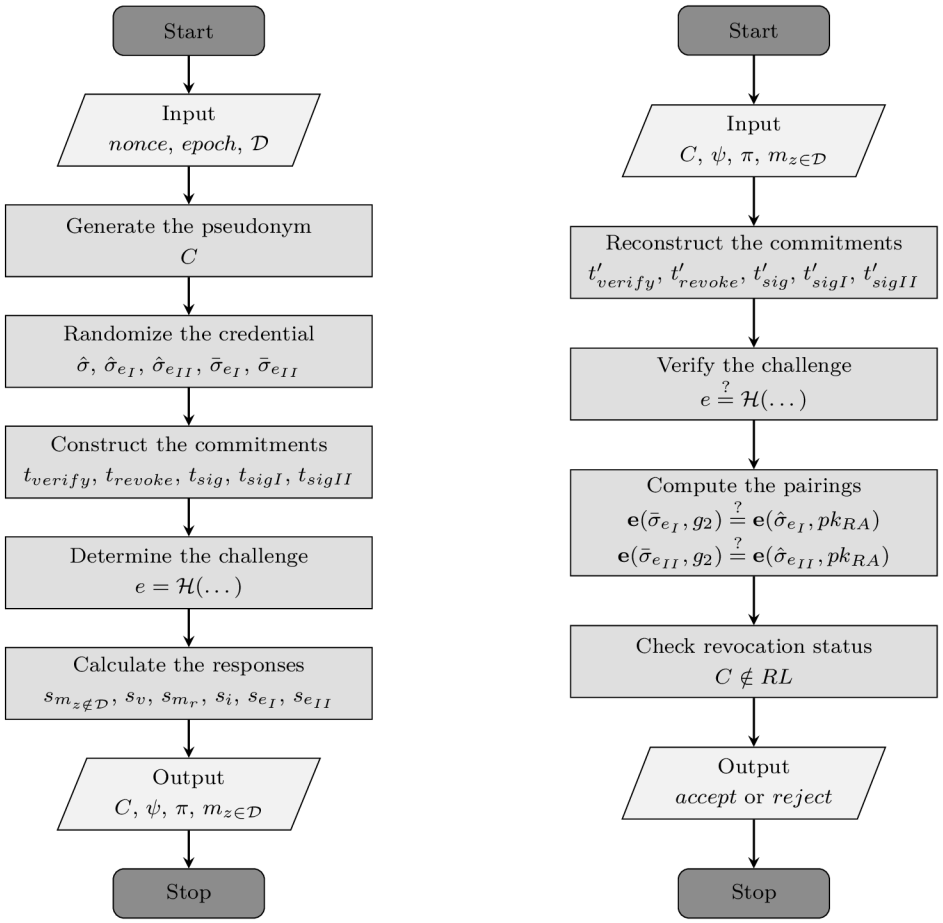
proof of possession of the attributes π, and the disclosed attributes $m_{z\in\mathcal{D}}$. Verifiers execute the `Verify` algorithm. Figure 3.4 depicts a comprehensive explanation of the `Verify` algorithm. The verifier begins by reconstructing the commitments t'_{verify}, t'_{revoke}, t'_{sig}, t'_{sigI}, and t'_{sigII}. The verifier then uses the commitments to create the cryptographic hash $\mathcal{H}$ and checks that the value of e received from the user matches. Lastly, the verifier computes bilinear pairings to ensure that the randomized credentials correspond to a genuine user of the system and were emitted by the revocation authority and the issuer. The algorithm concludes with the verifier undertaking the verification process to ensure that the pseudonym has not been revoked.

- $(RL) \leftarrow$ `Revoke` (C): the algorithm receives the one-time pseudonym C to revoke as input. The algorithm outputs the updated revocation list RL. The algorithm is run between the verifier and the revocation authority. The revocation authority initiates the process by computing all pseudonyms associated with each user in the system for a particular epoch $C = g_1^{\frac{1}{i-m_r+\mathcal{H}(epoch)}}$. Notably, the value i is determined for every possible combination of α_j and e_k. In our implementation, $j = 2$ and $k = 10$, thereby resulting in 100 pseudonyms for each epoch. Furthermore, the revocation attribute m_r is acquired from the list of revocation handlers RH. Subsequently, the revocation authority identifies the owner of the pseudonym, and it should be emphasized that the revocation authority maintains a comprehensive list of revocation attributes that are correlated with the users' identities. Upon identifying the user, the revocation authority adds all the pseudonyms to the revocation list RL to prevent the user from accessing the system.

The flowcharts depicted in Figure 3.5 provide a high-level description of the `Show` and `Verify` algorithms and are a valuable resource for gaining a thorough understanding of the protocols, their underlying mechanisms, and the succession of steps involved in their execution. By illustrating the key components of each algorithm and their interrelationships, the flowcharts enable readers to rapidly comprehend the fundamental concepts of our cryptographic protocol.

3.4 Security and privacy discussion

This section examines the security and privacy aspects of our novel Revocable Keyed-Verification Anonymous Credential protocol for systems with access control requirements. The protocol utilizes a strong authentication mechanism to guarantee secure access to the system. Our discussion validates the vital security, privacy, and functionality properties that the protocol offers. Our analysis confirms that the proposed



(a) High-level `Show` algorithm

(b) High-level `Verify` algorithm

Fig. 3.5: High-level definition of the `Show` and `Verify` algorithms

protocol satisfies these security and privacy requirements, establishing it as a reliable solution for securing access control in such systems. For a more formal and comprehensive analysis, please refer to Appendix A.

3.4.1 Required properties

We scrutinize the crucial security, privacy, and functionality properties, including anonymity, unlinkability, and untraceability; correctness; key-parameter consistency; as well as unforgeability, completeness, soundness, and zero-knowledge, that the protocol must provide to guarantee the security and privacy of access control systems.

Anonymity, *unlinkability*, and *untraceability* are of the utmost importance to ensure the user's privacy. The anonymity property guarantees that a party's identity remains hidden or confidential throughout a protocol's execution, preventing it from being linked to any exchanged information. The unlinkability property ensures that

various protocol executions cannot be linked to the same party or identity, preventing any correlation of the party's actions or information across multiple protocol executions. Lastly, the untraceability property guarantees that a party's actions or exchanged information during a protocol execution cannot be traced back to the party, avoiding any identification of the party's actions through any means, including network monitoring or traffic analysis. Our protocol achieves these properties of anonymity, unlinkability, and untraceability through the utilization of standard zero-knowledge protocols. The use of randomized user credentials in each protocol execution ensures that user identification or tracing is prevented, thus enhancing the security and privacy of the protocol. These properties have been formally proven in Proposition 5 of our security and privacy analysis.

Correctness is critical as it ensures the accuracy of information exchanged during a protocol's execution and that it is in line with the intended specification. This property guarantees that the protocol achieves its desired goal while preventing errors or misunderstandings in the exchange of information. Our protocol guarantees correctness by utilizing advanced cryptographic techniques such as commitment reconstruction, hash functions, and pairings. These techniques help to ensure that the information exchanged during the protocol execution is accurate and reflects the state of the system. This is achieved by verifying the integrity of the data to ensure that it has not been tampered with or altered in any way. This property has been formally proven in Proposition 2 of our security and privacy analysis.

Key-parameter consistency is crucial to ensuring the security of the protocol, as it guarantees the validity and consistency of the keys and parameters used during the protocol's execution. This property prevents the protocol from relying on faulty or invalid keys or parameters that could be exploited by an attacker. Our protocol achieves key-parameter consistency by generating keys and parameters from truly random sources, making them unpredictable and secure. Furthermore, we validate their authenticity and consistency before using them in the protocol, ensuring that only valid keys and parameters are used. This approach helps prevent attacks that exploit weaknesses or vulnerabilities in the keys and parameters used in the protocol. This property has been formally proven in Proposition 4 of our security and privacy analysis.

Unforgeability, *completeness*, *soundness*, and *zero-knowledge* are indispensable to the security and privacy of our protocol. The unforgeability property ensures that only authorized entities can produce valid signatures, thereby preventing adversaries from forging signatures without access to the signer's private key. The completeness property guarantees that the protocol execution is exhaustive and leaves no loopholes or opportunities for attacks, ensuring the overall security and reliability of the protocol. This property ensures that all valid inputs are accepted and that

the protocol produces a valid output, leaving no room for ambiguity or incomplete execution. The soundness property provides verifiable and irrefutable evidence during protocol execution that cannot be tampered with or disputed. It guarantees that any invalid inputs are rejected and further strengthens the protocol's security and integrity. Finally, the zero-knowledge property ensures that parties can prove their knowledge of secrets without revealing any more information than necessary. Our protocol achieves those properties by utilizing standard signature and zero-knowledge protocols. Specifically, the protocol leverages well-established techniques such as the weak Boneh-Boyen digital signature, the Fiat-Shamir transform, and the Schnorr identification scheme to achieve these goals. This guarantees that the protocol is both secure and efficient, while also providing the necessary guarantees to protect the privacy of users in the system. These properties have been formally proven in Propositions 1, 2, and 3 of our security and privacy analysis.

3.5 Implementation details

This section provides an overview of the design process and highlights the key points considered during the development of the application. Table 3.2 presents the smart card we utilized and its hardware and software specifications. The hardware and software specifications of the smart card encompass an array of technical details. These range from the MULTOS smart card information to the memory capacities of the *Electrically Erasable Programmable Read-Only Memory* (EEPROM), *Read-Only Memory* (ROM), and *Random Access Memory* (RAM), as well as the size of the *Application Protocol Data Unit* (APDU) buffer. Additionally, the *Microcontroller Unit* (MCU) and the communication protocol used by the smart card are also included in these specifications.

Tab. 3.2: Hardware and software specifications of the MULTOS smart card

Specification	Value
MULTOS model	ML4
MULTOS version	4.3.1
SmartDeck SDK	3.4.0.0
MULTOS Utility (MUtil)	2.11.1
MCU	SC23Z018
EEPROM size	18 KB
ROM size	250 KB
RAM size	1.28 KB
APDU buffer size	255 bytes
Communication protocol	T=0

In this cryptographic protocol implementation, the system is divided into two distinct parts: the desktop application and the smart card application. The desktop application acts as the revocation authority, issuer, and verifier, while the smart card application represents the user.

The protocol was designed and implemented using elliptic curve cryptography. Specifically, for the desktop application, we utilized the *Barreto-Naehrig 254-bit* (BN254) curve supplied by the `mcl` library [33]. The `mcl` library offers a comprehensive set of functionalities for pairing-based cryptography and is well-suited for use in cryptographic protocol implementations [34]. Moreover, we used the `OpenSSL` library [35] for the hash functions required by the protocol.

In contrast, to design the elliptic curve functions for the smart card application, we employed the MULTOS assembly provided by the SmartDeck *Software Development Kit* (SDK) [36, 37, 38, 39]. The MULTOS assembly reference [38] was utilized to construct essential functionalities like modular addition, subtraction, multiplication, elliptic curve addition, elliptic curve scalar multiplication, and other pertinent functions. Through this SDK, the smart card application can proficiently execute the required cryptographic computations while ensuring high-level security.

By harnessing industry-standard libraries and SDKs, the system can conduct the essential cryptographic computations with high accuracy and dependability. The utilization of the `mcl` and the `openssl` libraries for the desktop application and the SmartDeck SDK for the smart card application guarantees that the protocol is ideal for deployment in diverse applications that prioritize security.

Finally, for the installation of the smart card application, we utilized MUtil [40], a specialized software tool designed to facilitate the installation of applications onto smart cards. MUtil offered several advantages in the implementation process, including efficient and reliable loading of the application onto the smart card, ensuring the seamless integration of cryptographic functions. Its usage significantly contributed to the overall effectiveness of the cryptographic protocol implementation by simplifying the installation process while maintaining the required level of security.

3.6 Experimental results

This section outlines the outcomes of our cryptographic protocol implementation under varying numbers of user attributes stored on the smart card. To ascertain the practicality and assess the efficiency and speed of the algorithms, we conducted a series of experiments, with a focus on benchmarking the `Show` algorithm. Figure 3.6 presents the results of these experiments, showcasing the benchmarks in milliseconds and accounting for protocol run times and communication overhead.

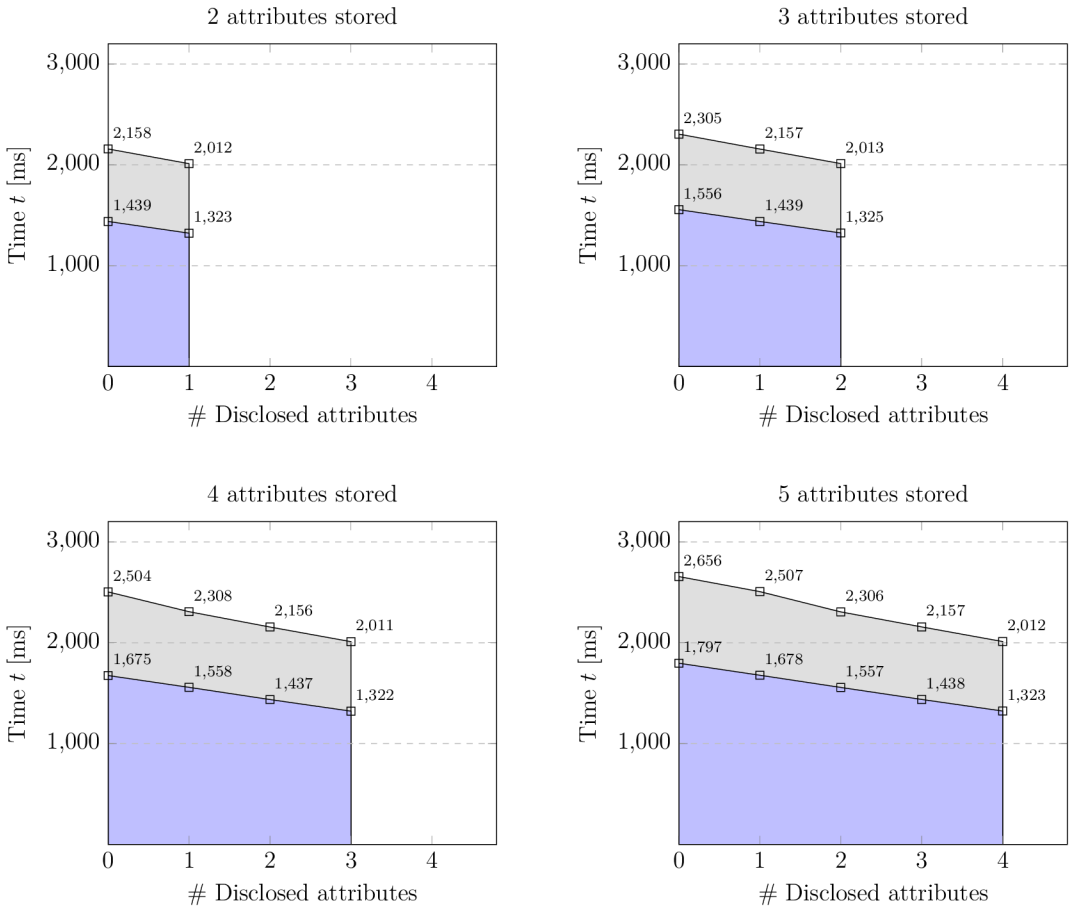


Fig. 3.6: Speed comparison of the `Show` algorithm and the transmission overhead

Each graph illustrates the runtime of the `Show` algorithm for varying numbers of attributes in the credential, ranging from 2 to 5. The runtime measurements comprise the time taken to compute the proof of knowledge while incrementally disclosing attributes. It is worth noting that the revocation attribute m_r remains hidden at all times; therefore, the maximum number of attributes that can be disclosed is $n - 1$. For instance, if the credential contains 5 attributes, the maximum number of attributes that can be disclosed is 4.

When all attributes are disclosed, the runtime is similar across all graphs, irrespective of the number of attributes in the credential. However, if an attribute is not disclosed, the time taken to generate the proof of knowledge increases by approximately 150 milliseconds for each undisclosed attribute. The computation time for generating the proof increases with the number of undisclosed attributes because the proof generation algorithm must execute several cryptographic operations for each undisclosed attribute, which requires additional processing power and time.

Furthermore, credentials containing more attributes take longer to compute the proof than those with fewer attributes. When all attributes are disclosed, the proof

generation algorithm can compute the proof faster since it does not need to execute cryptographic operations on undisclosed attributes.

The `Show` algorithm executes the fastest in 1,322 milliseconds without factoring in communication overhead and approximately 2 seconds with communication overhead. On the other hand, the slowest execution time is directly proportional to the number of attributes contained within the credential and, thus, grows linearly as the number of attributes increases. The communication overhead for all transmissions is around 700 milliseconds. Since the T=0 protocol is used, it is not feasible to reduce communication time. Lastly, it is worth noting that the `Verify` algorithm benchmarks are omitted since their execution time on a Raspberry Pi 4 Model B is in the order of microseconds.

3.7 Summary

This chapter focuses on the practical aspects of ABC technologies, presenting an integrated scheme that encompasses all privacy-preserving features and provides efficient revocation based on provably secure building blocks.

The scheme has been thoroughly benchmarked and implemented on a standard smart card, representing the first-ever implementation of a revocable anonymous credential scheme that achieves practical running times. In addition, the proposed scheme incorporates all privacy-enhancing features and provides efficient revocation, even in applications that serve millions of users. Its implementation and benchmarking on a standard smart card further enhance its practicality.

We strongly believe that the results highlighted in this chapter will make a significant contribution towards the practical implementation of ABC technologies. This is due to the convenience of smart cards, which are considered the most suitable devices for storing and verifying personal attributes given their high levels of security, durability, and portability. These technologies can be deployed in numerous applications, such as *electronic ID* (e-ID) cards, e-ticketing, mass transportation, privacy-enhanced person tracing, and smart quarantine.

The contents of this chapter draw upon the findings published by Hajny et al. [11]. The principal contributions of the PhD candidate to the research presented in this chapter encompass the conceptualization, implementation, and benchmarking of the smart card application. Furthermore, the candidate has dedicated efforts towards optimizing the application to fully leverage the capabilities bestowed by the MULTOS operating system. Ultimately, the candidate has designed and implemented the desktop application, establishing seamless communication with the smart card through the transmission of APDU commands via *Personal Computer / Smart Card* (PC/SC) protocols.

4 Boosting revocable attribute-based credentials on Java Cards

In the previous chapter, we introduced an ABC scheme for smart cards and demonstrated its performance on MULTOS-based smart cards. While these cards offer numerous benefits, they are less accessible to the general consumer. Additionally, support for cryptographic operations on elliptic curves is typically only available upon request. In contrast, Java Cards are more widely available. Unfortunately, their support for elliptic curve cryptography is severely limited. This chapter outlines strategies for optimizing the coprocessor's capabilities to enable ABC schemes on Java Cards. The chapter is structured as follows:

- An introduction to Java Card technology, including its cryptographic support and limitations.
- A comprehensive explanation of the transformations required for the implementation of modular arithmetic and elliptic curve operations.
- A detailed account of the implementation process for the KVAC and RKVAC schemes, along with benchmark results and performance analysis.
- A summary of the key findings.

4.1 Introduction

Smart cards are gaining widespread acceptance and usage, finding diverse applications, including access control, toll payment, public transport tickets, and user e-ID. With this increased usage comes the need to consider security management and user privacy. To address these concerns, anonymous credential schemes and cryptographic protocols are in high demand. These approaches enable users to prove their identity based on personal attributes, preserving anonymity and enabling unlinkable and untraceable transactions. For instance, users can prove their university status (student or professor) without revealing their name or their legal age without disclosing their date of birth. These schemes have been successfully developed on several platforms, including smart cards. Nevertheless, it is difficult to find an off-the-shelf smart card that provides all the necessary cryptographic operations.

Java Card technology has emerged as a popular choice for smart card application development. Its use of the Java programming language enables full interoperability of its applications between different manufacturers. This has fueled the adoption of the platform, and its popularity has increased considerably over the last few years. However, the large interoperability between vendors and the versatility and ease of the platform come at a price. Developers do not have direct access to the underlying

cryptographic components, such as the cryptographic coprocessor, and rely solely on the API for the functionality they require. Although the Java Card API supports numerous common cryptographic algorithms, it is incapable of covering the large number of cryptographic schemes currently in use. Indeed, it doesn't even support basic algebraic operations like modular arithmetic or *Elliptic Curve* (EC) operations, hindering the implementation of novel cryptographic schemes like ABCs.

4.2 State of the art

Several studies have explored the implementation of cryptographic operations on smart card platforms. Malina and Hajny [41, 42] proposed three multiplication techniques for the .NET smart card platform, including the classical, Comba's, and Montgomery multiplication methods. They also implemented a technique (ab)using RSA to take advantage of hardware acceleration. In the comparison, they show how the RSA Tunnel is the best option with large integers, while the basic multiplication techniques become very slow. Dzurenda et al. [34] provided an overview of the ability to compute elliptic curve operations on different operating systems for smart cards, such as MULTOS or Java Card. Focusing on the Java Card platform, they could not perform a strict comparison due to the lack of smart cards on the market that supported the required API functions. Malina et al. [43] evaluated security and cryptographic support on different smart card platforms and found that basic arithmetic operations on the Java Card platform are not supported.

More recently, Mavroudis and Svenda [44] introduced *JCMathLib* [45], an open-source library for low-level cryptographic operations in Java Cards that extends the Java Card API and does not rely on a proprietary interface. This library enables developers to implement cryptographic algorithms not supported by the Java Card specification.

Up until now, there have been very few implementations of anonymous credential schemes for Java Cards. However, the execution times are not very practical or do not allow the revocation of malicious or expired users. Bichsel et al. [46] implemented the Camenisch-Lysyanskaya (CL) anonymous credential system on a standard Java Card. They achieved good results for large key sizes by exploiting the RSA interface and efficiently managing smart card resources. Similarly, Sterckx et al. [47] implemented an anonymous authentication scheme for authenticating a hardware module remotely called Direct Anonymous Attestation (DAA). They mainly focused on efficiently solving modular multi-exponentiation and large number multiplication using RSA.

Unlike the aforementioned works, which concentrated on the Java Card platform, the following publications focused on MULTOS-based implementations. Mostowski

and Vullers [7] showed the implementation of Microsoft's U-Prove technology, while Vullers and Alpár [8] presented the first implementation of IBM's Idemix technology, permitting selective disclosure directly on the smart card. Correspondingly, Hajny and Malina [1], Hajny et al. [30, 48], and Dzurenda et al. [49] focused on implementations of anonymous credentials with practical revocation on MULTOS-based smart cards. Finally, Camenisch et al. [9] implemented a novel cryptographic scheme for anonymous ABCs, designed primarily for smart cards. They went further and added the user revocation in [11]. However, their implementation is available only for MULTOS-based smart cards. Unfortunately, all MULTOS-based implementations rely on modular arithmetic APIs provided directly by the smart cards or on specific elliptic curve APIs supported exclusively by custom-built cards and not by off-the-shelf cards.

4.3 Java Card technology

In this section, we present a comprehensive overview of the Java Card technology, its usability for the development of security applications, and its cryptographic support.

Java Card technology is a subset of the Java programming language that is specifically designed for use in small, resource-constrained devices such as smart cards. It enables the secure execution of Java-based applets on these devices. Java Card achieves great interoperability and platform independence by combining the *Java Card Virtual Machine* (JCVM) and standard libraries with a well-defined and documented API. This allows the same applet to run on different smart cards while maintaining the highest certification levels and standard compliance, and permits applets to be developed on one platform and deployed on another with minimal modification. The JCVM is a highly optimized runtime environment that is designed to run on resource-constrained devices. It includes a set of core Java classes, such as the `Object` class, as well as Java card-specific classes that provide access to the smart card's hardware resources.

Java Card Applets are small programs that run on the JCVM and provide specific functionality to the smart card. They are written in Java and are compiled into bytecode that is compatible with the JCVM. Applets are loaded onto the smart card and can be updated or deleted as needed. They are state machines that respond to commands received via the reader device by transmitting and receiving status codes and data.

One of the key features of Java Card technology is its security architecture. Java Card applets are executed within a secure sandbox environment that isolates them from the rest of the smart card's operating system and other applets. This provides a high degree of security and prevents unauthorized access to sensitive data or

functions. In addition to its security features, Java Card technology provides several benefits for smart card developers. Because it is based on the Java programming language, it is easy to learn and use and provides a familiar development environment for Java developers.

Java Card technology is used in applications such as secure access control systems, electronic payment systems, and public transportation systems. These applications rely on the security, interoperability, and platform independence of Java Card technology to provide secure and reliable services to users.

4.3.1 Cryptographic support

The Java Card API provides extensive support for a multitude of standard cryptographic algorithms, including symmetric encryption protocols, public-key cryptosystems, and message digest algorithms. One particularly notable feature of the Java Card API is its support for elliptic curve cryptography, which has gained increasing attention as a viable alternative to conventional public-key cryptography algorithms.

Elliptic Curve Cryptography (ECC) is based on the mathematical theory of elliptic curves and is known for its superior security characteristics compared to traditional public-key cryptosystems. Regrettably, the Java Card API lacks built-in support for essential elliptic curve primitives, including point addition, scalar point multiplication, and modular arithmetic, among other crucial algebraic operations necessary for ECC implementation.

It is worth noting that the Java Card API provides only partial access to the underlying algebraic operations required for cryptographic algorithms. As a result, it is impractical to construct efficient non-standard ECC applications using Java Card. Consequently, the current level of elliptic curve operations support provided by the Java Card API is inadequate for implementing non-standard cryptographic applications that require the use of ECC, such as attribute-based credential schemes.

4.4 Efficient derivation of low-level primitives

Since the Java Card API does not provide access to fundamental arithmetic operations, it is necessary to develop them manually. However, software implementation would considerably increase execution time and reduce performance. We present some generic transformations to efficiently derive low-level primitive operations from high-level ones, enabling the execution of modular and elliptic curve operations using the smart card coprocessor. We supply transformations for all operations needed

to implement the KVAC and RKVAC schemes. Likewise, we distinguish two types of low-level operations: *(i)* modular arithmetic; and *(ii)* elliptic curve operations.

4.4.1 Modular arithmetic

To construct KVAC and RKVAC schemes, we need the following modular operations:

Modular addition: This operation adds two operands $a, b \in \mathbb{Z}_q$, reduces the result of the addition modulo a given modulus q, and outputs $c \in \mathbb{Z}_q$, i.e., $c \equiv a+b(mod\, q)$.

Modular subtraction: This operation calculates the modular subtraction of two operands $a, b \in \mathbb{Z}_q$, reduces the result modulo a given modulus q, and outputs $c \in \mathbb{Z}_q$, i.e., $c \equiv a - b\,(mod\, q)$.

Modular negation: This operation calculates the modular negation $-a \in \mathbb{Z}_q$ of a value $a \in \mathbb{Z}_q$. A modular negation of an integer a is the integer $-a \equiv q - a\,(mod\, q)$.

Modular multiplication: This operation calculates the multiplication of two operands $a, b \in \mathbb{Z}_q^*$, reduces the result modulo a given modulus q, and outputs $c \in \mathbb{Z}_q^*$, i.e., $c \equiv a \cdot b\,(mod\, q)$. Since the software implementation of multiplication is a very time-consuming procedure, it is possible to deploy a binomial formula which converts modular multiplications to modular exponentiations, allowing hardware acceleration using a coprocessor [42]:

$$a \cdot b \equiv ((a+b)^2 - a^2 - b^2)/2\,(mod\, q) \tag{4.1}$$

To speed up this execution, the squares of numbers can be calculated directly on the smart card coprocessor using RSA by setting the private key to value 2. The multiplication is therefore converted into a modular addition, two modular subtractions, and a modular division. Modular division by two with a prime number as modulus is a simple modular right shift. If the dividend's *Least Significant Bit* (LSB) is zero, modular division by two is basically a right shift. If the LSB is one, we must add the dividend and the modulus, and then right shift the result.

Modular inversion: This operation calculates the modular inverse $b^{-1} \in \mathbb{Z}_q^*$ of a value $b \in \mathbb{Z}_q^*$. A modular inverse of an integer b is the integer b^{-1} such that $b \cdot b^{-1} \equiv 1\,(mod\, q)$. Similarly to multiplication, the inverse can be achieved using RSA. Furthermore, on some platforms, the private key (i.e., exponent) cannot be a negative value, and therefore, it cannot be directly utilized for the calculation. To be able to use the coprocessor, we can use the following transformation considering Fermat's little theorem as suggested in [44].

$$b^{q-1} \equiv 1 \,(mod\, q) \rightarrow b^{-1} \equiv b^{q-2} \,(mod\, q) \tag{4.2}$$

Modular reduction: This primitive essentially reduces an operand to a modulus q.

4.4.2 Elliptic curve arithmetic

To construct KVAC and RKVAC schemes, we need two elliptic curve operations:

Point addition: This operation adds two points on an elliptic curve $A, B \in E(\mathbb{F}_p)$ and outputs $C \in E(\mathbb{F}_p)$ such that $C = A + B$. This operation is not supported by Java Card by default. However, Java Card API 3.0.5 supports the *Password Authenticated Connection Establishment* (PACE) protocol [50] for authenticated key agreement. In particular, Java Card 3.0.5 supports *Generic Mapping* (GM) according to TR03110 [51]. The PACE performs the following calculation, which can be bypassed in favor of performing a simple point addition:

$$\hat{G} = s \cdot G + H \tag{4.3}$$

Where $s \in \mathbb{Z}_q$ (q is the curve order) is provided as EC private key value (protected by a user password), $G \in E(\mathbb{F}_p)$ is EC base point, and $H \in E(\mathbb{F}_p)$ is an auxiliary value derived from the Elliptic Curve Diffie-Hellman protocol. This algorithm can be used to compute the point addition by setting the EC private key value $s = 1$, while points $G = A$ and $H = B$ are set to two points we want to sum, i.e., $C = A + B = 1 \cdot G + H$.

Scalar point multiplication: This operation calculates the multiplication of an EC point by a scalar value on an elliptic curve. This operation is not supported by Java Card by default. However, the Java Card API supports the *Elliptic Curve Diffie-Hellman* (ECDH) protocol. Furthermore, since Java Card 3.0.5, it is possible to return an established ECDH secret without its hashing by *Secure Hash Algorithm 1* (SHA-1), as per *Institute of Electrical and Electronics Engineers* (IEEE) P1363. The ECDH performs the following calculation, which can be bypassed in favor of performing a simple scalar point multiplication:

$$K = a \cdot B \tag{4.4}$$

Where $a \in \mathbb{Z}_q$ (q is the curve order) is an ECDH secret key of side A, and $B \in E(\mathbb{F}_p)$ is an ECDH public key of side B. This operation produces the shared secret $K \in E(\mathbb{F}_p)$. The result of the EC scalar point multiplication is in uncompressed

and not hashed form. Therefore, this algorithm can be used to compute the scalar point multiplication by setting a and B to the values we want to multiply.

4.5 Implementation details

This section describes the development process and summarizes the main key points considered during the design of the application. Furthermore, we detail the implementation of the modular and elliptic curve operations introduced in Section 4.4. Moreover, we outline the techniques employed to optimize the KVAC and RKVAC implementations and speed up their executions.

First and foremost, we used `JCAlgTest` [52], an automated testing tool, to determine the cryptographic algorithms supported by Java-based smart cards. Based on these results and considering the algorithms we need for the implementation of KVAC and RKVAC schemes, we chose the Java Card J3H145. Table 4.1 shows the technical specifications of this smart card.

Tab. 4.1: Hardware and software specifications of the Java Card smart card

Specification	Value
Java Card model	NXP JCOP3 J3H145
Java Card version	3.0.4 Classic
Global Platform version	2.2.1
MCU	P60D144
EEPROM size	144 KB
ROM size	112 KB
RAM size	2.558 KB
APDU buffer size	261 bytes
Communication protocol	T=1

One of the key advantages of this card is the combination of accessible data memory and support for the required cryptographic algorithms. On the other hand, its reduced RAM capacity makes it impossible to run the whole protocol on it. This slows down the computation and, therefore, the execution time.

4.5.1 Application design

We built the Java Card application using the *Model-View-Controller* (MVC) design paradigm. Using this technique, we partitioned the code into data storage (*models*), APDU messages (*views*), and program logic (*controllers*). Due to the limitations imposed by the Java Card platform, it is not possible to divide the applet source code into packages. Therefore, all classes are in the main package. Furthermore,

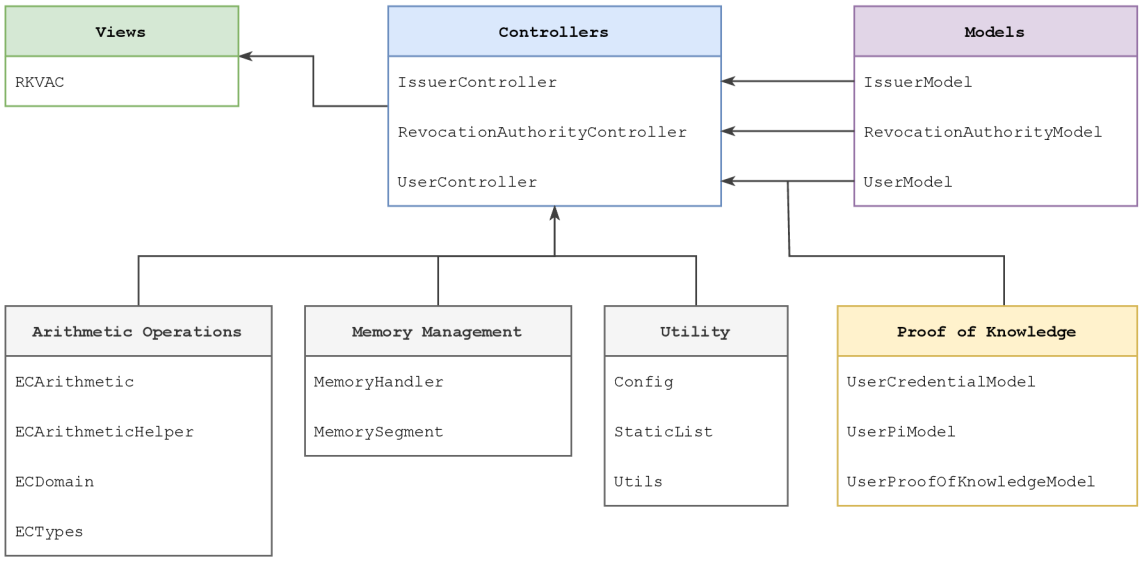


Fig. 4.1: Structure of the applet source code

we produced auxiliary code that does not adhere to the design pattern and is used to enhance the Java Card functionality. This code is separated into three parts: *(i)* arithmetic operations; *(ii)* memory management; and *(iii)* utility classes. Figure 4.1 graphically depicts this source code segregation.

The application's models play a crucial role in processing and storing the application data received from the controllers. To represent the system entities, the program uses three core models: the revocation authority, the issuer, and the user. Additionally, three auxiliary models separate the user data during the computation of the proof of knowledge. The controllers are responsible for directing the application logic and acting as a bridge between views and models. Each of the three core models has its own controller, and the user's controller also manages their auxiliary models. Finally, due to the smart card's lack of a graphical interface, the transmission and reception of data through APDU messages are considered views as they process the data input. Therefore, the application has only one view, which is the main class that drives the applet.

Regarding the auxiliary code, the arithmetic operations group comprises classes that implement modular and elliptic curve operations. We utilized the same elliptic curve as the original MULTOS-based smart card implementation. However, since the Java Card API does not officially support *Barreto-Naehrig* (BN) curves [53], we designed a class to provide domain parameters for BN254 curves. We also defined the data types required to represent elliptic curve information and how to operate with its points and scalar values. Finally, we implemented the arithmetic functions indicated in Section 4.4, with the aim of utilizing the coprocessor as much as possible. The memory management classes are responsible for managing smart card

resources. While performing mathematical operations in RAM could potentially yield higher performance than in EEPROM, RAM is limited, and managing it in a straightforward manner is essential. To this end, we partitioned the RAM into blocks called segments, each of which is 256 or 512 bytes in size and reserved when the applet is installed on the smart card. We also created a handler to temporarily lock these blocks to execute mathematical operations on them and speed up execution. Once the segment is no longer required, it can be unlocked for future use. Lastly, as the Java Card version is constrained and does not cover the entire language specification, we had to develop specialized utility classes and methods for dealing with lists, bit processing, and other functions. While these are offered as utility classes, they are crucial for the application's development.

4.5.2 Arithmetic operations

Basic algebraic operations, such as modular arithmetic or elliptic curve calculations, are not supported by the Java Card API. We set out to leverage the coprocessor as much as possible when implementing the operations required by the KVAC and RKVAC schemes.

Hardware-accelerated execution of arithmetic operations is available through the `BigNumber` class within the `javacardx.framework.math` package of the Java Card API. However, this package is not available on most smart cards as it is an optional one. If used on an unsupported smart card, the applet would be refused for loading or installation. To address this limitation, we combined the software implementation with various hardware algorithms.

Simple operations like addition, subtraction, negation, and division do not exert excessive strain on the CPU. They do not demand any type of hardware acceleration because the CPU can handle them. For this reason, we adopted the implementation of the `Bignat` library [54] and tweaked it to meet our requirements. On the other hand, multiplication and inversion are more costly operations; therefore, employing the coprocessor is critical to avoid unnecessarily slowing down the execution. We utilized RSA encryption with padding disabled (`MODE_ENCRYPT` and `ALG_RSA_NOPAD`) to carry out the computations depicted in Equation 4.1 and Equation 4.2. To compute the modular multiplication, we combined software execution with the utilization of the coprocessor. We used the RSA routine to compute the squares and the CPU to calculate the addition, subtraction, and division by two. Furthermore, to speed up the execution of multiplication, we may reuse the square computation (a^2 and b^2) when the operands are common, i.e., the same value is used in numerous multiplications. To calculate the modular inversion, we apply the transformation shown in Equation 4.2. Using the value $q - 2$ as the exponent in the RSA encryption

```
public static void ECAddition(ECPoint result, ECPoint point1, ECPoint point2) {
  ...

  keyPair = mh.getKeyPair(Config.ALG_EC_PACE_GM);

  pk = (ECPublicKey) keyPair.getPublic();
  sk = (ECPrivateKey) keyPair.getPrivate();

  pk.setW(point2.getPoint(), (short) 0, Config.ECP_SIZE);
  sk.setG(point1.getPoint(), (short) 0, Config.ECP_SIZE);

  keyA = mh.getKeyAgreement(Config.ALG_EC_PACE_GM, sk);
  keyA.generateSecret(point2.getPoint(), (short) 0, Config.ECP_SIZE, result.getPoint(), (short) 0);
}

public static void ECScalarMultiplication(ECPoint result, ECPoint point, ECFr operand) {
  ...

  keyPair = mh.getKeyPair(Config.ALG_EC_SVDP_DH_PLAIN_XY);

  pk = (ECPublicKey) keyPair.getPublic();
  sk = (ECPrivateKey) keyPair.getPrivate();

  pk.setW(point.getPoint(), (short) 0, Config.ECP_SIZE);
  sk.setS(operand.getFr(), (short) 0, Config.EC_SIZE);

  keyA = mh.getKeyAgreement(Config.ALG_EC_SVDP_DH_PLAIN_XY, sk);
  keyA.generateSecret(point.getPoint(), (short) 0, Config.ECP_SIZE, result.getPoint(), (short) 0);
}

```

Code 4.1: Implementation of elliptic curve operations.

routine, we obtain the inverse of a number. This $q-2$ value is common throughout the execution and can be precomputed. Note that q is the prime order of the EC base point. Finally, we evaluated utilizing RSA for the modular reduction, setting as the private key the value of 1. Nevertheless, we obtained the exception `CryptoException.ILLEGAL_USE` since, as stated in the documentation, the exception is thrown when the message value is greater than or equal to the modulus. Therefore, for modular reduction, we divided the value to be reduced by the order of the elliptic curve. The modular addition and subtraction operations carry out the standard operations before executing the modular reduction operation.

The implementation of elliptic curve operations presented a challenge since execution on the CPU would bring considerable overhead. Indeed, it was crucial to use hardware acceleration. We were able to leverage the coprocessor and perform such operations rather quickly by exploiting the techniques described in Equation 4.3 and Equation 4.4. However, the PACE GM algorithm and the version of ECDH that produces the plain `X` and `Y` coordinates are only accessible as of Java Card version 3.0.5. Fortunately, we observed that these features were still accessible from version

3.0.4, i.e., Java Card version 3.0.4 currently supports a subset of the features available in Java Card version 3.0.5. Thus, we rely on the class `KeyAgreement` of the Java Card API to perform point addition and scalar point multiplication operations. The definitions of version 3.0.5 for accessing the PACE GM and ECDH PLAIN XY protocols are `ALG_EC_PACE_GM` and `ALG_EC_SVDP_DH_PLAIN_XY`, respectively.

Code 4.1 presents the source code for both implementations, which enables us to construct KVAC and RKVAC schemes with convenient execution times.

4.5.3 Data exchange and message flow

We designed a life cycle for the protocol's execution. Figure 4.2 depicts the interaction between the four entities of the system, as well as the amount of data exchanged and the number of messages sent. Unlike the original version for MULTOS-based smart cards, we may transfer all the data in a single transmission thanks to the extended APDU feature of the T=1 communication protocol, which reduces the overhead. As part of the applet's installation process, we create instances of the three system models and controllers, i.e., user, revocation authority, and issuer. In addition, we instantiate the memory manager and allocate dynamic memory.

The user and the revocation authority entities exchange the first block of information, where the user transmits its identity (21 bytes) and the revocation authority returns the revocation information (1263 bytes). This information comprises not only the revocation handler and credential, as stated in the protocol description, but also the set of randomization pairs. We provide this additional information since it is necessary to store it on the smart card. We precompute the square of the revocation handler to accelerate the proof of knowledge calculation by eliminating one call to RSA.

The user and the issuer will have their next interaction. The issuer begins by sending a data block containing the attribute count (1 byte), followed by the attributes themselves (288 bytes). As with the revocation handler, upon receiving the attributes, we precompute the squares of all attributes. Next, the user submits to the issuer its unique identification, the revocation handler and credential, as well as the number and attributes contained on the card. This procedure is divided into two stages since the issuer initially has no idea how many attributes the user has access to. Thus, the two messages include the following information: 1) the user identification, revocation data, and the total number of attributes (119 bytes), and 2) the attributes (288 bytes). The issuer transmits back the user's credentials (715 bytes) in a single message.

Finally, the verifier transmits a nonce (32 bytes), an epoch (4 bytes), and the number of attributes to disclose to the user. We supply this value in the APDU

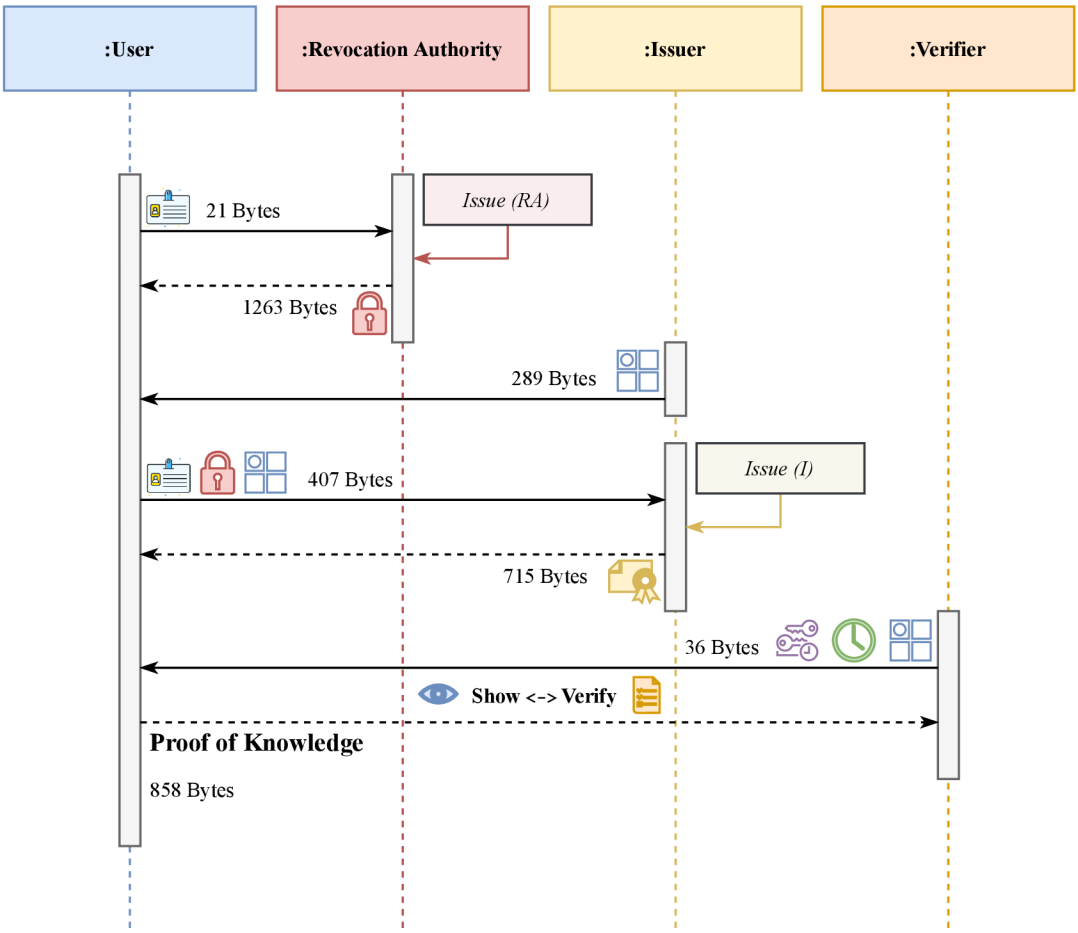


Fig. 4.2: Life cycle of the RKVAC scheme

packet's parameters, not in the data itself. The user employs the memory handler to conduct certain complex mathematical operations on RAM and the precomputed values of the attributes during the proof of knowledge calculation, which speeds up the execution performance. The user provides the verifier with the outcome of the proof of knowledge (858 bytes).

Both interactions between the revocation entity and the issuer occur once during the smart card's initialization process. The one with the verifier is self-contained and may be run several times to enable the user to authenticate anonymously.

4.5.4 Acceleration techniques

We propose an off-card precomputation approach to accelerate the execution of the `Show` algorithm, which is shown in Figure 3.5a at a high level. By precomputing roughly three of the four stated stages, i.e., selecting a unique per-session value, generating the transaction pseudonym, and randomizing the credential, we can minimize the execution times shown in Figure 4.4 by more than 50%. The `Show` algorithm

enables anonymous user authentication. However, for the same epoch value, the number of authentications is limited to 100. Table 4.2 illustrates the amount of storage space necessary to store each precomputation (i.e., $n = 1$), as well as the precomputations for the 100 authentications (i.e., $n = 100$).

Tab. 4.2: Space required for authentication precomputations

Stage	$n = 1$	$n = 100$
1) Select a unique per-session value	32 bytes	3.2 KB
2) Generate the transaction pseudonym	65 bytes	6.5 KB
3) Randomize the credential	517 bytes	51.7 KB
4) Compute the proof of knowledge (*partial*)	325 bytes	32.5 KB
Total	939 bytes	93.9 KB

We excluded the part where we compute the proof of knowledge for undisclosed attributes from the precomputation. As with the nonce, the attributes to disclose are data sent by the verifier and not previously known, i.e., session data.

Considering the available data space on the smart card used for implementation, it is conceivable to store the 100 precomputations. Nevertheless, it would be possible to include the part where we compute the proof of knowledge for undisclosed attributes, calculating the five alternative states of discovery, i.e., without disclosing attributes, disclosing one, disclosing two, and so forth. Due to a shortage of space, we merely implemented it for testing and benchmarking.

4.6 Experimental results

To ascertain the feasibility of our approach, we conducted several experiments, measuring and comparing the execution speed with that of the MULTOS-based smart card implementation. Figure 4.3 and Figure 4.4 illustrate the performance of the KVAC and RKVAC protocols using MULTOS and Java Card technologies, respectively. Both figures depict the benchmarks in milliseconds and include both computation time and communication overhead. The outcomes of the MULTOS-based smart card implementation stem from the research conducted by Camenisch et al. [9] and Chapter 3, which draws upon the work of Hajny et al. [11].

We can observe that the Java Card implementation of the KVAC protocol takes approximately the same amount of time to disclose all of its attributes as the MULTOS implementation does without disclosure. The protocol takes roughly one second longer to run for each attribute that is not disclosed. The complexity of the

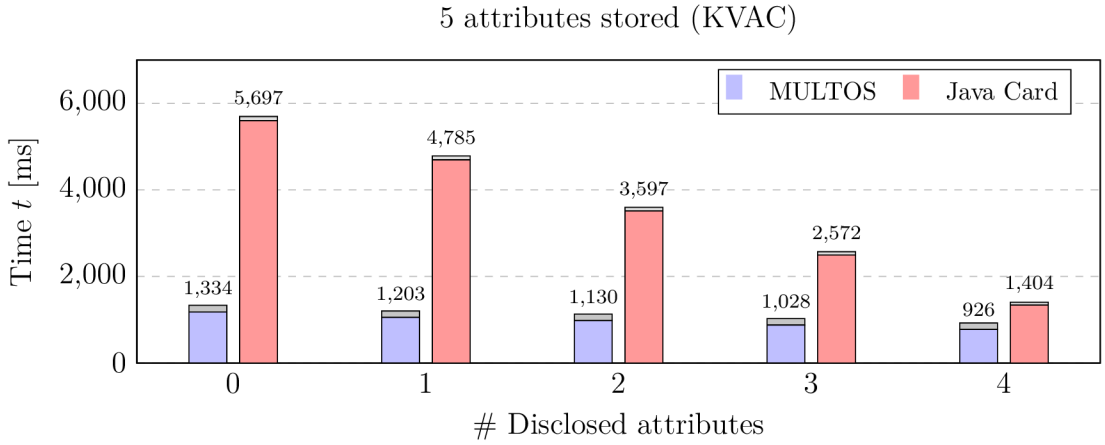


Fig. 4.3: Speed comparison of KVAC between MULTOS and Java Card implementations

RKVAC protocol is mirrored in the execution speed of the Java Card implementation, which is close to five times slower than the MULTOS implementation. Additionally, likewise with the KVAC protocol, the execution time is increased by around one second for each undisclosed attribute.

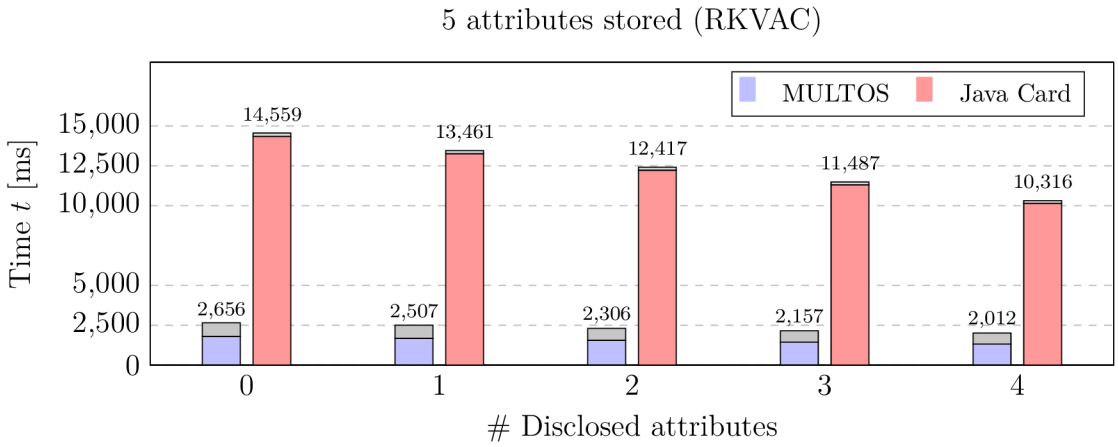


Fig. 4.4: Speed comparison of RKVAC between MULTOS and Java Card implementations

However, we can appreciate that the communication overhead in Java Card is significantly inferior to that of MULTOS. We achieve this gain by using the T=1 communication protocol, which enables us to transmit all the data in a single message rather than split it into several packets.

We applied the acceleration techniques described in Subsection 4.5.4, obtaining the results depicted in Figure 4.5. Both acceleration techniques fully precalculate the first three phases of the `Show` protocol (see Table 4.2). Concerning the fourth phase,

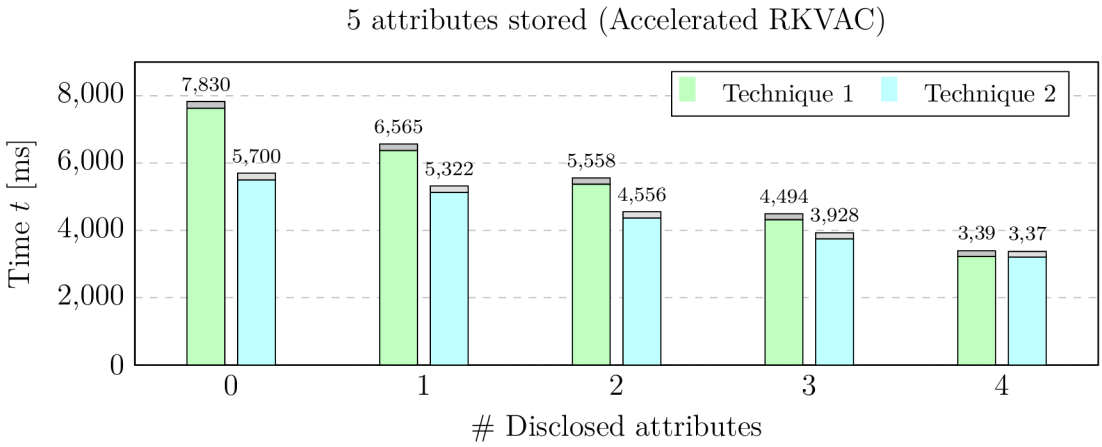


Fig. 4.5: Speed comparison between acceleration techniques for RKVAC

the first technique precomputes the values $t_{revoke}, t_{sig}, t_{sigI}, t_{sigII}$ and partially t_{verify}, since the number of attributes to be disclosed is unknown. Alternatively, the second technique precomputes the five possible values of t_{verify}, i.e., when no attributes are disclosed, when one attribute is disclosed, and so forth, and employs the one that is necessary at the time. Through the use of these acceleration techniques, we detected a remarkable decrease in the execution duration of the protocol.

Note that the issuer (i.e., an external device such as a Raspberry Pi) performs the authentication precalculations. This information is stored on the smart card during the smart card personalization phase for use in the subsequent authentication process (i.e., the `Show` algorithm). We did not include the time necessary to precompute the 100 authentications since it is negligible, requiring less than a half-second for both acceleration techniques.

The execution times achieved are still far from those we found with the MULTOS-based smart cards. Evaluating both acceleration techniques, we can glean that the gain obtained by precomputing the attribute disclosure is impractical if we consider the memory space it requires. Moreover, due to space limitations, it is not possible to store all 100 precomputations, including attribute disclosure, on the smart card. We can state that the optimal configuration is the first acceleration technique.

4.7 Summary

This chapter is dedicated to the implementation of the KVAC and RKVAC anonymous credential schemes for Java-based smart cards and provides a comparison of our approach with the MULTOS implementation. We have leveraged algebraic derivations and abused the limited Java Card API to perform certain computations

on the coprocessor. Additionally, we have applied various optimization and acceleration techniques to further enhance the protocol's performance. In fact, our off-card precomputation approach allows us to speed up the `Show` algorithm by more than 50%. Unfortunately, all of these techniques, as seen in the figures, are insufficient to achieve the efficiency and speed of MULTOS-based cards.

It is clearly impossible to compete with hardware acceleration using a software implementation. Nonetheless, we have obtained encouraging results with a platform as limited as Java Card in terms of modular operations and elliptic curve operations. Considering the difference in data memory available on the Java Card, employing our acceleration techniques but performing all fundamental modular and elliptic curve operations with hardware acceleration would yield much better results.

The contents of this chapter are based on the publication by Casanova-Marqués et al. [12].

5 Privacy-enhancing authentication system

In the previous chapter, we presented a comprehensive demonstration of how to implement an ABC scheme using Java Card technology, given the challenges of procuring MULTOS smart cards. Nonetheless, the use of smart cards poses some functional challenges, including the lack of visibility regarding the attributes requested and the absence of an option to accept or decline their disclosure. In this chapter, we introduce the PEAS, a robust solution that is fully prepared for deployment in real-world scenarios. PEAS is compatible with smart cards and works seamlessly with smartphones and smartwatches. The chapter is organized as follows:

- An introduction to ABC technology, including its evolution and readiness, emerging trends and future prospects, as well as practical applications and deployment.
- An explanation of the system architecture and technical aspects.
- Use-case scenarios and a pilot deployment demonstrating the practical applications of the proposed system.
- Implementation details and benchmark results.
- A summary of the key findings.

5.1 Introduction

In an increasingly digitalized era, the imperative for secure and privacy-preserving authentication mechanisms has gained paramount significance. Anonymous credential schemes have emerged as a promising solution, allowing individuals to validate their personal attributes while preserving their anonymity. However, the conventional implementation of ABC schemes on smart cards presents functional challenges that curtail their efficacy in real-world scenarios.

One notable challenge lies in the limited visibility afforded to users regarding the attributes requested during authentication processes. With smart cards lacking user interfaces, individuals face difficulties verifying which attributes are being disclosed. This opacity not only gives rise to privacy concerns but also restricts control over personal information. Moreover, the absence of an option to accept or decline the disclosure of specific attributes further impedes user acceptance and adoption of ABC technology on smart cards.

To surmount these limitations and enhance the user experience, there is a growing interest in exploring alternative platforms for ABC implementations, such as smartphones and smartwatches. By harnessing the advanced capabilities and intuitive interfaces of these devices, users can enjoy heightened visibility and control over attribute disclosure. Shifting the implementation focus toward these widely

embraced personal devices opens the door to creating authentication experiences that are both user-friendly and privacy-preserving. Consequently, this endeavor fosters greater acceptance and adoption of ABC technology across diverse real-world scenarios.

5.2 State of the art

Several ABC schemes have been proposed to enable pseudonymous authentication of users by service providers using attributes. These include the works by Brands [5], Verheul [55], Camenisch and Lysyanskaya [6], Persiano and Visconti [56], Chase et al. [57], and Hajny et al. [48]. In addition to theoretical proposals, various privacy-preserving systems based on ABCs have been implemented. These range from proof-of-concept, such as *I Prove Possession of Attributes* (I2PA) [58], to open-source implementations and pilots, such as IBM's Idemix [59] and Microsoft's U-Prove [60], which provide various functionalities. Recent studies have also presented practical and efficient smart card implementations of ABCs [9].

Numerous ongoing projects aim to develop ABC systems for smartphones and wearables. For example, van den Broek et al. [61] proposed a scheme for obtaining credentials on smartphones from e-ID documents using IRMA ABC technology based on Idemix's cryptographic core. Alpár et al. [62] presented the IRMA solution as a privacy-friendly identity management solution using smartphones and demonstrated how IRMA applications for users, service providers, and issuers can interact with each other. In addition, they discussed Idemix's future challenges. Sene et al. [58] compared Idemix, U-Prove, and I2PA and presented their implementation results on the Android and Raspberry Pi platforms. Furthermore, Papaioannou et al. [63] introduced a general privacy-preserving user authentication mechanism using pseudonyms, with an implementation tested on smartphones, that aims to maintain privacy in Smart City services connected by 5G. Although their work presented a demo setup of a local implementation with the Android emulator, further results on real devices or additional details were not provided.

5.3 Attribute-based credential technology

This section presents an overview of ABCs technology, discussing its key schemes, projects, and research challenges.

5.3.1 Evolution and readiness

Numerous research articles have been published on ABC technology, including notable works by Brands [5], Verheul [55], Camenisch and Lysyanskaya [6], Chase et al. [57], Hajny et al. [48], and Ringers et al. [64].

The concept of a credential system without direct identification was first introduced by Chaum in 1985 using the RSA cryptosystem [4]. Brands subsequently developed an innovative protocol in 2000 that allowed users to selectively disclose a set of attributes from their credentials while maintaining their privacy [5]. This protocol establishes the foundation of Microsoft's U-Prove technology [60], a user-centric cryptographic system that facilitates the issuance and presentation of cryptographically protected statements. U-Prove tokens, which encode user attributes, can be either on-demand (one-time) or long-lived (reusable with an expiration time). Microsoft has released two implementations of U-Prove technology: the U-Prove C# SDK and the U-Prove Extensions SDK, which includes extensions to the U-Prove Cryptographic Specification, in 2014. Further information on U-Prove technology is available in [65].

In 2001, Camenisch and Lysyanskaya [6] presented advanced multi-show credentials, which allow users to prove possession of their attributes unlinkably as many times as they wish while also providing optional anonymity revocation for malicious users. This protocol constitutes the basis of Idemix [59], an anonymous credential system developed at IBM Research and released in 2007. The Idemix system enables the issuer to sign the user's attributes to construct a cryptographic credential within the issue protocol, while the user randomizes and sends the credential to a verifier, anonymously proving the possession of attributes using zero-knowledge protocols. Further information on Idemix technology is available on IBM's website[1]. The Idemix implementation is also available on the GitHub repository[2].

ABCs have reached a mature stage of development and are now ready for deployment in modern *Information and Communication Technology* (ICT) systems. Ongoing pilot projects, such as the IRMA smart card and mobile application products for privacy-friendly authentication, demonstrate the viability of ABCs. Contemporary ABC schemes are also efficient enough to run on IoT devices, as exemplified by the anonymous scheme presented in the article by Camenisch et al. [9]. This scheme executes the show algorithm of the aforementioned protocol in under 500 ms (in the case of three stored attributes) on contemporary smart cards. Authentication and identification systems in the *United States* (US) and the *European Union* (EU) have adopted ABC technology because it is essential. The revocation issue that had

[1] `https://www.zurich.ibm.com/idemix`

[2] `https://github.com/p2abcengine/p2abcengine`

plagued ABC technology has recently been resolved, as demonstrated in the article by Camenisch et al. [3].

5.3.2 Emerging trends and future prospects

One of the prevailing tendencies in the field is the pursuit of *Post-Quantum Attribute-based Credential* (PQABC) schemes. PQABCs are often derived from *Post-Quantum Group Signature* (PQGS) primitives or *Attribute-based Signature* (ABS) schemes. In 2012, Camenisch et al. [66] presented lattice-based constructions for *Anonymous Attribute* (AA) tokens, whereby users employ attribute-containing credentials that divulge only a portion of their attributes. Later, in 2018, Boschini et al. [67] introduced a lattice-based AA token scheme featuring short zero-knowledge proofs. AA tokens from lattices have a size of 17.77 MB. Additionally, in 2019, Yang et al. [68] proposed lattice-based zero-knowledge arguments with standard soundness and privacy-preserving methodologies based on lattices.

Recently, research has also focused on decentralization and the incorporation of blockchain technology into ABCs. For example, Sonnino et al. [69] presented Coconut in 2018, a scheme that supports distributed threshold issuance, public and private attributes, re-randomization, and multiple unlinkable selective attribute revelations. Coconut can interface with blockchains by employing a smart contract library for Chainspace and Ethereum. Moreover, in 2020, Singh et al. [70] suggested a user-centric and privacy-preserving scheme with self-blindable credentials that could be verified on the blockchain.

To enhance the resilience of ABCs to quantum cryptanalysis, developing *Post-Quantum* (PQ)-based constructions is essential. Integrating decentralization into ABCs could also reduce reliance on *Trusted Third Parties* (TTP), making it an important area for further research in this field.

5.3.3 Practical applications and deployment

ABC schemes have many practical applications and scenarios where one attribute or a combination of attributes can be utilized.

- *Access control system*: a user can request access to secure areas, such as an office or laboratory, by proving their status as an employee, student[3], professor, or other authorized role [48, 11].
- *Club membership*: a user can provide proof of their membership and valid payment for membership fees [59].

[3] `https://privacybydesign.foundation/demo-en/student/`

- *Driving, renting, or sharing a car*: a user can rent or drive a car or use a car-sharing service by demonstrating that they possess a valid driving license in the appropriate category [62].
- *Electronic identification*: A user with a government-issued electronic ID can prove their attributes, like age range and EU citizenship, to authorized officers in the EU [62].
- *Legal restriction*: A user can demonstrate that they are over the age of 18 or 21 without revealing their date of birth[4].
- *Parking*: a user can prove their membership in a parking zone or lot, as well as their valid payment for parking, and park their car in the designated parking zone [71, 72].
- *Public transport*: a user can apply for a discount on public transport by proving that they are eligible, such as being a child, student, or senior, while also having a valid ticket.
- *Smart health*: ABCs can offer several advantages in e-healthcare services [73]. For example, a user can use attributes like reduced mobility or vision problems to verify their agility level.
- *Vaccination certificate*: A user can present a vaccination certificate when traveling internationally to provide a record of their vaccinations while also maintaining privacy. This allows everyone to demonstrate their vaccination status without revealing their identity.
- *Vehicular communication*: As shown by Neven et al. [74] in the context of *Cooperative Intelligent Transport System* (C-ITS), a user can use ABCs to improve privacy in vehicular communication. de Fuentes et al. [75] also explored the use of ABC techniques, such as Idemix, in various C-ITS use cases. These techniques allow for various attributes to be utilized as privacy-preserving tokens for spreading notifications about road conditions, approaching emergency vehicles, vehicle ownership, car insurance and financial services, fleet membership, and more.

5.4 System architecture and technical aspects

This section elucidates the technical aspects of our PEAS, which we have developed to enable secure and anonymous access to both electronic services and physically protected areas. By leveraging our technology, users can preserve their anonymity without disclosing their complete identity or becoming subject to tracking or profiling by system administrators. Nevertheless, in the event of malicious activities, the

[4] `https://privacybydesign.foundation/demo-en/18plus/`

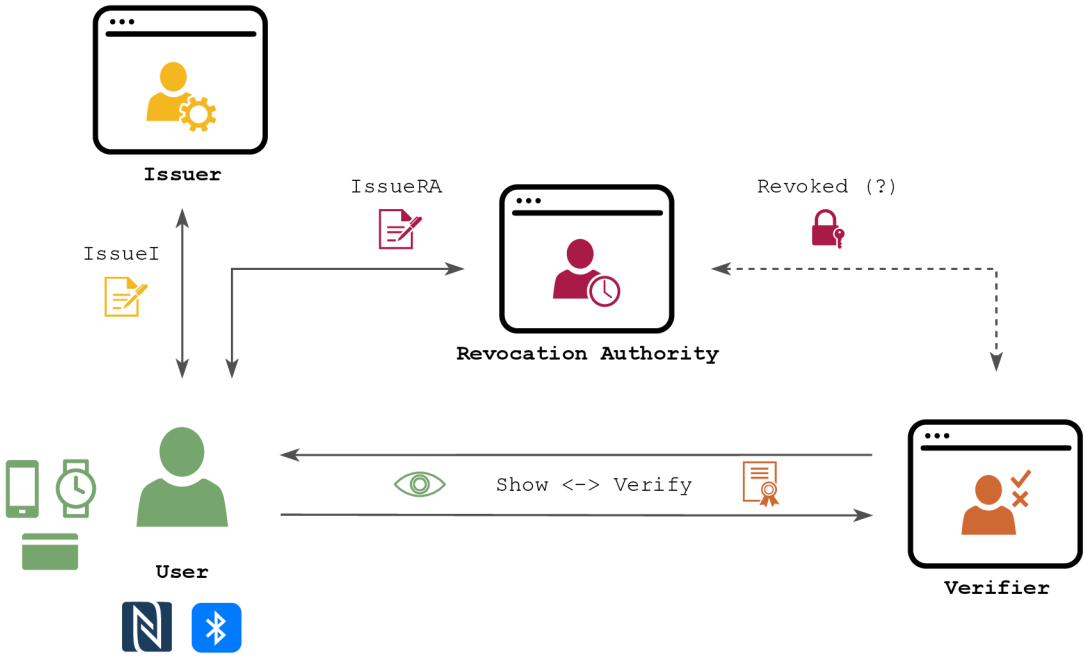


Fig. 5.1: High-level topology of PEAS

revocation authority plays a vital role in detecting and identifying any rogue users. By collaborating with the revocation authority, PEAS guarantees that, although user identities are anonymous, any illegal activity can still be traced back to the perpetrator.

PEAS represents user identity through multiple attributes, including but not limited to name, surname, age, gender, and job title. During the verification process, only necessary identity attributes, such as age, gender, and job title, are disclosed, ensuring that the user's privacy is safeguarded. Furthermore, the authentication sessions of PEAS are mutually unlinkable, guaranteeing the non-profiling and non-tracking of users.

The system is built upon the RKVAC technology proposed by Hajny et al. [11] and presented in Chapter 3, which serves as the cryptographic core of the system. Likewise, we optimized and expanded the RKVAC technology for deployment in real-world scenarios. PEAS is highly practical and adaptable to various platforms, including smart cards, smartphones, smartwatches, and single-board computers, commonly employed in IoT environments. Moreover, PEAS adheres to a user-centric approach, where users have complete authority over their personal data, deciding which information to provide and when to offer it to service providers. This approach ensures that irrelevant personal data is not stored, aligning with the *General Data Protection Regulation* (GDPR).

Our implementation of PEAS comprises software applications for all entities involved in the authentication process, namely the revocation authority, the issuer

of attributes, the verifier of attributes, and user devices holding personal attributes. Figure 5.1 provides a detailed illustration of the PEAS architecture.

5.5 Use cases for the proposed system

This section presents some use cases for the Privacy-Enhancing Authentication System.

PEAS is a versatile system with multiple possible use cases that ensure the privacy and security of its users. It can be employed in a variety of scenarios, including public transport, car rental and sharing services, access control systems, club memberships, parking, and age verification. In public transport, for instance, PEAS allows users to prove their subscription attributes without disclosing their identity. Similarly, in access control systems, users can provide proof of their work position or role without compromising their privacy. PEAS can be used to support general membership attributes for website clubs and to verify parking permits and payments.

The system boasts a remarkable degree of versatility and capability, meaning it can be used in a wide range of applications, as detailed in Subsection 5.3.3. Its adaptability allows it to provide reliable and efficient solutions across various industries while also catering to the unique requirements of any given scenario. Its extensive range of applications not only highlights its effectiveness but also its potential to overcome complex challenges and drive innovation in various sectors, thereby cementing its position as a valuable privacy tool for organizations. Additionally, the system can be expanded for use in C-ITS, smart healthcare, and e-ID integration. However, such expansions necessitate careful integration and adherence to relevant standards, such as *ETSI TS 103 097 V1.3.1* and *ETSI TS 102 941 V1.3.1* standards in C-ITS, among others.

5.5.1 Pilot deployment

We conducted a pilot program to evaluate the effectiveness of PEAS in an access control system scenario. Our university campus provided the ideal testing ground, where we installed a PEAS access terminal with both contact and contactless smart card readers to regulate access to restricted rooms. Please see Figure 5.2 for a visual representation of the pilot PEAS deployment. During the pilot, university employees and students used their personalized user devices, such as smart cards and smartphones, to gain access.

Fig. 5.2: Pilot of PEAS technology on the university campus

5.6 Implementation details

This section describes the development process and summarizes the main key points considered during the design of the system. Table 5.1 outlines the hardware and software specifications of the devices that we used. We utilized personal devices that were equipped with PC/SC and Bluetooth interfaces to establish communication with the system. Specifically, the smart card interfaced with the system through the PC/SC interface in contact mode, whereas the smartphone supported both contactless PC/SC (i.e., *Near-Field Communication* (NFC)) and Bluetooth connectivity. In contrast, the smartwatch exclusively relied on Bluetooth for communication with the system.

We divided the implementation of the application into three components: *(i)* the *libpeas* library for core functionality; *(ii)* the *server-side* to operate with the revocation authority, the issuer, and the verifier; and *(iii)* the *client-side* for user functionality.

Tab. 5.1: Hardware and software specifications of the devices

Device	MCU / CPU	OS	RAM
Smart cards			
MULTOS ML4	SC23Z018	MULTOS 4.3.1	1.28 KB
Smartphones			
Samsung Galaxy S21+ 5G	Exynos 2100	Android 11	8 GB
Smartwatches			
Galaxy Watch 4 Classic	Exynos W920	Wear OS 3.2	1.5 GB

5.6.1 Core library

At the core of our application is the purpose-built *libpeas* library, which provides a comprehensive implementation of the protocol. This library offers a suite of highly optimized cryptographic functions, including user revocation, authentication and verification, and user attribute issuance. It also includes communication and storage routines that enable efficient data transmission over a variety of interfaces, such as PC/SC, Bluetooth, and *Transmission Control Protocol* (TCP), as well as secure and private database storage. Designed from scratch to achieve unparalleled performance and security across a wide range of devices, *libpeas* is written in the C programming language and relies on several third-party libraries. Specifically, we utilized `mcl` [33] for elliptic curve cryptography; `openssl`[35] for cryptographic hash algorithm support; `pcsc-lite` [76], `ccid` [77], and `bluetooth` [78] for device communications; and `libcjson` [79] and `libwebsockets` [80] for web server connections.

The protocol was designed and implemented using elliptic curve cryptography. Hajny et al. [81] conducted a thorough evaluation of various cryptographic libraries, including *Pairing-based Cryptography* (PBC) [82], *Multiprecision Integer and Rational Arithmetic Cryptographic Library* (MIRACL) [83], *University of Tsukuba Elliptic Curve and Pairing Library* (TEPLA) [84], *Efficient LIbrary for Cryptography* (RELIC) [85], and `mcl` [33]. Their evaluation found that `mcl` provides the most promising performance results. Additionally, it offers the highest level of security due to its support for BN [53] pairing-friendly elliptic curves. Consequently, we utilized the BN254 curve supplied by the `mcl` library for our implementation.

5.6.2 Server-side specifics

The server implementation features a highly modular design that enables effortless maintenance and expansion of the system's various components, including separate layers and custom features. It comprises a front-end and a back-end, with the

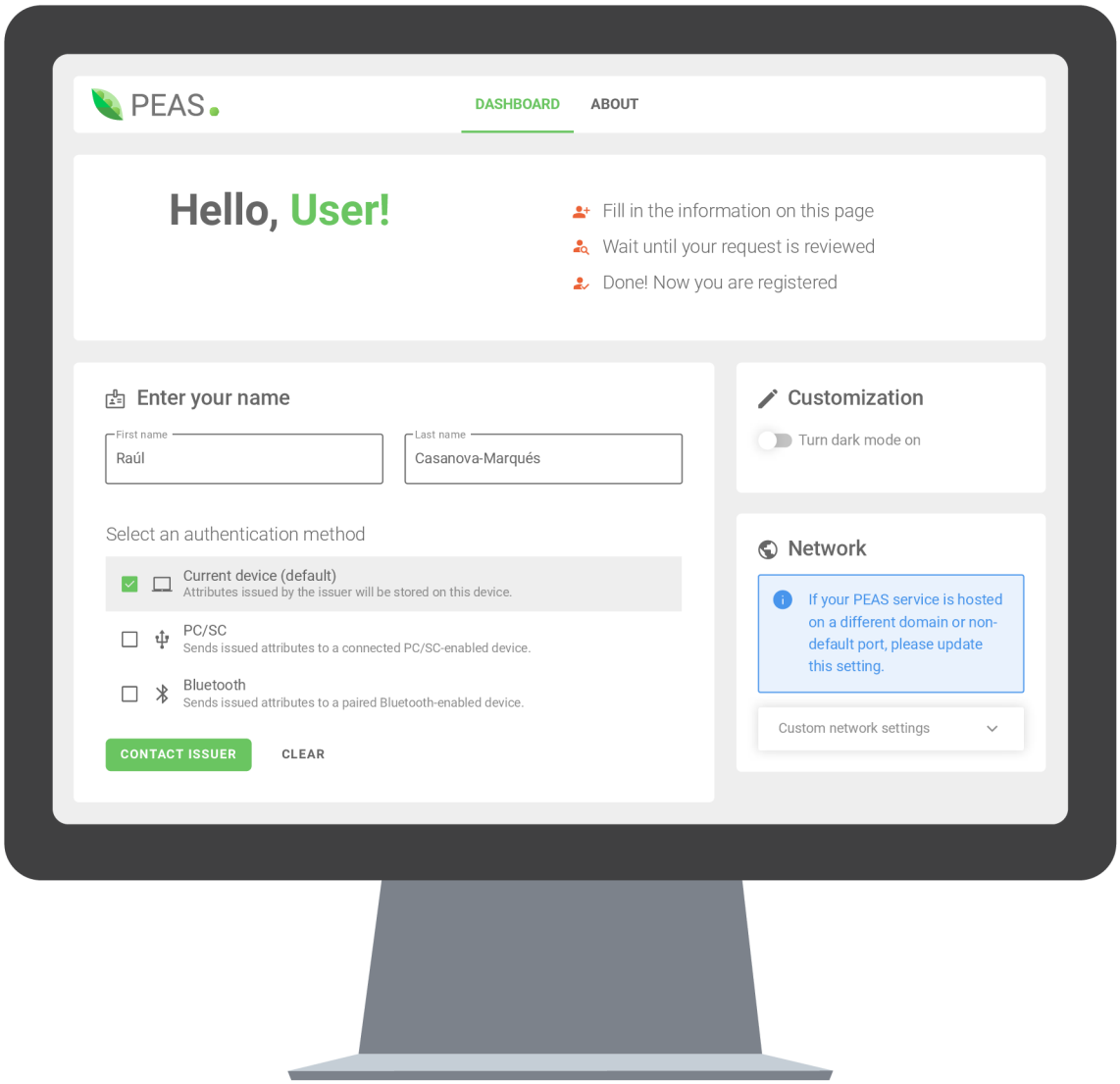


Fig. 5.3: Dashboard of the user's web application

front-end serving as a user-friendly graphical interface built on cutting-edge web technologies such as `Node.js` [86] and `Vue.js` [87]. The *Graphical User Interface* (GUI) components are set up using the `Vuetify` library [88]. The front-end offers four distinct applications for each of the system entities: revocation authority, issuer, user, and verifier. Each entity's interface provides specific functionality, such as the revocation authority's ability to aggregate the logs of revoked users to enhance system security. The issuer can emit personal attributes, manage user accounts, and revoke them. Users can send registration and authentication requests and manage their device and network preferences. The verifier, on the other hand, can grant access to users, select necessary attributes, and set epochs.

Figures 5.3 and 5.4 depict the user and verifier web application dashboards, respectively. The user is endowed with the ability to choose from multiple authenti-

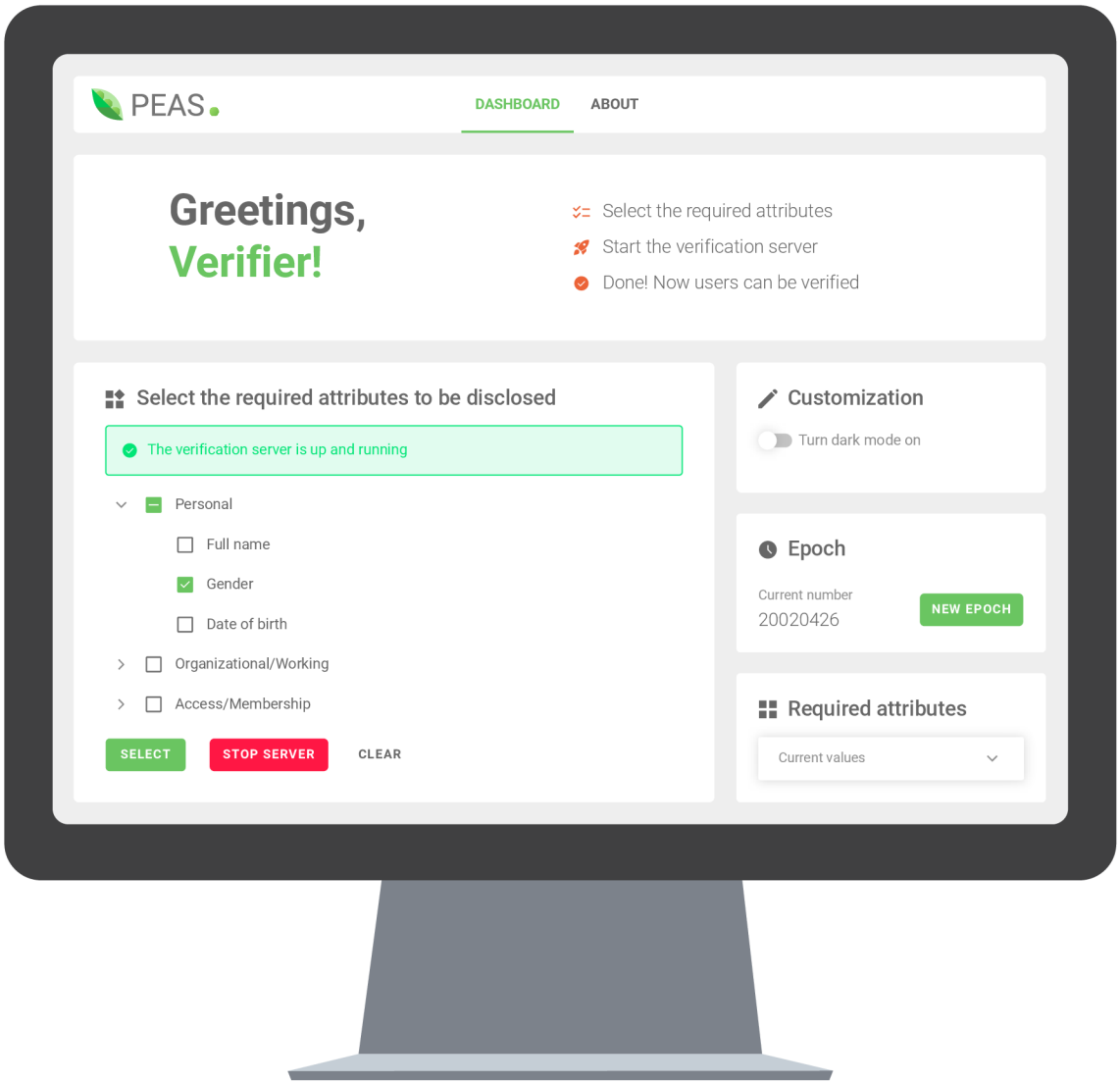


Fig. 5.4: Dashboard of the verifier's web application

cation methods that encapsulate their personal attributes. These methods comprise the current device that hosts the web server, typically a personal computer, a smart card, or a smartphone that necessitates an integrated or plugged-in smart card reader utilizing the PC/SC interface. Furthermore, the user may employ a smartwatch or smartphone that mandates an integrated or plugged-in Bluetooth adapter. On the other hand, the verifier has the prerogative to select the pertinent personal attribute that necessitates disclosure, modify the current epoch, which signifies the verification time frame, and scrutinize the user's access history.

Our backend component leverages the *libpeas* library to support cryptographic operations based on the RKVAC. In addition, it offers robust system management, secure data storage, and seamless communication with all entities in the system.

To enable communication between the frontend and backend components, as

well as other system entities, we employ various interfaces, including the *REpresentational State Transfer* (REST) API, *Command Line Interface* (CLI), TCP socket, and WebSocket. The REST API facilitates web communication with the server through the dashboard of multiple entities and is triggered when the web GUI initiates communication. The CLI allows for easy control of the PEAS core by submitting commands directly from the console. For network communication among entities such as the user, issuer, verifier, and revocation authority, we deploy a TCP socket. This communication is secured by *Transport Layer Security* (TLS) and establishes connections between microservices running each entity. Finally, for local communication between the web and the server, we leverage a WebSocket. This technique is used when the web server initiates the communication, enabling PEAS core control through the GUI instead of terminal commands. Lastly, we used `Docker` for straightforward deployment.

5.6.3 Client-side specifics

The implementation of PEAS for user devices involves the deployment of two distinct applications, namely the MULTOS application for smart cards and the Android application for smartphones and smartwatches. These applications serve as the user interface and are designed to be fully interoperable with each other and other PEAS entities, including the revocation authority, issuer, and verifier. Each of the applications is capable of storing up to nine personal attributes that can be repeatedly issued to the user. The tenth attribute is reserved for the revocation attribute issued by the revocation authority to identify and revoke malicious users. The cryptographic operations required for attribute verification are performed by the applications themselves.

In the case of the smart card application, we utilized the MULTOS ML4 smart card, which was programmed using MULTOS assembly code and the MULTOS API. We opted for this smart card due to its hardware acceleration for modular arithmetic and elliptic curve operations using the coprocessor, which significantly improves performance efficiency [34]. However, this card supports only the T=0 transmission protocol and has a limited RAM, which can adversely impact its effectiveness.

On the other hand, the Android applications leverage the latest trends in smartphone and smartwatch technology to provide users with an alternative to smart cards. We implemented the applications using the Java programming language and C functions through the Android *Native Development Kit* (NDK) to achieve optimal performance results. The applications require at least SDK version 24 (Android 7.0 and higher) and use the `mcl` library, along with standard Android and Java libraries.

To ensure the security of personal access to the Android application, we provide a

4-digit *Personal Identification Number* (PIN) code. Additionally, users have the option to activate the utilization of their fingerprint by enabling the designated setting indicated by the yellow arrow in Figure 5.5. Users can actively disable the issuance of attributes for security reasons, as denoted by the red arrow. Furthermore, it is important to highlight that the application cannot undergo repeated re-personalization without a reset of all settings and personalized data. The Android applications also maintain a history of events (i.e., logs) and offer the option to reset device settings, as indicated by the blue arrow. Communication with other system entities is facilitated using NFC and Bluetooth technologies. Users can attach their smartphones to an NFC reader or select a paired terminal from a drop-down list and confirm it by clicking on the `Connect` button if they intend to use Bluetooth. Figure 5.5 illustrates the graphical activities of the PEAS Android-based application for smartphones.

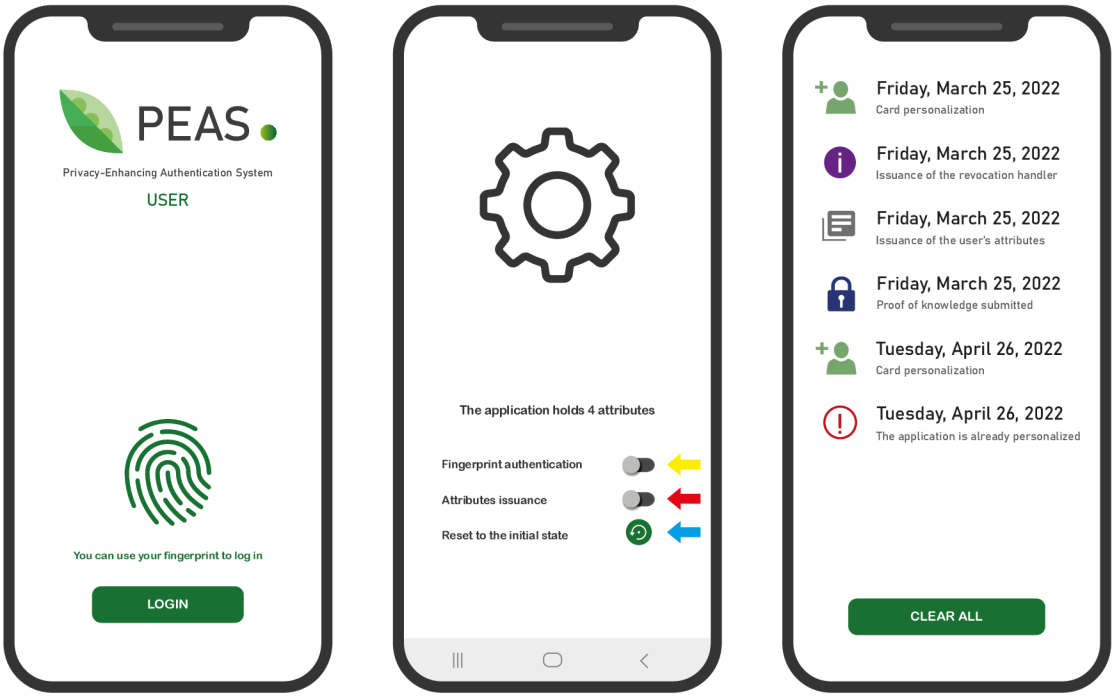


Fig. 5.5: Android application of PEAS for smartphones

Figure 5.6 illustrates the graphical interface of the PEAS Android-based application designed for smartwatches. The application shares fundamental similarities with its smartphone counterpart, but with one significant distinction: it lacks a login dialog consisting of a PIN code and fingerprint to enhance user-friendliness and convenience when accessing the application on the go.

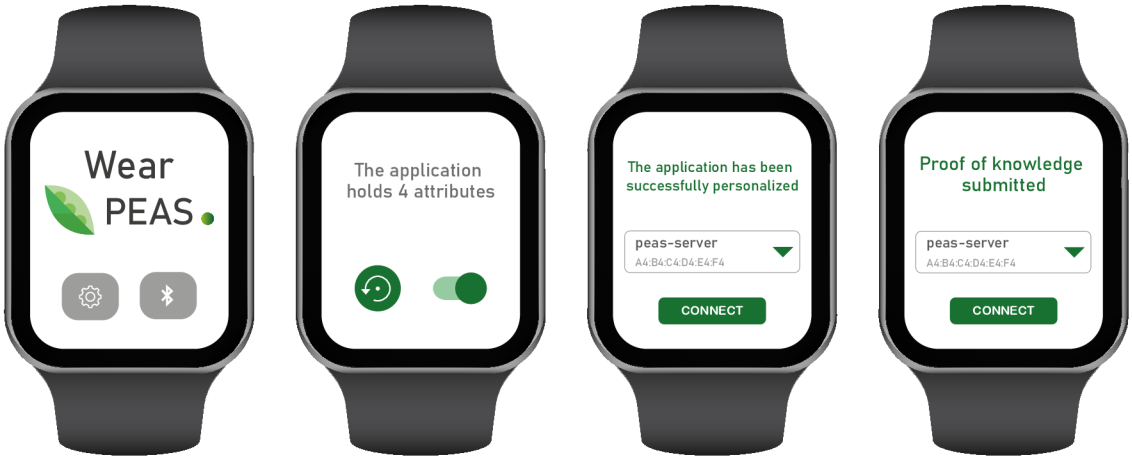


Fig. 5.6: Android application of PEAS for smartwatches

5.7 Experimental results

This section presents the benchmarks of our system implementation, which are exclusively centered on the `Show` and `Verify` algorithms. The findings are organized according to the communication interface deployed. Figure 5.7 depicts the benchmarks for PC/SC-enabled devices, while Figure 5.8 illustrates the benchmarks for Bluetooth-enabled devices.

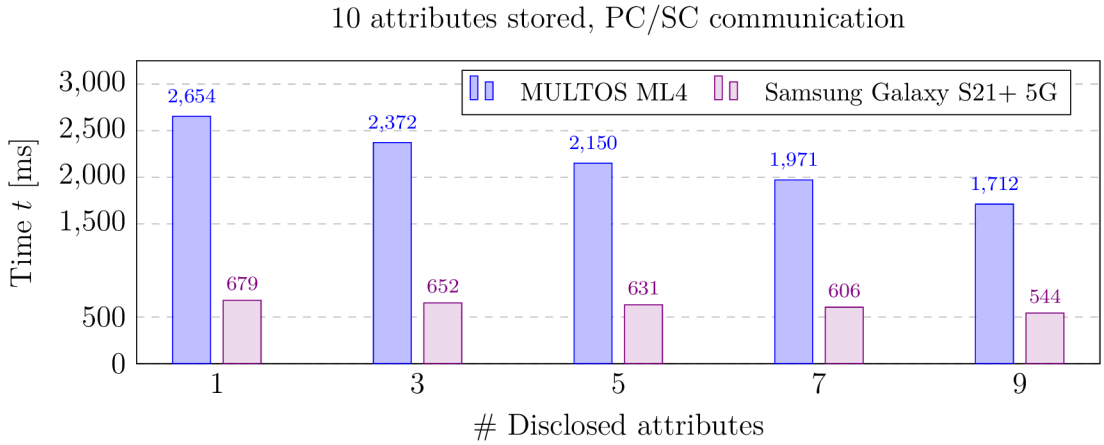


Fig. 5.7: Speed comparison of PEAS execution (client-side) on PC/SC-enabled devices

In each scenario, we assume the cryptographic credential stores 10 attributes, of which 1, 3, 5, 7, and 9 are disclosed while the remaining attributes remain concealed. To prove knowledge of each hidden attribute, the user's device must compute a corresponding proof. Our benchmarks consider both compute and transmission latency and are reported in milliseconds. The entities involved in the tests were the

issuer, verifier, and revocation authority, run on a conventional personal computer, and the user entity, represented by a smart card, smartphone, and smartwatch.

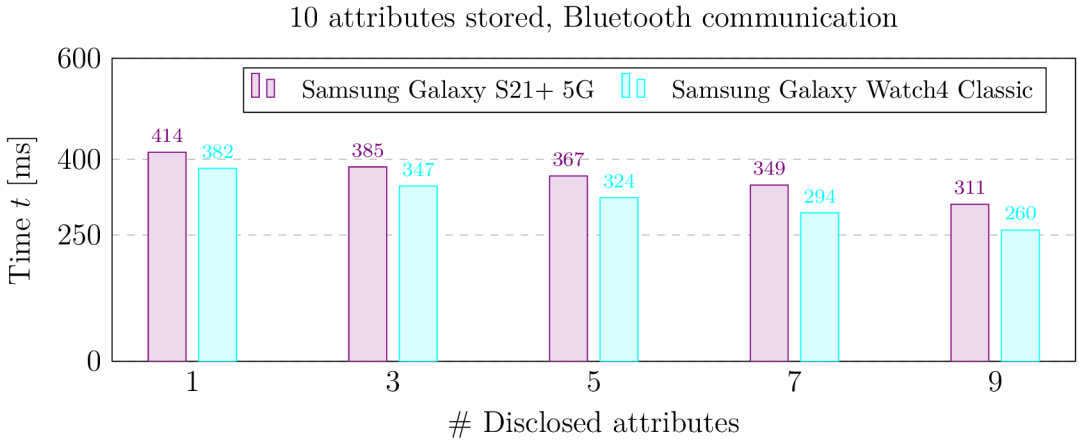


Fig. 5.8: Speed comparison of PEAS execution (client-side) on Bluetooth-enabled devices

After comparing the outcomes depicted in Figure 5.7, it is apparent that the smart card operates at a rate three to four times slower than the smartphone. Additionally, Figure 5.8 displays the comparable outcomes obtained from the smartphone and smartwatch. A comparison of the results obtained from the smartphone with PC/SC and Bluetooth communication interfaces reveals that Bluetooth technology is the superior alternative, as it reduces communication overhead and proves to be the faster choice.

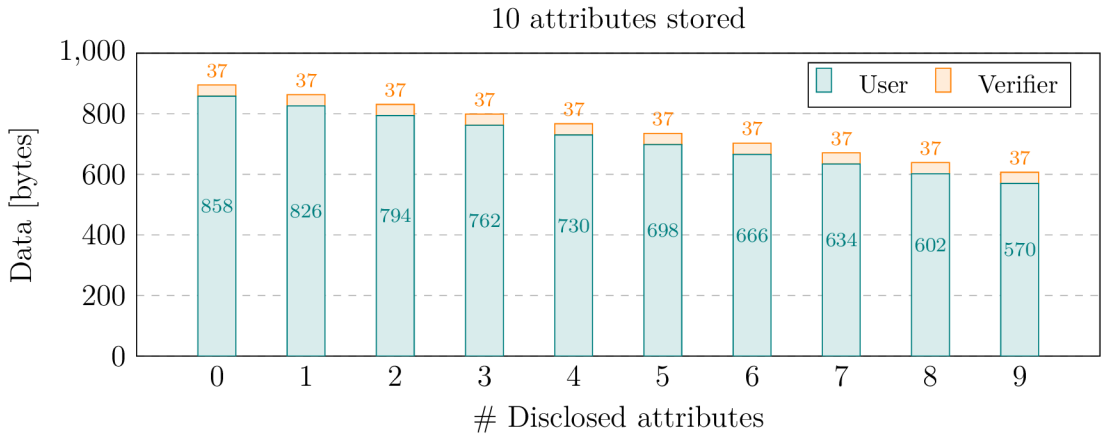


Fig. 5.9: Total amount of transferred data during the authentication phase

Figure 5.9 presents the aggregate amount of data that must be transmitted during the authentication phase between the user and verifier. We consider all possible scenarios where the cryptographic credential stores ten attributes and the disclosure

process starts from none to all nine attributes. The tenth attribute, which is the revocation attribute and is confidential to the user, is never disclosed. It is worth noting that the amount of data transmitted by the user is significantly greater than the amount transmitted by the verifier. The verifier only sends the authentication challenge, epoch identifier, and 1 byte of data that defines the disclosed attributes. In contrast, the user sends at least 570 bytes (all attributes are disclosed) and up to 858 bytes (all attributes are concealed). It is important to note that the PEAS implementation does not transmit disclosed attributes from the user's device to the verifier. The verifier is aware of the disclosed attributes and only verifies if the user holds them or not.

5.8 Summary

In this chapter, we have analyzed ABCs as a mature technology suitable for safeguarding user privacy and digital identity in contemporary access control systems.

The RKVAC scheme has demonstrated robust performance even on resource-constrained devices such as smart cards. This scheme is based on the work of Hajny et al. [11] and presented in Chapter 3. It meets privacy and security requirements and implements reliable revocation mechanisms, which are critical features for practical deployment. Building on the RKVAC cryptographic core, we have developed our own PEAS technology, which has been tested and piloted in a real-world environment using smart cards, smartphones, and smartwatches as user authentication devices. Our results suggest that PEAS is suitable for real-world deployment, particularly when wearables and smartphones are used as user devices because of their high efficiency, support for graphical user interfaces, and fast operation.

The contents of this chapter are derived from the work published by Casanova-Marqués and Dzurenda [13] and extended in [14]. The key contributions of the PhD candidate to the research discussed in this chapter encompass the design, development, and benchmarking of both the smartphone and smartwatch applications. The smart card application is derived from Chapter 3 and has been further refined with targeted enhancements and optimizations. Finally, the candidate has committed efforts to partitioning the desktop application introduced in Chapter 3. This involves relocating the cryptographic core to the *libpeas* library, thereby enabling its utilization across diverse devices. In addition, the candidate has worked on enhancing the desktop application to broaden its scope to not only encompass PC/SC protocols but also seamlessly integrate with TCP and Bluetooth communication protocols.

6 Zero-knowledge proofs for secure cooperative indoor positioning

In the context of CIPS, safeguarding privacy and security is of paramount importance, given that these systems process and handle sensitive location data that can be exploited to monitor and infringe upon individuals' privacy. However, existing solutions for CIPS do not provide adequate privacy protection, primarily due to their reliance on centralized data sources, which may expose users to data breaches and unauthorized access. Moreover, the positioning information in CIPS is often transmitted in plaintext, further exacerbating the risk of privacy violations. This chapter introduces a novel approach to addressing privacy and security concerns in CIPSs. The structure of this chapter is as follows:

- An introduction to CIPS, including an overview of their security and privacy concerns and background information.
- A detailed explanation of the cryptographic scheme and a comprehensive security and privacy analysis.
- Use-case scenarios demonstrating the practical applications of the proposed scheme.
- Implementation details and benchmark results.
- A comparative analysis of the proposed protocol with existing protocols in the field.
- A summary of the key findings.

6.1 Introduction

Undoubtedly, *Location-based Services* (LBSs) are present in our daily lives. They can suggest optimal routes to reach a place, enable autonomous industrial vehicles [89, 90], or even track lost or stolen objects or elders at home [91, 92]. Recently, wearable-based CIPSs have gained prominence, spurred by the desire to overcome the disadvantages of traditional approaches such as Wi-Fi or BLE fingerprinting [93]. That is, an expensive infrastructure of beacons and servers, as well as limited positioning accuracy of around a few meters [94]. These collaborative systems enable users to exchange information, for instance in terms of BLE advertising packets such as iBeacon, which can be used to compute the relative distances between them using the *Received Signal Strength* (RSS) values gathered. This alternative is simple to set up and provides a reasonable trade-off between power consumption and performance [95]. However, the existing BLE protocols, such as iBeacon, are not currently prepared to provide sufficient privacy protection.

Data privacy is both a regulatory must and an increasing consumer expectation [96, 97]. A few years ago, people were concerned about the privacy of their personal information; now, privacy is a requirement. Every user's data must be subjected to the most stringent protection and security analysis. This concern for privacy and the requirement that a user cannot be easily or unambiguously identified compel us to review the authentication mechanisms currently in use. Given this predisposition, it is imperative to establish authentication techniques that allow users to apply their privacy requirements. Currently, authentication schemes based on smart cards or biometrics are traceable and linkable [98]. These schemes acquire excessive information from users to verify that they have legitimate credentials to utilize a certain service or get access to a restricted area.

Systems of such an invasive nature have prompted the development of alternatives that respect the privacy of their users more. These suggestions are primarily concerned with the use of zero-knowledge proofs for the anonymous authentication and verification of users. In particular, *Attribute-based Authentication* (ABA) schemes have gained popularity since they enable users to anonymously and selectively demonstrate their ownership of personal attributes. Personal characteristics such as the legal age, citizenship, the validity of a transportation ticket, and the SARS-CoV-2 test result. Nevertheless, these schemes are typically centralized models in which the issuer and verifier are the same entity. This design constitutes a "single point of failure" in the system's verification procedures, jeopardizing its security and availability.

Existing CIPSs are also based on centralized models [93]. Adopting a decentralized architecture is a potential countermeasure to avoid the drawbacks of centralized systems. However, the usage of decentralized paradigms poses significant security and privacy concerns due to the constant communication among unknown devices and the absence of secure communication protocols. In addition, the information transferred through BLE, utilizing protocols such as iBeacon, is sent in plain text and without authentication. Since malicious users are capable of impersonating genuine users, injecting fake information, and performing eavesdropping [10], every piece of information exchanged must be appropriately authenticated and protected.

6.2 State of the art

In the field of research, numerous methods have been presented to safeguard the privacy of user location information in the context of Location-based Services. These solutions aim to protect sensitive information from being disclosed to unauthorized parties, such as third-party advertisers or malicious individuals, while still allowing LBS applications to provide personalized and relevant services to users. This is

becoming increasingly important as the use of LBS continues to grow and more data is generated through these services.

An illustration of Location-based Services can be found in the realm of the *Internet of Vehicles* (IoV). IoV applications are often dependent on the collection and processing of sensitive information. This information may contain driving habits, personal information, and location data, making privacy a critical issue that must be addressed to ensure the protection of users' information from unauthorized disclosure or misuse. Malandrino et al. [99] proposed a method to verify and infer the positions of vehicles in vehicular networks while preserving their privacy. The authors used the technique of anonymous beaconing, which broadcasts the positions of vehicles without revealing their identities, to achieve this goal. Liu et al. [100] presented a new privacy-preserving trust evaluation scheme named LPPTE, which aimed to enhance the fusion of data from different sources in cooperative vehicular safety applications. The authors pointed out that conventional trust evaluation schemes were not appropriate for these applications because of privacy issues, and introduced a lightweight alternative that maintained privacy while enabling the evaluation of trust among sources. The scheme was thoroughly explained, and its performance was assessed through simulations. Huang et al. [101] explored the privacy challenges in LBSs in the IoV. The authors proposed a privacy-preserving scheme that aims to protect the privacy of users while still providing accurate location information. Xi et al. [102] discussed a privacy-enhancing technology for the IoV. The authors proposed a *Zero-knowledge Proof* (ZKP)-based anonymous mutual authentication scheme called ZAMA to provide secure communication between vehicles and other entities in the IoV while preserving privacy. ZAMA uses ZKPs to verify the identity of participants without revealing any personal information, thereby improving the security and privacy of the IoV. The aforementioned publications suffer from several security concerns, including the transmission of data in plaintext, reliance on centralized servers, linkability and traceability of user actions, and the lack of revocation mechanisms.

The utilization of LBSs extends to *Global Positioning System* (GPS) navigation, location-based advertising, and location-based social networking. As LBS continue to evolve, preserving privacy and security is becoming increasingly important. Several research articles explore methods and techniques for ensuring privacy in LBS applications, such as privacy-preserving schemes and algorithms, differential privacy, and blockchain technology. These privacy-preserving solutions are not only relevant to existing LBS applications, but also have the potential to enhance privacy and security in emerging applications, such as *Coronavirus Disease 2019* (COVID-19) contact-tracing solutions.

Peng et al. [103] presented a study on a privacy-preserving scheme for LBSs that

addressed the issue of trajectory privacy. The authors proposed a collaborative approach for preserving the privacy of a user's trajectory by using a combination of cryptographic methods and data perturbation techniques. The study evaluated the performance of the proposed scheme and compared it to existing methods in terms of privacy protection, computation overhead, and communication costs. Jarvinen et al. [104] introduced PILOT, a pioneering *Indoor Positioning System* (IPS) that addresses privacy concerns by using several advanced techniques such as secure multi-party computation of distance metrics, quantization of RSS values, the *k-Nearest Neighbors* (k-NN) algorithm, and oblivious array access. They reported a running time below 1 second and provided a comprehensive evaluation of their solution. Gupta and Shanker [105] focused on improving the performance of LBSs through data caching. The authors suggested a new cache management policy called OMCPR that uses spatial k-anonymity to balance the trade-off between preserving user privacy and efficiently using cached data. The results indicated that OMCPR outperformed other policies in terms of both privacy protection and performance enhancement for LBSs. Shubina et al. [106] delved into the delicate equilibrium between location accuracy and privacy in wearable networks, exploring the challenges that arise when seeking to achieve high accuracy while ensuring user privacy. The study proposed several potential solutions to this conundrum and evaluated their performance. Additionally, the implications associated with implementing these solutions were discussed in depth. Kim et al. [107] introduced a survey paper that examines the use of *Differential Privacy* (DP) techniques in LBSs. The authors provided an overview of the existing privacy-preserving methods for location data and discussed their applicability to real-world LBSs. Barsocchi et al. [108] presented a new reference architecture for indoor positioning and discussed the challenges encountered during its installation and operation in real-world scenarios. In particular, they explored the database infrastructure and security procedures required to ensure data isolation, anonymization, and preservation in accordance with current legislation. Jiang et al. [109] presented an overview of the opportunities, challenges, and potential applications of DP in the *Industrial Internet of Things* (IIoT). The authors highlighted the importance of DP in preserving the privacy of users in IIoT, while enabling efficient data processing and analysis. Shubina et al. [110] conducted a qualitative comparison of current COVID-19 contact-tracing solutions in development, analyzing factors such as positioning technology, measurement, architecture, detection accuracy, energy efficiency, and privacy level. The authors emphasized the importance of privacy as a critical factor in these solutions, while also highlighting the specific strengths and characteristics of each solution. Yang et al. [111] presented a privacy-preserving solution for indoor navigation systems by incorporating location-based oblivious sharing. This allows for shared access to location

information while protecting users' privacy by obscuring their exact locations. Li et al. [112] explored the integration of LBSs with blockchain technology. The authors proposed a novel method to enhance the security and trust in location data management using a blockchain-based system named SAGIN. This method aimed to improve the efficiency of location data management while preserving the privacy and security of users' data. Hu et al. [113] introduced PriHorus, a novel privacy-preserving IPS based on RSS and partial homomorphic encryption. Unlike prior work such as PILOT, PriHorus relied on maximum likelihood estimation instead of the classical k-NN algorithm. Guo et al. [114] presented FedPos, a novel federated transfer learning framework for indoor positioning based on *Channel State Information* (CSI) from a single *Access Point* (AP). Although their model showed high transferability, it is currently limited to devices capable of measuring CSI data, which remains unavailable in wearable devices. The primary issue with most of the aforementioned publications is their utilization of a centralized server architecture to support their proposed LBSs, which may introduce security and privacy vulnerabilities and increase the risk of unauthorized data access or breaches. Moreover, some of these papers suggest mechanisms that may lead to traceable or linkable information, posing potential threats to user privacy.

Location-based Services span a broader range of applications, including Collaborative Indoor Positioning Systems. Unfortunately, privacy in CIPS has received far too little attention. The systematic review conducted by Pascacio et al. [93] served as a starting point for our analysis of the various CIPSs already in use. The review selected and assessed a total of 84 papers that were published between 2006 and 2020. None of the examined works accounted for user privacy or CIPSs security. We performed a literature search using scientific databases, including the most prominent computer science journals and conferences: IEEE Xplore, ACM Digital Library, Elsevier, ScienceDirect, and Springer Link. We were only able to choose two publications that directly or indirectly address the absence of security and privacy in these systems. Zidek et al. [115] designed the Bellrock scheme, which combines an ecosystem of standard beacons with user-specific beacons. Bellrock offers access control to conventional beacons and anonymity to user-based beacons, utilizing three methods, i.e., random, synchronized, and encrypted, to produce pseudo-anonymous identifiers that may be unmasked by a server. Yin et al. [116] suggested a federated localization framework called FedLoc. The framework seeks to provide precise location services cooperatively without jeopardizing user privacy, particularly sensitive information pertaining to their geographic trajectories. The above proposals have a significant drawback in that they rely on a central server, making the system a centralized environment. This dependence on a central server poses various risks, including the possibility of a *Denial of Service* (DoS) attack or even the risk of

spoofing, among others.

The landscape of collaborative LBSs has recently seen several proposals introduced. Fraile and Koulamas [117] proposed an indoor positioning system based on mobile devices that uses BLE signals to estimate the location of mobile devices in indoor environments. Delgado et al. [118] suggested a system for connected robots that employs a distributed architecture with multiple robots collaborating to accomplish mission-critical tasks. Pascacio et al. [119] put forward a neural network-based approach to improve indoor positioning accuracy using BLE RSS lateration. Finally, Wong and Lee [120] introduced an indoor navigation and information sharing system for emergency response situations that uses *Building Information Modeling* (BIM) and multi-user networking to facilitate collaboration among emergency responders. All of these proposals could potentially expose user data and undermine system reliability without adequate privacy protocols. Therefore, developing robust privacy protocols is crucial to ensure the privacy and security of users.

6.3 Cryptographic scheme

We propose an innovative decentralized attribute-based authentication protocol. This section discusses the entities involved, as well as the cryptographic design of the algorithms comprising the protocol.

To aid in the clarity and readability of this chapter, we provide a table of symbols used throughout our cryptographic protocol. Table 6.1 defines each symbol and its associated meaning, allowing for a comprehensive understanding of the protocol's components.

6.3.1 Entities

The following entities comprise the system model presented in Figure 6.1:

- *Issuer*: is responsible for issuing the personal attribute m_{ID} gathered in a cryptographic credential using the `Issue` algorithm. The personal attribute m_{ID} is the user identifier obtained during the system registration. The cryptographic credential is signed by the private key of the issuer sk_I. In our design, the issuer is also responsible for revoking invalid users from the system. This is done to simplify interactions with other entities. The revocation attribute m_r is additionally aggregated to the cryptographic credential.
- *User*: acquires the unique credential, including the attributes issued by the issuer, to gain access to the system or service. Using the `Show` algorithm, the user then anonymously demonstrates ownership of the attributes to the

Tab. 6.1: Table of symbols

Symbol	Definition
$q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathbf{e}, g_1, g_2$	parameters for the selected pairing-friendly elliptic curve.
m_{ID}	attribute with the user's identifier.
m_r	attribute for revocation based on the week and year.
$sk_I, \mathcal{X}_0$	key pair of the issuer (private and public keys).
$i_r, \mathcal{I}_r$	shared key pair among system users (private and public keys).
$\sigma, \sigma_{x_0}, \sigma_{x_r}, \sigma_{x_{ID}}$	cryptographic credential issued to the user.
$\hat{\sigma}, \hat{\sigma}_{x_0}, \hat{\sigma}_{x_r}, \hat{\sigma}_{x_{ID}}$	cryptographic credential randomized by the user.
ρ	random number used to randomize the cryptographic credential.
ρ_κ, ρ_{ID}	random numbers used to compute the protocol commitment and responses.
t_κ	cryptographic commitment computed by the user.
e	challenge used in the cryptographic protocol.
s_κ, s_{ID}	responses obtained during the execution of the cryptographic protocol.
τ	random number used to randomize the transaction identifier.
$\mathcal{R}$	transaction identifier.
ψ	alias of $\hat{\sigma}, \hat{\sigma}_{x_0}, \hat{\sigma}_{x_r}, \hat{\sigma}_{x_{ID}}$.
π	alias of e, s_κ, s_{ID}.
λ	information transmitted or received in plaintext.
ξ	information transmitted or received in ciphertext.
t'_κ	cryptographic commitment reconstructed by the verifier.

verifier. The user may additionally transmit information to the verifier in either plaintext or encrypted format.

- *Verifier*: utilizes the `Verify` algorithm to confirm the user's possession of the attributes. If the ownership of the attributes is successfully validated and the user's access has not been revoked, the verifier accepts the received information. If not, the information will be rejected.

6.3.2 Protocol specification

Following is a description of the algorithms, including their input and output parameters:

- $(params) \leftarrow$ `Setup` (1^κ): the algorithm receives the security parameter κ as input and generates the public system parameters. These parameters are a bilinear group with parameters $params = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathbf{e}, g_1, g_2)$ that satisfy $|q| = \kappa$.
- $(sk_I, \mathcal{X}_0, i_r, \mathcal{I}_r) \leftarrow$ `KeyGen` $(params)$: the algorithm randomly selects the private keys $sk_I \leftarrow (x_0, x_r, x_{ID}) \in_R \mathbb{Z}_q$ and computes the issuer's public key

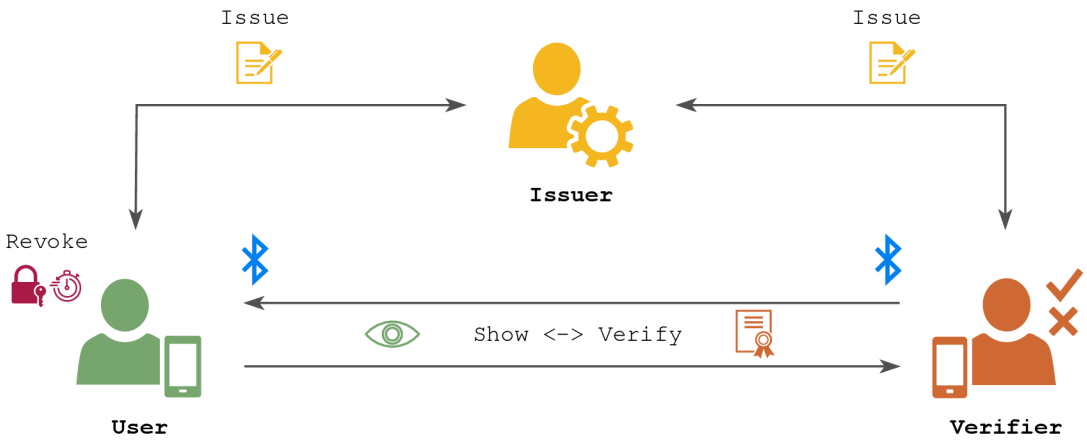


Fig. 6.1: Entities and algorithms constituting the proposed protocol

$\mathcal{X}_0 = g_2^{x_0}$, based on the system parameters *params*. In addition, the algorithm also randomly generates the shared secret key $i_r \in_R \mathbb{Z}_q$, and calculates the shared public key $\mathcal{I}_r = g_1^{i_r}$, which will be utilized by system users. The algorithm outputs the issuer keys $(sk_I, \mathcal{X}_0, i_r, \mathcal{I}_r)$. The issuer runs the `KeyGen` algorithm.

- $(\sigma, \sigma_{x_0}, \sigma_{x_r}, \sigma_{x_{ID}}, i_r) \leftarrow$ `Issue` $(params, sk_I, m_r, m_{ID})$: the algorithm inputs the private keys of the issuer $sk_I \leftarrow (x_0, x_r, x_{ID}) \in_R \mathbb{Z}_q$, the revocation attribute m_r, and the user attribute m_{ID}. The revocation attribute m_r is set to $ww/yyyy$ in our implementation, which means that m_r is the current year and the week of that year. The algorithm is run between the user and the issuer and is shown in Figure 6.2. First, the user sends its attribute m_{ID} to the issuer. Next, the issuer signs the attributes m_r and m_{ID} with the secret keys as $\sigma = g_1^{\frac{1}{x_0 + m_r x_r + m_{ID} x_{ID}}}$ and computes auxiliary values $\sigma_{x_0}, \sigma_{x_r}, \sigma_{x_{ID}}$, where $\sigma_{x_0} = \sigma^{x_0}$, $\sigma_{x_r} = \sigma^{x_r}$, and $\sigma_{x_{ID}} = \sigma^{x_{ID}}$. The algorithm outputs the cryptographic credential σ and auxiliary values $(\sigma_{x_0}, \sigma_{x_r}, \sigma_{x_{ID}})$ to the user. The algorithm also securely provides the user with the shared secret key i_r. This key must remain secure on the device.
- $(\mathcal{R}, \psi, \pi, \lambda \text{ or } \xi) \leftarrow$ `Show` $(params, m_r, m_{ID}, \sigma, \sigma_{x_0}, \sigma_{x_r}, \sigma_{x_{ID}}, \mathcal{I}_r, \lambda, timestamp)$: the algorithm receives the timestamp from the verifier, the revocation attribute m_r, and the user attribute m_{ID} as inputs. To prevent replay attacks, the user is required to verify that the received timestamp is no more than two seconds earlier or later than the current time. In our implementation, we assume all users have an *Network Time Protocol* (NTP)-synchronized date and time on their devices. The algorithm outputs the transaction identifier $\mathcal{R}$, the randomized user credentials ψ, the cryptographic proof of possession of the attributes π, and the data to be transferred in an encrypted format ξ. Nonetheless, if

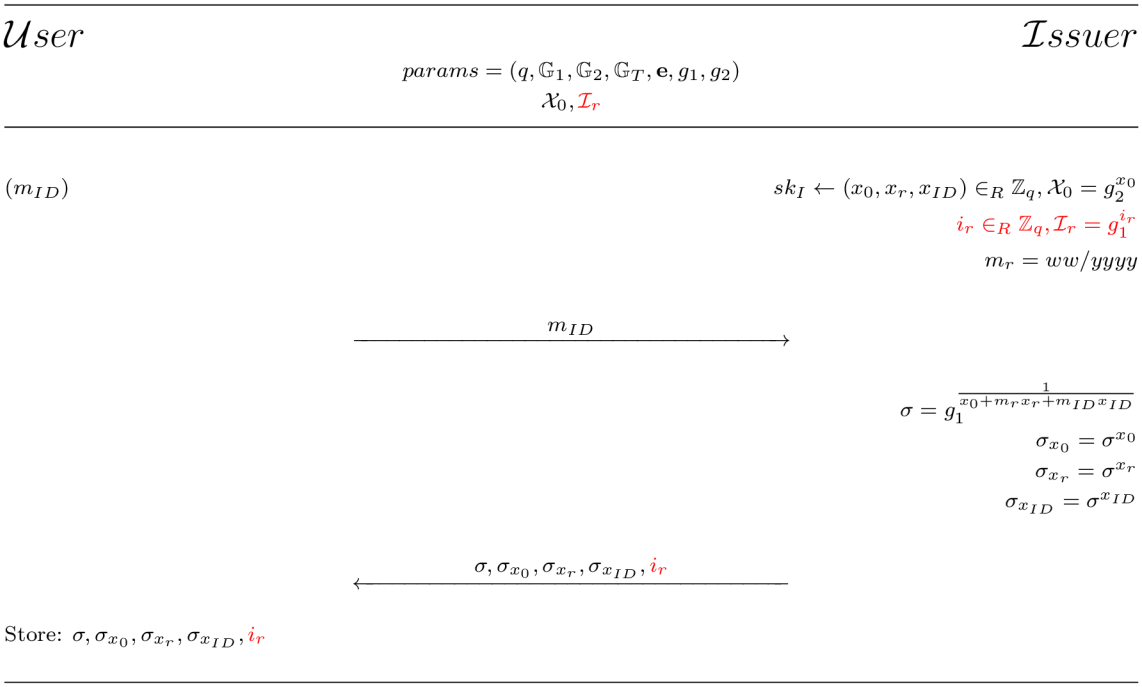


Fig. 6.2: Definition of the `Issue` algorithm

the user executes the algorithm in a public environment, data is transmitted in plaintext λ, i.e., unencrypted. Thus, the output of the algorithm is λ and not ξ. This characteristic is described in full in Section 6.5. Users execute the `Show` algorithm. Figure 6.3 depicts a comprehensive explanation of the `Show` algorithm. The user begins by randomizing their credentials. The user then calculates the commitment t_κ and uses the hash of t_κ as the symmetric key to encrypt the data that needs to be sent. Lastly, the user computes a proof of knowledge for all the attributes in the credential. Note that when we require encrypted data transmission, we utilize the red-highlighted formulae. The $\mathcal{R}$ value is the transaction identifier, but it is also required for the verifier to construct the decryption key $\mathcal{H}(t'_\kappa)$. If the information is to be sent as plaintext λ, we can omit the operations marked in red and send the data straight.

- $(0/1) \leftarrow$ `Verify` $(params, timestamp, \mathcal{X}_0, i_r, \mathcal{R}, \psi, \pi, \lambda$ or $\xi)$: the algorithm inputs the timestamp previously generated for the user, the shared secret key i_r, the transaction identifier $\mathcal{R}$, the randomized user credentials ψ, the cryptographic proof of possession of the attributes π, and the user data in an encrypted format ξ. Nonetheless, if the verifier executes the algorithm in a public environment, data is received in plaintext λ, i.e., unencrypted. Thus, the input of the algorithm is λ and not ξ. This characteristic is described in full in Section 6.5. Verifiers execute the `Verify` algorithm. Figure 6.3 depicts a comprehensive explanation of the `Verify` algorithm. The verifier begins by

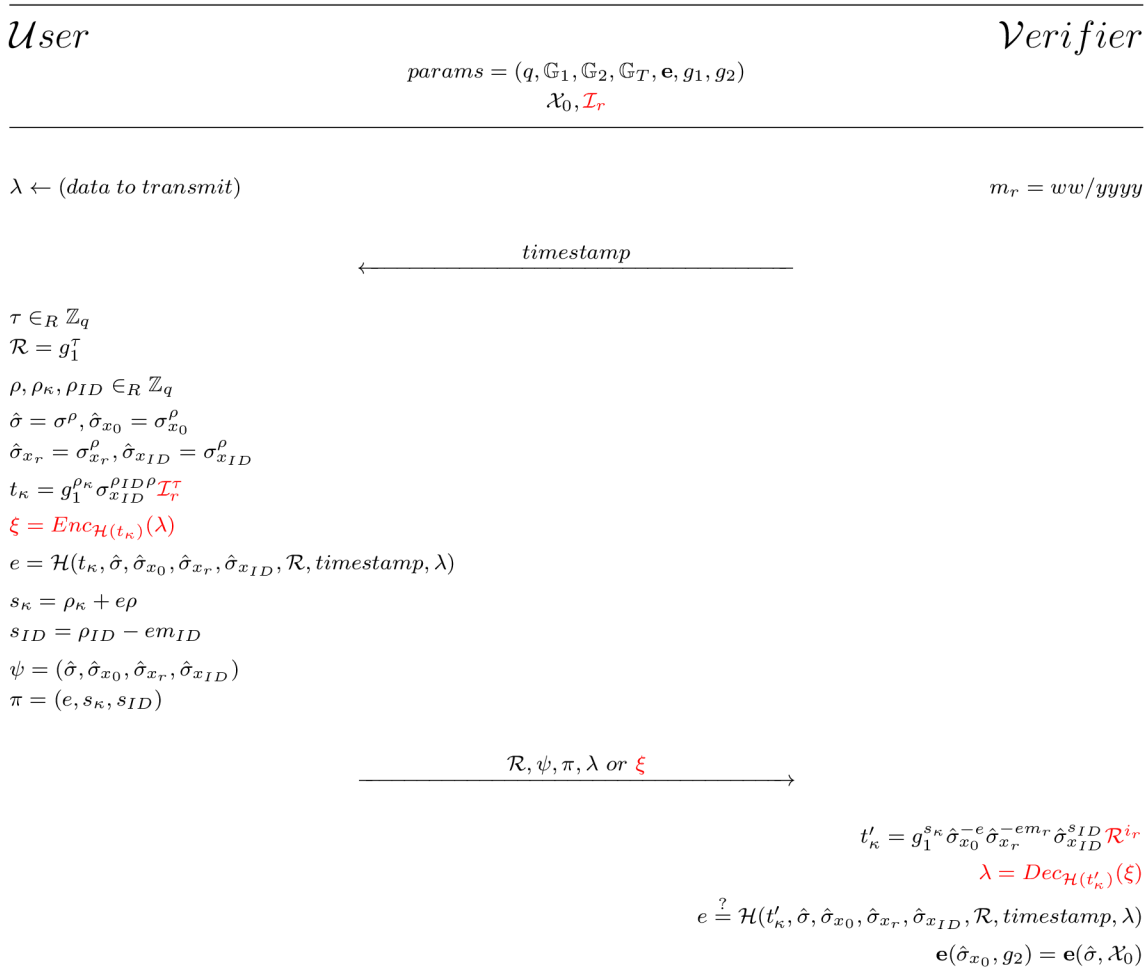


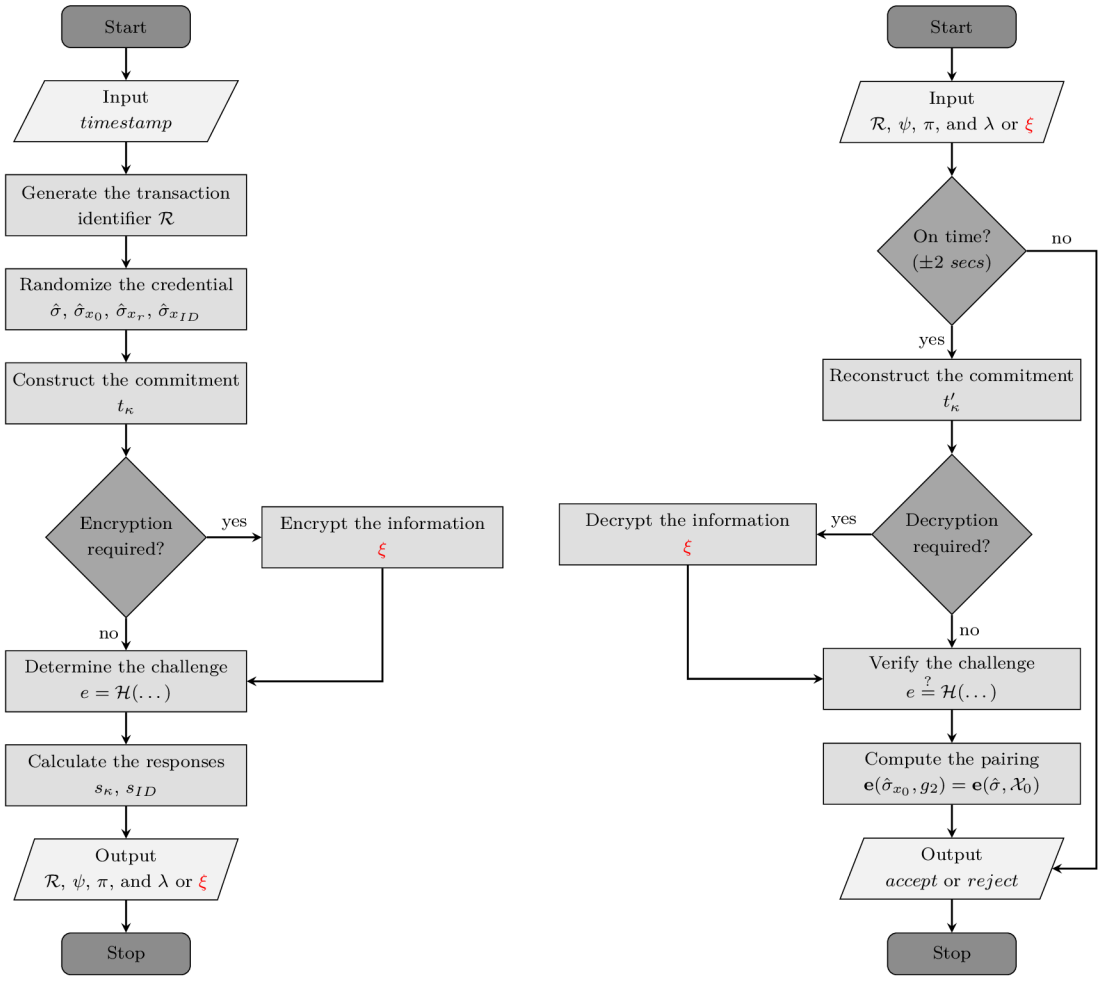
Fig. 6.3: Definition of the `Show` and `Verify` algorithms

recalculating the commitment t'_κ. The verifier then uses the hash of t'_κ as the symmetric key to decrypt the user data. In addition, the verifier also uses t'_κ to create the cryptographic hash $\mathcal{H}(t'_\kappa, \hat{\sigma}, \hat{\sigma}_{x_0}, \hat{\sigma}_{x_r}, \hat{\sigma}_{x_{ID}}, \mathcal{R}, timestamp, \lambda)$ and checks that the value of e received from the user matches. Lastly, the verifier computes a bilinear pairing to ensure that the randomized credentials correspond to a genuine user of the system and were emitted by the issuer. Note that when we require encrypted data reception, we utilize the red-highlighted formulae. Additionally, the shared secret key i_r must remain secure on the device to prevent non-system users from accessing the data. If the information is to be received as plaintext λ, we can omit the operations marked in red to facilitate the calculation.

- `Revoke`: the algorithm has no input parameters or output. In addition, it is not directly executed by any entity in the system. The revocation model is based on the expiration of an epoch. In our implementation, the revocation

attribute m_r indicates the week of the current year. Thus, when the week number changes, the credentials of all users are automatically revoked. The users must request reissuance from the issuer once their credentials have expired. Note that the issuer's shared keys are regenerated weekly to prevent revoked users who fail to renew their credentials from continuing to read data. Therefore, every week the i_r and $\mathcal{I}_r$ keys are invalidated, and new shared keys are produced.

The flowcharts presented in Figure 6.4 provide a high-level definition of the `Show` and `Verify` algorithms and are a valuable tool for facilitating a comprehensive understanding of the protocols, their underlying mechanisms, and the sequence of steps involved in their execution. By visually presenting the key components of each algorithm and the relationships between them, the flowcharts enable readers to quickly grasp the fundamental concepts of our cryptographic protocol.



(a) High-level `Show` algorithm (b) High-level `Verify` algorithm

Fig. 6.4: High-level definition of the `Show` and `Verify` algorithms

6.4 Security and privacy discussion

This section provides an examination of the security and privacy requirements of our novel decentralized ABA protocol for CIPSs. The protocol relies on a robust authentication mechanism to ensure secure interactions among users. Our discussion substantiates the critical security, privacy, and functionality properties that the protocol provides. This analysis demonstrates that the proposed protocol satisfies these security and privacy requirements, establishing it as a dependable solution for securing CIPSs. For a formal and more in-depth analysis, please refer to B.

6.4.1 Required properties

We examine the essential security, privacy, and functionality properties, including anonymity, unlinkability, and untraceability; confidentiality and integrity; correctness; key-parameter consistency; as well as unforgeability, completeness, soundness, and zero-knowledge, that the protocol must provide to guarantee the security and privacy of CIPSs.

Anonymity, *unlinkability*, and *untraceability* are essential properties to ensure the user's privacy. The anonymity property ensures that a party's identity is hidden or kept confidential during a protocol's execution, preventing it from being linked to any exchanged information. The unlinkability property guarantees that different protocol executions cannot be linked to the same party or identity, thereby preventing any correlation of the party's actions or information across multiple protocol executions. Lastly, the untraceability property ensures that a party's actions or information exchanged during a protocol execution cannot be traced back to the party, avoiding any identification of the party's actions through any means, including network monitoring or traffic analysis. The properties of anonymity, unlinkability, and untraceability are achieved in our protocol through the utilization of standard zero-knowledge protocols. The randomized user credential in each protocol execution ensures the prevention of user identification or tracing, which further enhances the security and privacy of the protocol. These properties have been formally proven in Proposition 10 of our security and privacy analysis.

Confidentiality and *integrity* properties are fundamental requirements in secure communication protocols. Confidentiality ensures that sensitive information exchanged during a protocol's execution remains confidential and protected from unauthorized disclosure, and only authorized parties can access or read it. Integrity guarantees that the information remains trustworthy throughout the protocol execution and is not tampered with or altered in any way. Our protocol achieves confidentiality and integrity by using standard cryptographic primitives such as *Advanced*

Encryption Standard (AES) and *Secure Hash Algorithm 3* (SHA-3), which are widely used and trusted in the field. Additionally, our protocol offers flexibility to the user by providing the option to encrypt the information exchanged during the protocol execution, ensuring its confidentiality and protection from unauthorized access. Nevertheless, the user can also decide to share the information in plaintext, making it available to everyone without compromising its integrity. These properties have been formally proven in Proposition 11 of our security and privacy analysis.

Correctness is of the utmost importance, as it ensures that the information exchanged during a protocol's execution is accurate and in line with the intended specification. This property guarantees that the protocol achieves its desired goal while preventing errors or misunderstandings in the exchange of information. Our protocol guarantees correctness through the use of advanced cryptographic techniques such as commitment reconstruction, hash functions, and pairings. These techniques help to ensure that the information exchanged during the protocol execution is accurate and reflects the state of the system. This is accomplished by verifying the integrity of the data to ensure that it has not been tampered with or altered in any way. This property has been formally proven in Proposition 7 of our security and privacy analysis.

Key-parameter consistency is crucial to ensure the security of the protocol, as it guarantees that the keys and parameters used during a protocol's execution are valid and consistent. This property prevents the protocol from relying on faulty or invalid keys or parameters that could be exploited by an attacker. Our protocol ensures key-parameter consistency by generating keys and parameters from truly random sources, which makes them both unpredictable and secure. We also validate their authenticity and consistency before using them in the protocol to ensure that only legitimate and properly generated keys and parameters are used. This approach helps prevent attacks that exploit weaknesses or vulnerabilities in the keys and parameters used in the protocol. This property has been formally proven in Proposition 9 of our security and privacy analysis.

Unforgeability, *completeness*, *soundness*, and *zero-knowledge* are indispensable to the security and privacy of our protocol. The unforgeability property ensures that only the authorized entity can produce valid signatures and prevents adversaries from forging signatures without access to the signer's private key. The completeness property is essential to ensure that a protocol execution is comprehensive and does not leave any loopholes or opportunities for attacks, ultimately guaranteeing the protocol's overall security and reliability. This property ensures that all valid inputs are accepted and that the protocol produces a valid output, leaving no room for ambiguity or incomplete execution. The soundness property ensures that a protocol execution provides verifiable and unambiguous evidence that cannot be tampered

with or disputed. It guarantees that any invalid inputs are rejected and further strengthens the protocol's security and integrity. Finally, the zero-knowledge property ensures that parties can prove their knowledge of secrets without revealing any information beyond what is strictly necessary. Our protocol satisfies those properties through the use of standard signature and zero-knowledge protocols. Specifically, the protocol leverages well-established techniques such as the weak Boneh-Boyen digital signature, the Fiat-Shamir transform, and the Schnorr identification scheme to achieve these goals. This ensures that the protocol is both secure and efficient while also providing the necessary guarantees to protect the privacy of users in the system. These properties have been formally proven in Propositions 6, 7, and 8 of our security and privacy analysis.

6.5 Use cases for the proposed scheme

This section presents two use cases for the decentralized attribute-based authentication protocol. In addition, we introduce the privacy-enhanced mode. In this mode, users and verifiers may select whether to actively participate in the system or not.

6.5.1 Public environments

In this first scenario, we assume users are present in an environment with public access. For instance, shopping malls, universities, and hospitals. Workers, as well as customers, students, and patients, have unrestricted access to these buildings. In this scenario, the data can be transmitted authenticated but unencrypted, i.e., in plaintext. Users can download the mobile application and decide whether to register with the system. An unregistered user will be able to obtain positioning data that has been authenticated by valid users of the system, but will be unable to transmit data because they lack valid credentials. In contrast, registered users will also be able to transmit positioning information. Executing the `Show` and `Verify` algorithms, depicted in Figure 6.3, enables the transmission of authenticated information in plaintext, i.e., unencrypted. These algorithms must be executed without performing the red calculations. For a high-level overview of the processes involved, please refer to the accompanying flowcharts of the `Show` algorithm in Figure 6.4a and the `Verify` algorithm in Figure 6.4b.

Figure 6.5 depicts various types of users and verifiers participating in a public environment, i.e., positioning them in a shopping center. The green figures represent users who have registered with the system and are permitted to transmit positioning information. The blue figures represent users who are not registered with the system and can therefore only receive and validate the information from

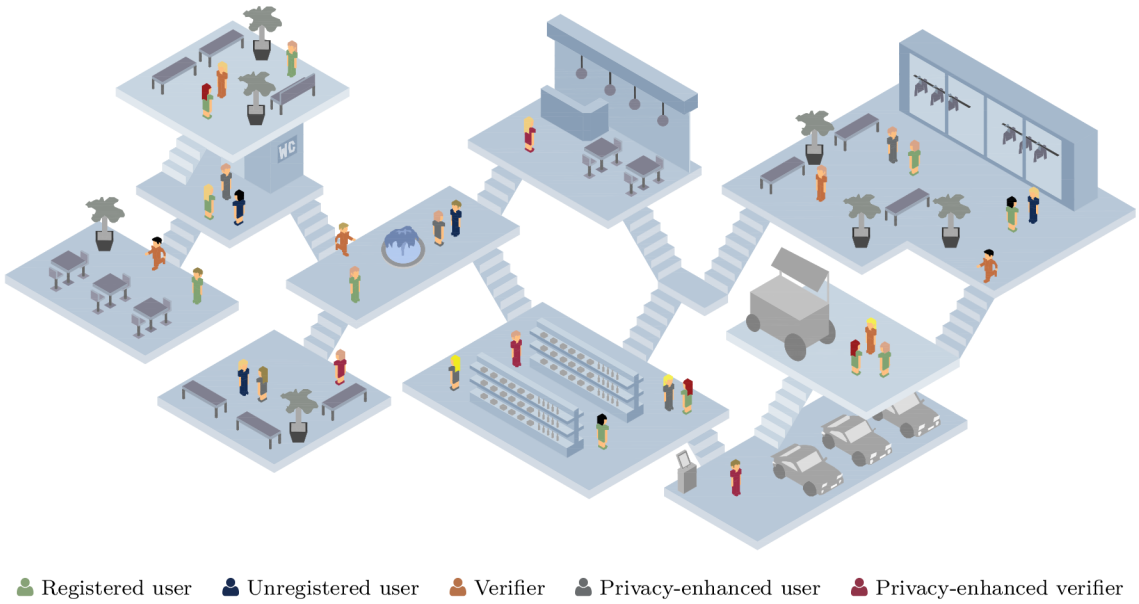


Fig. 6.5: Practical use case of the decentralized attribute-based authentication protocol

the green users, meaning they can only act as verifiers requesting positioning information. Verifiers, or users who have requested positioning data to improve their location, are represented by the orange figures. Finally, users and verifiers operating in privacy-enhanced mode are represented by the gray and red figures, respectively. This mode is described in detail in Subsection 6.5.3.

6.5.2 Private environments

In this second scenario, we assume users are present in an environment with non-public access. For instance, military facilities, critical infrastructures, or private corporations. Access to these buildings is restricted, and only employees have permission to enter. In this scenario, data should be transmitted authenticated and encrypted. Users are already registered within the system and must only download the mobile application. All users will possess valid credentials, allowing them to freely transmit positioning data. Executing the `Show` and `Verify` algorithms, depicted in Figure 6.3, enables the transmission of authenticated and encrypted information. These algorithms must be executed, including the red calculations. For a high-level overview of the processes involved, please refer to the accompanying flowcharts of the `Show` algorithm in Figure 6.4a and the `Verify` algorithm in Figure 6.4b.

This use case is similar to the one shown in Figure 6.5, but all users must be registered and part of the system, so the blue figures are eliminated.

6.5.3 Privacy-enhanced mode

This mode of operation allows users and verifiers to enhance their privacy by blocking the transmission of information but permitting its reception. It is useful for improving the privacy of users in environments with few devices and where their identities are easily discernible.

Users participating in the system respond to verifier requests and send their positioning data in plaintext or encrypted. However, when they are using the privacy-enhanced mode, users do not respond to verifiers' requests and therefore do not interact with the system.

Verifiers participating in the system request positioning data from nearby users. When they receive the information, they can validate it and use it to refine their location. However, when they are using the privacy-enhanced mode, verifiers do not request information from users. They use the information received from users but requested by other verifiers. Therefore, they can validate and utilize the information received from users to improve their location while remaining hidden.

6.6 Implementation details

This section describes the development process and summarizes the main key points considered during the design of the application. Table 6.2 outlines the devices we employed and their hardware and software specifications. The device selection process was intentionally agnostic and unbiased, with devices chosen at random to ensure a diverse sample. This strategy ensured that both legacy and modern devices were included in our implementation, providing compatibility with a broad range of hardware and reinforcing the protocol's viability in real-world scenarios. The application was designed for several platforms, including single-board computers, smartphones, smartwatches, and microcontrollers. Such heterogeneous device types are common in IoT ecosystems and are representative of the different use cases that our protocol can support. While smartphones and smartwatches are ideal for obtaining positioning data actively by users, single-board computers and microcontrollers are better suited for industrial settings. They can enable autonomous vehicle positioning or enhance the location sensing capabilities of other devices, further extending the reach and flexibility of our solution.

We divided the implementation of the application into three components: *(i)* the Libre Collaborative Indoor Positioning (LibreCIP) library for core functionality; *(ii)* the *device wrappers* to ensure compatibility with the selected IoT devices (see Table 6.2); and *(iii)* the BLE integration for transmitting and receiving data using Bluetooth Low Energy.

Tab. 6.2: Hardware and software specifications of the devices

Device	CPU	OS	RAM
Single-board computers			
Raspberry Pi 4 Model B	ARM Cortex-A72	Raspberry Pi OS	4 GB
Smartphones			
Samsung Galaxy S21+ 5G	Exynos 2100	Android 11	8 GB
Samsung Galaxy S20 FE	Exynos 990	Android 10	6 GB
Samsung Galaxy A52	SDM720G	Android 11	6 GB
Samsung Galaxy A32	MTK D720 Dual + Hexa	Android 11	4 GB
Samsung Galaxy S8	Exynos 8895	Android 9	4 GB
iPhone 11	A13 Bionic	iOS 16.2	4 GB
iPhone XS Max	A12 Bionic	iOS 16.2	4 GB
PinePhone Pro	Rockchip RK3399S 64bit SoC	Arch Linux ARM	4 GB
Smartwatches			
Huawei Watch 2	Snapdragon 2100	Android Wear 2	768 MB
Apple Watch Series 5	Apple S5 (64-bit dual-core)	watchOS 9.2	1 GB
PineTime	ARM Cortex-M4F	RIOT 2022.10	64 KB
Microcontrollers			
Arduino Nano 33 BLE	NINA-b3 (nRF52840)	RIOT 2022.10	256 KB
Arduino Nano 33 IoT	ATSAMD21	RIOT 2022.10	32 KB

6.6.1 Core library

The cornerstone of the application is our LibreCIP library, a purpose-built software package that offers the complete implementation of the protocol. LibreCIP includes a highly optimized suite of advanced cryptographic functions such as user authentication and verification, data encryption and decryption, etc. In addition, it contains compression and decompression routines that enable efficient data transmission and storage while preserving the security and privacy of user information. The library is designed from scratch to offer unparalleled performance and security on a wide range of devices. It is written in the C programming language and relies on several third-party libraries. Specifically, we utilized `mcl` [33] and `relic-toolkit` [85] for elliptic curve cryptographic support; `lz4` [121] for providing data compression and decompression support; and `crypto`, the native cryptographic library of RIOT [122], for data encryption and decryption, as well as for cryptographic hash algorithm support. Note that we avoided the use of costly and difficult-to-compile libraries on devices with limited resources to enable code portability and easy integration into wearable devices. Consequently, we exclusively utilized libraries that are compatible with the RIOT operating system [123]. The library used to implement the

cryptographic core varies based on the device. We used `mcl` for Apple, Android, Raspberry Pi, and PinePhone Pro devices since it offers faster execution speeds. In contrast, we utilized the `relic-toolkit` for the PineTime and microcontrollers. The cryptographic backend can be selected during compilation.

The protocol was designed and implemented using elliptic curve cryptography. Specifically, we utilized the BN254 curve supplied by the `mcl` and the `relic-toolkit` libraries. Uncompressed points occupy 64 bytes and compressed points 33 bytes with this curve size. A scalar integer has a size of 32 bytes. We explored transmitting the points in their compressed form to reduce the size of the data.

Likewise, we examined data compression to further reduce the size of the information to be transmitted. We evaluated a number of open-source compression algorithms to select the one with the highest compression ratio for our protocol. Table 6.3 compares various compression algorithms, including `zlib` [124], `xz` [125], and `lz4` [121]. However, when working with such a small amount of data, there is no discernible difference between the benchmarked data compression algorithms. In addition, data size may grow owing to the inclusion of compression headers. Therefore, we chose `lz4` because the RIOT operating system natively supports it and because it is the only algorithm that reduces the size of the original data. I.e., the compressed data size is lower than the original data size.

Tab. 6.3: Comparison of the zlib, xz, and lz4 compression algorithms

Compression algorithm	Data size [B]	Compressed data size [B]	Compression ratio [%]
`zlib v1.2.13`	294	297	1.010
`xz v5.4.0`	294	299	1.017
`lz4 v1.9.4`	294	284	0.966

Lastly, we used the RIOT `crypto` library to encrypt and decrypt data and calculate cryptographic hashes. Specifically, we employed the AES algorithm in *Cipher Block Chaining* (CBC) mode with 128-bit keys for encrypting and decrypting and the SHA-3 function with 256-bit digests.

6.6.2 Device wrappers

The wrappers for each device are written in their respective native languages. They consist of the *User Interfaces* (UIs) of the devices specified in Table 6.2 as well as the BLE routines and libraries required for communication. The UIs are designed based on the device type. For Android, iOS, Wear OS, and watchOS devices, we created GUIs. In contrast, we developed CLIs for the Raspberry Pi and PinePhone Pro. We did not create UIs for the microcontrollers or the PineTime.

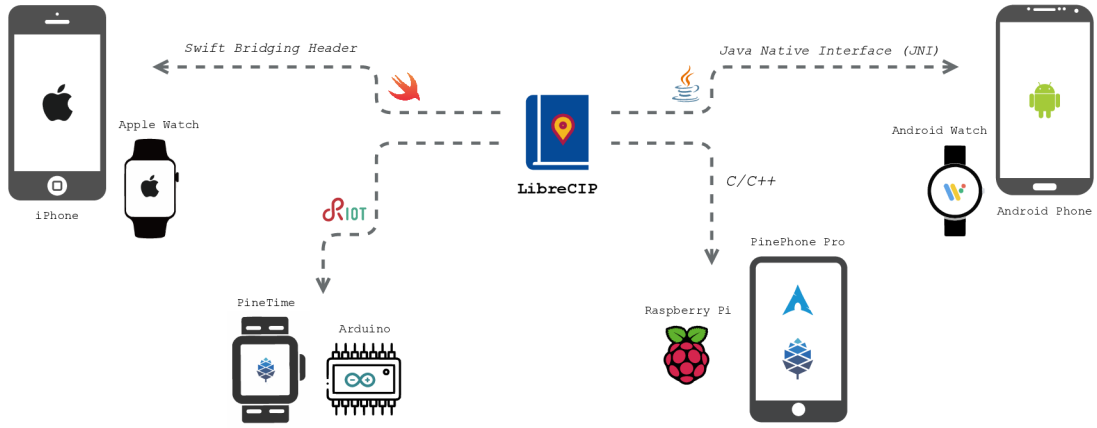


Fig. 6.6: Interoperability between LibreCIP and different devices

We used the Java SE Development Kit 17.0.6 to create the Android and Wear OS apps. The Android NDK enables us to develop portions of our application in native code using languages such as C and C++. This permits us to invoke the LibreCIP library functions via *Java Native Interface* (JNI). We used Swift to create the apps for iOS and watchOS. This language permits the execution of C-written functions through a bridge. This enables us to invoke LibreCIP library functions. The PinePhone Pro and Raspberry Pi applications were developed in C, so they can use the LibreCIP library directly without additional layers. The PinePhone Pro utilizes the Arch Linux ARM operating system with the *sxmo* desktop. We merely created a command-line program to assess its performance, as it is not a device ready for end users. The Raspberry Pi application is also a command-line application. We used RIOT for the microcontroller's applications. These devices lack display and output peripherals; therefore, we pondered creating applications that would always respond to user requests to improve their location accuracy and occasionally request their location to refine the precision of the device's own location. The applications are written in C and run when the devices are powered on. Although the PineTime device has a display, we did not design a GUI; hence, we also considered running the same type of application. Figure 6.6 depicts the different device wrappers and the technologies used by each to call LibreCIP library functions.

6.6.3 Bluetooth Low Energy integration

Our protocol transfers data using BLE advertising packets. Bluetooth 4.2 [126] and earlier versions specified a 31-byte payload size for a single BLE advertising packet. Bluetooth 5.0 [127], on the other hand, introduced a significant enhancement by

expanding the capacity of advertising packets. The advertising payload may include up to 254 bytes with Low-Energy Advertising Extensions. Since it is impossible to fit all data into 31 bytes, we contemplated utilizing Bluetooth 5.0. Unfortunately, Bluetooth 5.0 and the Low-Energy Advertising Extension are not supported by all actual devices. Therefore, we designed a packet format that is compatible with Bluetooth 4.2 and previous versions. This packet structure permits the transmission of information by fragmenting it into smaller packets and chaining them together, enabling other users to identify the number and sequence of packets to be received. Figure 6.7 depicts the BLE packet structure we designed for transmitting our protocol data.

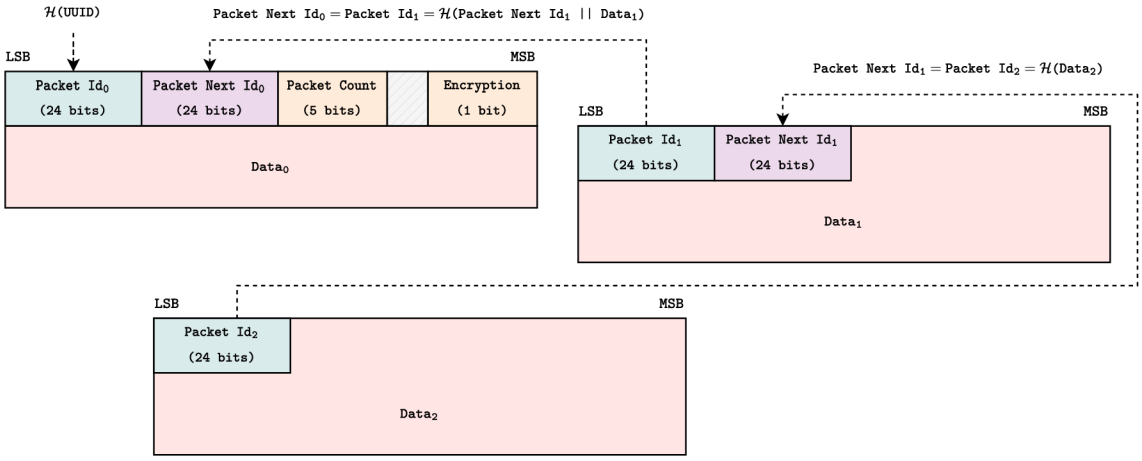


Fig. 6.7: Structure of the BLE advertising packets

The first packet comprises a 7-byte header and 24 bytes of data. The `Packet Id` field corresponds to the first three bytes of the application's *Universally Unique IDentifier* (UUID) hash value. That is, $\mathcal{H}(\texttt{UUID}) = \texttt{0x414...f5e}$. The `Packet Next Id` field corresponds to the first three bytes of the next packet's hash value. That is, $\mathcal{H}(\texttt{Packet}_1) = \texttt{0x141...5ef}$. The `Packet Count` field indicates the total number of packets. The `Encryption` field specifies whether the data is in plaintext or encrypted format. Intermediate packets maintain the `Packet Id` and `Packet Next Id` fields but lack the `Packet Count` and `Encryption` fields. The `Packet Id` field in these packets is equal to the first three bytes of the packet's hash. The final packet keeps the `Packet Id` field but omits the `Packet Next Id` field. The `Packet Id` field in this packet is also equal to the first three bytes of the packet's hash.

6.7 Experimental results

This section presents the results of the implementation of our protocol on different types of devices. To ascertain the feasibility of our approach and evaluate the

performance and speed of the algorithms, we conducted several experiments, benchmarking the entire protocol. Figure 6.8 illustrates the results of the execution. The figure depicts the benchmarks in milliseconds and includes the protocol run times and BLE communication overhead.

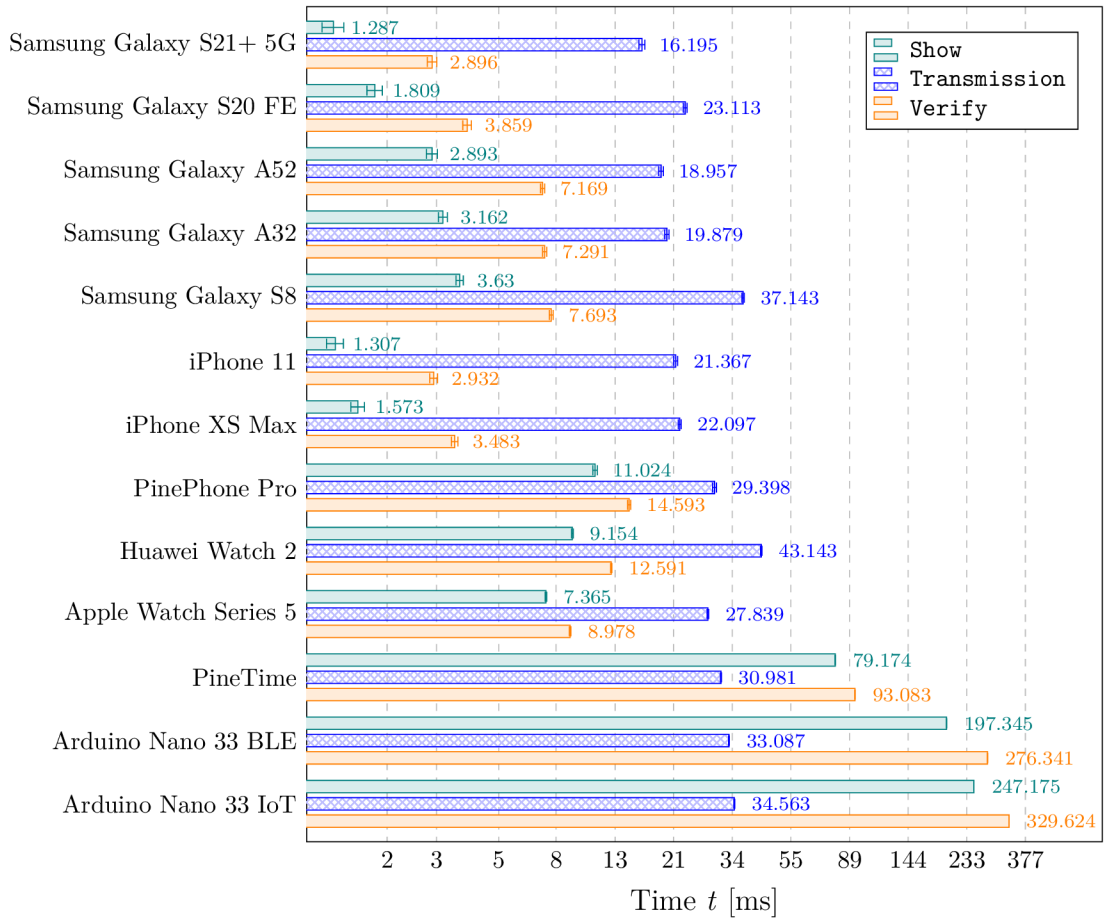


Fig. 6.8: Speed comparison of the `Show` and `Verify` algorithms, and the transmission overhead

To interpret these results, it is important to consider the limitations of the study. For example, the times were taken in a laboratory, away from environments crowded with transmitting devices, which could impact the generalizability of the results. Nevertheless, these results suggest that the protocol may be a promising option for protecting privacy in current CIPSs. Future studies should seek to replicate these results in larger and more diverse environments to fully understand their performance.

It is clear from the data that the smartphones, particularly the Samsung Galaxy S21+ 5G and iPhone 11, have the fastest computation times, with values around 1.3 milliseconds and 2.9 milliseconds for the `Show` and `Verify` algorithms, respectively. On the other hand, the Arduino Nano 33 BLE and Arduino Nano 33 IoT, which are

microcontroller boards designed for IoT applications, have much slower computation times, with values in the range of 197.345 milliseconds to 329.624 milliseconds. This difference in performance is largely because microcontroller boards have less processing power and memory resources compared to smartphones and smartwatches, which are equipped with more advanced and powerful hardware.

It is notable that all popularly used smartphones and smartwatches maintain `Show` and `Verify` algorithm processing times below 15 milliseconds. The quickest device executes the `Show` algorithm in 1.287 milliseconds, while the slowest does it in 9.154 milliseconds. In contrast, the fastest execution of the `Verify` algorithm requires 2.896 milliseconds, and the slowest one requires 12.591 milliseconds. The Raspberry Pi 4 benchmarks are omitted from the figure due to their negligibility. This device's execution time is on the order of μs. `Show` runs in 0.013 milliseconds, whereas `Verify` executes in 0.015 milliseconds. The communication overhead is 26.371 milliseconds.

It is also worth noting that the Huawei Watch 2, Samsung Galaxy S8, PineTime, and Arduino Nano devices have the slowest transmission times, compared to the top-performing smartphones, with values over 30 milliseconds. This can be attributed to the fact that these devices are equipped with older versions of Bluetooth technology or, as in the case of the PineTime, are not primarily designed for BLE communication.

The error bars in the Figure 6.8 represent the variability in the measurements obtained from multiple executions of our application on different devices. We observe that the smartphones in our study exhibit larger errors than the microcontrollers. This can be attributed to the fact that smartphones run full-fledged operating systems with multiple concurrent processes and services, which introduces more variability in the results. In contrast, the microcontrollers run a single application without sharing execution power, resulting in less variability and hence smaller errors that are almost imperceptible. Therefore, the difference in the complexity of the execution environment can explain the observed variability in our results.

Finally, during the development phase, we employed the Energy Profiler tool in Android Studio to evaluate the effect on energy consumption for Android applications. In addition, Xcode presents a comparable solution for iOS and watchOS devices through its "Instruments" tool. These tools offer developers a comprehensive analysis of application performance, memory utilization, and energy consumption, providing them with the necessary information to identify any potential problems and enhance the performance of their applications. To our satisfaction, the results indicated that there was no adverse effect on energy utilization.

6.8 Discussion

The findings presented in the preceding section demonstrate the efficiency of the proposed protocol across a range of devices, including wearables and low-power devices. This section provides a qualitative comparison of the proposed solution with other existing protocols in the field, highlighting its unique features and advantages.

Table 6.4 provides a comparison of key features between our proposed protocol and two existing solutions, Bellrock and FedLoc. Our protocol outperforms both Bellrock [115] and FedLoc [116] in terms of ensuring data authenticity, providing anonymity protection, supporting revocation, adopting serverless authentication architecture, achieving beacon-independent localization, supporting wearables and IoT devices, and ensuring scalability. Our protocol also preserves privacy, which is an essential characteristic for any secure and privacy-preserving localization system.

Tab. 6.4: Comparison between our proposed protocol and two existing solutions, Bellrock and FedLoc

Characteristics	Our protocol	Bellrock [115]	FedLoc [116]
Ensures data authenticity	yes	no	no
Provides anonymity protection	yes	yes	no
Preserves privacy	yes	yes	yes
Supports revocation	yes	no	no
Adopts serverless authentication architecture	yes	no	no
Achieves beacon-independent localization	yes	no	yes
Supports wearables and IoT devices	yes	partial	yes
Ensures scalability	yes	no	yes

The protocol we propose excels at ensuring data authenticity by implementing a robust authentication mechanism that verifies the identity of users before accepting any location data. This supplementary security measure plays a critical role in safeguarding both the precision and integrity of the localization system by thwarting any attempts to generate counterfeit or tampered positioning data. On the other hand, both Bellrock and FedLoc lack a mechanism to authenticate the location data, making them vulnerable to tampering and impersonation attacks.

To protect user anonymity, the cryptographic scheme we suggest employs randomized user credentials with every transmission, thereby ensuring that the identities of users remain concealed. Bellrock similarly presents itself as a strong privacy-preserving option by leveraging various techniques to generate pseudo-anonymous identifiers that shield user identities. Conversely, FedLoc neglects the need for anonymity protection and, therefore, poses potential risks to user identity disclosure. The significance of anonymity in the context of localization systems cannot

be overstated, and in this respect, our protocol and Bellrock have a clear advantage over FedLoc. On the other hand, it is worth noting that all three schemes share the common goal of preserving the privacy of positioning data. This is a crucial feature for any secure and privacy-preserving localization system, as users' location data is often sensitive and must be protected against unauthorized access. All solutions provide measures to ensure that users' positioning data remains private, including encryption and secure communication protocols. By preserving privacy, these schemes enable users to benefit from the advantages of LBSs without sacrificing their privacy.

By adopting a serverless authentication architecture, the cryptographic protocol we introduce stands out from both Bellrock and FedLoc. Both Bellrock and FedLoc rely on a centralized server, which could be a potential vulnerability. In contrast, our protocol does not require a centralized server, reducing the risk of a single point of failure and making it more resistant to attacks, thus offering a more secure and reliable solution.

The installation and maintenance of numerous beacons for accurate localization can be a complex and expensive process in beacon-dependent solutions. Unlike beacon-dependent solutions, the protocol we advance and FedLoc offer a beacon-independent approach, eliminating the need for numerous physical beacons. This not only simplifies the infrastructure requirements but also increases the flexibility of deployment across various environments. Moreover, both our cryptographic scheme and FedLoc exhibit scalability, enabling effortless system expansion to accommodate more users and devices. However, the beacon-dependent approach of Bellrock could potentially limit its scalability and cost-effectiveness in larger deployments. In addition, the suggested protocol supports a wide range of devices and platforms, making it highly versatile and adaptable to different user needs. In contrast, Bellrock only supports Android smartphones, which may limit its applicability in certain contexts. FedLoc, on the other hand, also supports some IoT devices, but not as many as our approach. The ability to support numerous devices and platforms is a crucial characteristic for LBSs, as it allows for greater user adoption and flexibility. Therefore, our protocol's wider support for devices and platforms gives it an advantage over both Bellrock and FedLoc in terms of usability and accessibility.

Revocation is a crucial aspect of any secure localization protocol, as it enables the system to remove invalid or malicious users that may compromise the integrity of the data. This is a critical feature for ensuring the long-term security and reliability of the system and provides an added layer of protection against potential attacks. Both Bellrock and FedLoc do not consider user revocation, leaving the system vulnerable to potential threats from compromised or malicious users. Our proposed scheme is the only one among the three that takes user revocation into account, allowing for

the removal of any users who may pose a threat to the system's security.

Overall, the cryptographic scheme we put forth stands out as a comprehensive and advanced solution for LBSs, offering a range of robust security measures, a scalable architecture, broad device and platform support, and user revocation capabilities. These characteristics make our protocol an ideal choice for various applications where security and reliability are paramount.

6.9 Summary

This chapter has presented a novel approach to addressing privacy and security concerns in Collaborative Indoor Positioning Systems through the development of a decentralized authentication scheme.

By examining Attribute-based Authentication as a solution, we designed and implemented a decentralized scheme that utilizes encrypted and anonymized location information transmitted via Bluetooth Low Energy advertising. The authentication scheme provides robust privacy protection in a fully decentralized environment, with no reliance on centralized data sources. Our protocol has undergone extensive performance testing on a range of devices, including single-board computers, smartphones, smartwatches, and microcontrollers. We have demonstrated high throughput and low latency, with durations well under 350 milliseconds, even on the slowest devices.

To the best of our knowledge, this protocol represents the first fully decentralized ABA scheme running over BLE, providing a promising solution for protecting user privacy in CIPSs.

The contents of this chapter are based on the publications by Casanova-Marqués et al. [15] and Casanova-Marqués et al. [16].

7 Conclusion

This thesis aims to design and evaluate novel cryptographic technologies for the protection of privacy and the digital identity of electronic users, with a focus on attribute-based authentication in electronic systems and user authenticity in dynamic wearable architectures. Through our research, we have addressed several challenges, including the inefficient revocation of invalid users, the missing identification of malicious users, and low performance on constrained devices such as wearables. We have developed and tested new algorithms for these purposes and benchmarked their performance on existing wearable hardware devices, such as smart cards, smartwatches, and smartphones. In this chapter, we will summarize our key findings, answering each of the research questions posed in Chapter 1.

7.1 Answering the research questions

The research questions presented in Chapter 1 are addressed and answered in this section.

- *How can anonymous credential schemes be adapted to support user revocation while maintaining privacy?*
  Large-scale revocation of users in attribute-based authentication schemes introduces formidable challenges, particularly when seeking to preserve privacy on resource-constrained devices like smart cards. In response, Chapter 3 introduces a sophisticated attribute-based authentication scheme featuring pseudonymous-based revocation. This innovative design ensures that all user transactions remain non-traceable and non-linkable, safeguarding their privacy. In the event of malicious behavior, the revocation authority possesses the capability to compute all associated pseudonyms and promptly revoke their anonymity. The proposed protocol has been diligently implemented and rigorously tested on MULTOS smart cards, delivering highly promising outcomes in terms of both efficiency and security. This approach effectively addresses the intricate issue of user revocation within anonymous credential schemes, even in environments characterized by limited computational resources.
- *What strategies can be employed to enable attribute-based authentication protocols on smart cards with limited support for elliptic curve cryptography?*
  Smart cards, particularly those utilizing Java Card technology, frequently encounter constraints in terms of computational power and a lack of support for fundamental operations involving modular arithmetic and elliptic curves. Although these cards offer limited elliptic curve functionalities, their restricted

API inhibits the realization of their full potential. In response to this inquiry, Chapter 4 introduces a range of techniques that ingeniously transform mathematical operations to operate within the limitations of the Java Card API. These innovative transformations have enabled efficient implementation of the attribute-based authentication protocol, as outlined in Chapter 3. While the obtained results may not currently support real-world deployment, they pave the way for future investigations and advancements in this field.

- *What are the usability challenges associated with using anonymous credentials in various applications, and how can they be addressed?*
  Implementing attribute-based authentication schemes on smart cards poses inherent challenges due to the absence of a user interface. The inability to visualize requested attributes or provide consent for disclosure hinders the seamless integration of these schemes in real-world environments, limiting user privacy and security. In Chapter 5, we introduce a meticulously crafted platform tailored for real-world deployment. This platform utilizes the cryptographic core presented in Chapter 3 and seamlessly executes on mobile devices and smartwatches. By effectively resolving the aforementioned usability problems, it ensures a secure and streamlined user experience, bridging the gap between attribute-based authentication and practical application.

- *How can anonymous credential schemes be integrated into collaborative indoor positioning systems to enhance privacy and security?*
  The exploration of attribute-based authentication schemes beyond access control remains an active area of research. Collaborative indoor positioning systems face inherent vulnerabilities due to their interaction with unknown devices and the lack of cryptographic protocols to safeguard user privacy and security. In response to this challenge and recognizing the potential of attribute-based authentication, our research has yielded a decentralized cryptographic scheme tailored to enhance privacy and security within these systems. This innovative scheme presented in Chapter 6 focuses on providing complete anonymity through the randomization of identity in each information transmission and the encryption of transmitted data. Additionally, automatic user revocation adds an extra layer of privacy and security to the collaborative indoor positioning system. The scheme's deployment holds significant promise for fortifying user privacy and fostering trust among stakeholders.

- *How can anonymous credential schemes be implemented in resource-constrained environments, such as* IoT devices?
  The limitations in computational power and support for mathematical operations extend beyond smart cards and also affect IoT devices. To address

this research question, we successfully implemented the cryptographic scheme introduced in Chapter 6 on resource-constrained IoT devices, specifically Arduino boards. This implementation utilized specialized libraries tailored for IoT environments and optimized operating systems designed for low-power devices. Through extensive computation times and rigorous testing, our implementation showcased promising results, demonstrating notable improvements in both efficiency and security.

- *Are attribute-based authentication schemes suitable for ensuring user authenticity in dynamic wearable architectures?*
  Throughout the thesis, a diverse range of cryptographic schemes have been meticulously explored, targeting various contexts and environments. Extensive implementations have been carried out on a myriad of wearable devices, encompassing various operating systems and architectures. The achieved results have consistently demonstrated commendable efficiency, firmly establishing their practical viability for real-world deployment. Hence, in direct response to the research question, it unequivocally affirms the suitability of attribute-based authentication schemes for effectively ensuring user authenticity within dynamic wearable architectures.

7.2 Impact of the publications

This dissertation expands on some journal articles [16] and conference papers [11, 12, 13, 14, 15], including a paper awarded as one of the best papers. The award-winning paper was presented at the STUDENT EEICT 2022 conference, where it received the award for one of the best papers; it was expanded in Chapter 5.

In addition to the aforementioned publications, I have collaborated on several works related to this thesis. Even though these works are not explicitly described in this thesis,

- Hajny, Dzurenda, Casanova-Marqués, and Malina [128] tackles the issue of security and privacy protection in resource-constrained networks, presenting efficient cryptographic protocols for secure channel establishment and anonymous authentication. These protocols are designed to work on devices with limited cryptographic capabilities and demonstrate their effectiveness through benchmarks and real-world application integration.
- Ometov, Shubina, Klus, Skibińska, Saafi, Pascacio, Flueratoru, Gaibor, Chukhno, Chukhno, Ali, Channa, Svertoka, Qaim, Casanova-Marqués, Holcer, Torres-Sospedra, Casteleyn, Ruggeri, Araniti, Burget, Hosek, and Lohan [129] explores the evolution of wearable devices and offers a comprehensive review of

the current state of the wearable market. It provides an in-depth classification of wearables, covering factors such as wireless communication technologies, architectures, data processing, and market status. Additionally, the survey addresses various challenges in wearable technology and discusses existing and future solutions to overcome them.

- Dzurenda, Ricci, Casanova-Marqués, Hajny, and Cika [130] presents two efficient *Authenticated Key Agreement* (AKA) schemes based on elliptic curves, designed for implementation on constrained devices. The proposed schemes incorporate a proof of knowledge concept and secret sharing techniques, enabling secure communication channels and multi-device/multifactor authentication features with fast execution times, even on resource-limited devices.
- Ricci, Dzurenda, Casanova-Marqués, and Cika [131] proposes a novel (n, t)-threshold signature scheme that enhances security and privacy in Blockchain technology. The scheme enables the division of a Blockchain wallet into multiple devices, requiring a threshold number of devices for signing. This division enhances transaction security and enables anonymous signing on behalf of the user group sharing the wallet.

7.3 Future work

Throughout the thesis, our primary focus has been on the evolution and transition of technologies with the objective of enhancing accessibility and user privacy. Our exploration begins by delving into smart cards and progresses to an examination of contemporary devices such as smartwatches and smartphones. Consequently, further research is imperative to optimize and enhance *Anonymous Credential* (AC) schemes specifically tailored for mobile devices. Given the heightened computational power available on these devices, a unique opportunity presents itself to expand existing protocols and explore advanced approaches.

Regarding Collaborative Indoor Positioning Systems, this thesis introduces a groundbreaking decentralized protocol explicitly designed for implementation in real-world environments. However, despite this noteworthy advancement, there remains ample scope for improvement and optimization across various facets. For instance, it is of paramount importance to explore and develop more efficient and reliable revocation mechanisms that ensure the utmost security and privacy in CIPSs.

Bibliography

[1] Jan Hajny and Lukas Malina. Unlinkable Attribute-based Credentials with Practical Revocation on Smart Cards. In *Smart Card Research and Advanced Applications*, pages 62–76, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. doi: 10.1007/978-3-642-37288-9_5.

[2] Jan Camenisch, Stephan Krenn, Anja Lehmann, Gert Læssøe Mikkelsen, Gregory Neven, and Michael Østergaard Pedersen. Formal Treatment of Privacy-Enhancing Credential Systems. In *Selected Areas in Cryptography – SAC 2015*, pages 3–24, Cham, 2016. Springer International Publishing. doi: 10.1007/978-3-319-31301-6_1.

[3] Jan Camenisch, Manu Drijvers, and Jan Hajny. 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*, pages 123–133, Vienna Austria, October 2016. ACM. doi: 10.1145/2994620.2994625.

[4] David Chaum. Security without Identification: Transaction Systems to Make Big Brother Obsolete. *Communications of the ACM*, 28(10):1030–1044, October 1985. doi: 10.1145/4372.4373.

[5] Stefan A. Brands. *Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy.* MIT Press, Cambridge, Mass, 2000. ISBN 978-0-262-02491-4.

[6] Jan Camenisch and Anna Lysyanskaya. An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation. In *Advances in Cryptology – EUROCRYPT 2001*, pages 93–118, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg. doi: 10.1007/3-540-44987-6_7.

[7] Wojciech Mostowski and Pim Vullers. Efficient U-Prove Implementation for Anonymous Credentials on Smart Cards. In *Security and Privacy in Communication Networks*, pages 243–260, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. doi: 10.1007/978-3-642-31909-9_14.

[8] Pim Vullers and Gergely Alpár. Efficient Selective Disclosure on Smart Cards using Idemix. In *Policies and Research in Identity Management*, pages 53–67, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. doi: 10.1007/978-3-642-37282-7_5.

[9] Jan Camenisch, Manu Drijvers, Petr Dzurenda, and Jan Hajny. Fast Keyed-Verification Anonymous Credentials on Standard Smart Cards. In *ICT Systems Security and Privacy Protection*, pages 286–298, Cham, 2019. Springer International Publishing. doi: 10.1007/978-3-030-22312-0_20.

[10] Mouna S. Chebli, Heba Mohammad, and Khalifa Al Amer. An Overview of Wireless Indoor Positioning Systems: Techniques, Security, and Countermeasures. In *Internet and Distributed Computing Systems*, pages 223–233, Cham, 2019. Springer International Publishing. doi: 10.1007/978-3-030-34914-1_22.

[11] Jan Hajny, Petr Dzurenda, Raúl Casanova-Marqués, and Lukas Malina. Privacy ABCs: Now Ready for Your Wallets! In *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, pages 686–691, Kassel, Germany, March 2021. IEEE. doi: 10.1109/PerComWorkshops51409.2021.9431139.

[12] Raúl Casanova-Marqués, Petr Dzurenda, and Jan Hajny. Implementation of Revocable Keyed-Verification Anonymous Credentials on Java Card. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*, pages 1–8, Vienna Austria, August 2022. ACM. doi: 10.1145/3538969.3543798.

[13] Raúl Casanova-Marqués and Petr Dzurenda. Readiness of Anonymous Credentials for Real Environment Deployment. In *Proceedings II of the 28th Conference STUDENT EEICT 2022*, pages 308–312, Brno, Czech Republic, April 2022. Brno University of Technology, Faculty of Electrical Engineering and Communication. doi: 10.5281/zenodo.6623794.

[14] Petr Dzurenda, Raúl Casanova-Marqués, Lukas Malina, and Jan Hajny. Real-world Deployment of Privacy-Enhancing Authentication System using Attribute-based Credentials. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*, pages 1–9, Vienna, Austria, August 2022. ACM. doi: 10.1145/3538969.3543803.

[15] Raúl Casanova-Marqués, Pavel Pascacio, Jan Hajny, and Joaquín Torres-Sospedra. Anonymous Attribute-based Credentials in Collaborative Indoor Positioning Systems. In *Proceedings of the 18th International Conference on Security and Cryptography*, pages 791–797, Online Streaming, 2021. SCITEPRESS - Science and Technology Publications. doi: 10.5220/0010582507910797.

[16] Raúl Casanova-Marqués, Joaquín Torres-Sospedra, Jan Hajny, and Michael Gould. Maximizing Privacy and Security of Collaborative Indoor Positioning using Zero-knowledge Proofs. *Internet of Things*, 22(100801):1–18, July 2023. doi: 10.1016/j.iot.2023.100801.

[17] Whitfield Diffie and Martin Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, November 1976. doi: 10.1109/TIT.1976.1055638.

[18] Dan Boneh and Xavier Boyen. Short Signatures without Random Oracles and the SDH Assumption in Bilinear Groups. *Journal of Cryptology*, 21(2): 149–177, April 2008. doi: 10.1007/s00145-007-9005-7.

[19] Ronald John Fitzgerald Cramer. *Modular Design of Secure yet Practical Cryptographic Protocols*. PhD thesis, Universiteit van Amsterdam, 1996.

[20] Dan Boneh and Xavier Boyen. Short Signatures without Random Oracles. In *Advances in Cryptology - EUROCRYPT 2004*, pages 56–73, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg. doi: 10.1007/978-3-540-24676-3_4.

[21] Jan Camenisch and Markus Stadler. Proof Systems for General Statements about Discrete Logarithms. Technical report, ETH Zurich, 1997.

[22] Ronald Cramer, Ivan Damgård, and Philip MacKenzie. Efficient Zero-Knowledge Proofs of Knowledge without Intractability Assumptions. In *Public Key Cryptography*, pages 354–372, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg. doi: 10.1007/978-3-540-46588-1_24.

[23] Amos Fiat and Adi Shamir. How to Prove Yourself: Practical Solutions to Identification and Signature Problems. In *Advances in Cryptology – CRYPTO' 86*, pages 186–194, Berlin, Heidelberg, 1987. Springer Berlin Heidelberg. doi: 10.1007/3-540-47721-7_12.

[24] Antonio de la Piedra, Jaap-Henk Hoepman, and Pim Vullers. Towards a Full-Featured Implementation of Attribute-based Credentials on Smart Cards. In *Cryptology and Network Security*, pages 270–289, Cham, 2014. Springer International Publishing. doi: 10.1007/978-3-319-12280-9_18.

[25] Jan Camenisch and Anna Lysyanskaya. Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials. In *Advances in Cryptology – CRYPTO 2002*, pages 61–76, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg. doi: 10.1007/3-540-45708-9_5.

[26] Lan Nguyen. Accumulators from Bilinear Pairings and Applications. In *Topics in Cryptology – CT-RSA 2005*, pages 275–292, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. doi: 10.1007/978-3-540-30574-3_19.

[27] Patrick P. Tsang, Man Ho Au, Apu Kapadia, and Sean W. Smith. Blacklistable Anonymous Credentials: Blocking Misbehaving Users without TTPs. In *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pages 72–81, Alexandria Virginia USA, October 2007. ACM. doi: 10.1145/1315245.1315256.

[28] Jan Camenisch, Markulf Kohlweiss, and Claudio Soriente. Solving Revocation with Efficient Update of Anonymous Credentials. In *Security and Cryptography for Networks*, pages 454–471, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. doi: 10.1007/978-3-642-15317-4_28.

[29] Patrick P. Tsang, Apu Kapadia, Cory Cornelius, and Sean W. Smith. Nymble: Blocking Misbehaving Users in Anonymizing Networks. *IEEE Transactions on Dependable and Secure Computing*, 8(2):256–269, March 2011. doi: 10.1109/TDSC.2009.38.

[30] Jan Hajny, Petr Dzurenda, and Lukas Malina. Privacy-PAC: Privacy-Enhanced Physical Access Control. In *Proceedings of the 13th Workshop on Privacy in the Electronic Society*, pages 93–96, Scottsdale Arizona USA, November 2014. ACM. doi: 10.1145/2665943.2665969.

[31] Wouter Lueks, Gergely Alpár, Jaap-Henk Hoepman, and Pim Vullers. Fast Revocation of Attribute-based Credentials for Both Users and Verifiers. *Computers & Security*, 67:308–323, June 2017. doi: 10.1016/j.cose.2016.11.018.

[32] Eric R. Verheul. Practical Backward Unlinkable Revocation in FIDO, German e-ID, Idemix and U-Prove. Cryptology ePrint Archive, Paper 2016/217, 2016. URL `https://eprint.iacr.org/2016/217`.

[33] Mitsunari Shigeo. MCL Library. `https://github.com/herumi/mcl`, 2023.

[34] Petr Dzurenda, Sara Ricci, Jan Hajny, and Lukas Malina. Performance Analysis and Comparison of Different Elliptic Curves on Smart Cards. In *2017 15th Annual Conference on Privacy, Security and Trust (PST)*, pages 365–36509, Calgary, AB, August 2017. IEEE. doi: 10.1109/PST.2017.00050.

[35] OpenSSL Project Authors. OpenSSL: Cryptography and SSL/TLS Toolkit. `https://www.openssl.org`, 2023.

[36] MULTOS Limited. MULTOS SmartDeck Developer's Reference Manual. Technical Report v3.4.0.0, MULTOS, 2021. URL https://multos.com/wp-content/uploads/2021/06/MDG.pdf.

[37] MULTOS Limited. MDG: MULTOS Developer's Guide. Technical Report MAO-DOC-TEC-005 v1.43, MULTOS, 2021. URL https://multos.com/wp-content/uploads/2021/06/MDG.pdf.

[38] MULTOS Limited. MDRM: MULTOS Developer's Reference Manual. Technical Report MAO-DOC-TEC-006 v1.59, MULTOS, 2022. URL https://multos.com/wp-content/uploads/2022/11/MDRM.pdf.

[39] MULTOS Limited. C-API V2: MULTOS Standard C-API. Technical Report MAO-DOC-TEC-016 v2.3, MULTOS, 2021. URL https://multos.com/wp-content/uploads/2021/06/CAPIv2.pdf.

[40] MULTOS Limited. MUM: MULTOS Utility Manual. Technical Report MAO-DOC-TEC-017 v2.11.0, MULTOS, 2021. URL https://multos.com/wp-content/uploads/2021/06/MUM.pdf.

[41] Lukas Malina and Jan Hajny. Accelerated Modular Arithmetic for Low-Performance Devices. In *2011 34th International Conference on Telecommunications and Signal Processing (TSP)*, pages 131–135, Budapest, Hungary, 2011. IEEE. doi: 10.1109/TSP.2011.6043757.

[42] Lukas Malina and Jan Hajny. Efficient Modular Multiplication for Programmable Smart Cards. *Telecommunication Systems*, 55(4):491–498, April 2014. doi: 10.1007/s11235-013-9804-0.

[43] Lukas Malina, Petr Dzurenda, Jan Hajny, and Zdenek Martinasek. Assessment of Cryptography Support and Security on Programmable Smart Cards. In *2018 41st International Conference on Telecommunications and Signal Processing (TSP)*, pages 1–5, Athens, 2018. IEEE. doi: 10.1109/TSP.2018.8441334.

[44] Vasilios Mavroudis and Petr Svenda. Towards Low-level Cryptographic Primitives for Java Cards. *CoRR*, abs/1810.01662:1–16, 2018.

[45] Vasilios Mavroudis and Petr Svenda. JCMathLib: Wrapper Cryptographic Library for Transparent and Certifiable Java Card Applets. In *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 89–96, Genoa, Italy, 2020. IEEE. doi: 10.1109/EuroSPW51379.2020.00022.

[46] Patrik Bichsel, Jan Camenisch, Thomas Groß, and Victor Shoup. Anonymous Credentials on a Standard Java Card. In *Proceedings of the 16th ACM Conference on Computer and Communications Security*, page 600–610, Chicago, Illinois, USA, 2009. ACM Press. doi: 10.1145/1653662.1653734.

[47] Michaël Sterckx, Benedikt Gierlichs, Bart Preneel, and Ingrid Verbauwhede. Efficient Implementation of Anonymous Credentials on Java Card Smart Cards. In *2009 First IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 106–110, London, 2009. IEEE. doi: 10.1109/WIFS.2009.5386474.

[48] Jan Hajny, Petr Dzurenda, and Lukas Malina. Attribute-based Credentials with Cryptographic Collusion Prevention. *Security and Communication Networks*, 8(18):3836–3846, December 2015. doi: 10.1002/sec.1304.

[49] Petr Dzurenda, Jan Hajny, Lukas Malina, and Sara Ricci. Anonymous Credentials with Practical Revocation using Elliptic Curves. In *Proceedings of the 14th International Joint Conference on e-Business and Telecommunications*, pages 534–539, Madrid, Spain, 2017. SCITEPRESS - Science and Technology Publications. doi: 10.5220/0006467705340539.

[50] Jens Bender, Marc Fischlin, and Dennis Kügler. Security Analysis of the PACE Key-Agreement Protocol. In *Information Security*, pages 33–48, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. doi: 10.1007/978-3-642-04474-8_3.

[51] Federal Office for Information Security (BSI). Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token. Technical Guideline BSI-TR-03110, Federal Office for Information Security (BSI), 2008. URL `https://www.bsi.bund.de/dok/TR-03110-en`.

[52] Petr Svenda. Java Card Algorithm Test: Detailed Analysis of Cryptographic Smart Cards Running with Java Card Platform. `https://github.com/crocs-muni/JCAlgTest`, 2013.

[53] Paulo S. L. M. Barreto and Michael Naehrig. Pairing-Friendly Elliptic Curves of Prime Order. In *Selected Areas in Cryptography*, pages 319–331, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. doi: 10.1007/11693383_22.

[54] Hendrik Tews. OV-Chip 2.0 Hacker's Guide. `https://www.sos.cs.ru.nl/ovchip/`, 2010.

[55] Eric R. Verheul. Self-Blindable Credential Certificates from the Weil Pairing. In *Advances in Cryptology — ASIACRYPT 2001*, pages 533–551, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg. doi: 10.1007/3-540-45682-1_31.

[56] Giuseppe Persiano and Ivan Visconti. An Efficient and Usable Multi-show Non-transferable Anonymous Credential System. In *Financial Cryptography*, pages 196–211, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg. doi: 10.1007/978-3-540-27809-2_21.

[57] Melissa Chase, Sarah Meiklejohn, and Greg Zaverucha. Algebraic MACs and Keyed-Verification Anonymous Credentials. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 1205–1216, Scottsdale Arizona USA, 2014. ACM. doi: 10.1145/2660267.2660328.

[58] Ibou Sene, Abdoul Aziz Ciss, and Oumar Niang. I2PA: An Efficient ABC for IoT. *Cryptography*, 3(2):16, June 2019. doi: 10.3390/cryptography3020016.

[59] Jan Camenisch and Els Van Herreweghen. Design and Implementation of the Idemix Anonymous Credential System. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pages 21–30, Washington, DC, USA, 2002. ACM Press. doi: 10.1145/586110.586114.

[60] Christian Paquin and Greg Zaverucha. U-Prove Cryptographic Specification v1.1 (Revision 3). Technical Report 1.1, Microsoft Corporation, December 2013. URL `https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/U-Prove20Cryptographic20Specification20V1.1.pdf`.

[61] Fabian van den Broek, Brinda Hampiholi, and Bart Jacobs. Securely Derived Identity Credentials on Smart Phones via Self-enrolment. In *Security and Trust Management*, pages 106–121, Cham, 2016. Springer International Publishing. doi: 10.1007/978-3-319-46598-2_8.

[62] Gergely Alpár, Fabian van den Broek, Brinda Hampiholi, Bart Jacobs, Wouter Lueks, and Sietse Ringers. IRMA: Practical, Decentralized and Privacy-Friendly Identity Management using Smartphones. In *10th Workshop on Hot Topics in Privacy Enhancing Technologies (HotPETs 2017)*, pages 1–2, Minneapolis, USA, 2017. HotPETs. URL `https://petsymposium.org/2017/papers/hotpets/irma-hotpets.pdf`.

[63] Maria Papaioannou, Jose C Ribeiro, Valdemar Monteiro, Victor Sucasas, Georgios Mantas, and Jonathan Rodriguez. A Privacy-Preserving User Authentication Mechanism for Smart City Mobile Apps. In *2021 IEEE 26th International Workshop on Computer-Aided Modeling and Design of Communication Links and Networks (CAMAD)*, pages 1–5, Porto, Portugal, 2021. IEEE. doi: 10.1109/CAMAD52502.2021.9617809.

[64] Sietse Ringers, Eric Verheul, and Jaap-Henk Hoepman. An Efficient Self-Blindable Attribute-based Credential Scheme. In *Financial Cryptography and Data Security*, pages 3–20, Cham, 2017. Springer International Publishing. doi: 10.1007/978-3-319-70972-7_1.

[65] Christian Paquin. U-Prove Technology Overview v1.1 (Revision 2). Technical Report 1.1, Microsoft Corporation, April 2013. URL `https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/U-Prove20Technology20Overview20V1.120Revision202.pdf`.

[66] Jan Camenisch, Gregory Neven, and Markus Rückert. Fully Anonymous Attribute Tokens from Lattices. In *Security and Cryptography for Networks*, pages 57–75, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. doi: 10.1007/978-3-642-32928-9_4.

[67] Cecilia Boschini, Jan Camenisch, and Gregory Neven. Relaxed Lattice-based Signatures with Short Zero-Knowledge Proofs. In *Developments in Language Theory*, pages 3–22, Cham, 2018. Springer International Publishing. doi: 10.1007/978-3-319-99136-8_1.

[68] Rupeng Yang, Man Ho Au, Zhenfei Zhang, Qiuliang Xu, Zuoxia Yu, and William Whyte. Efficient Lattice-based Zero-Knowledge Arguments with Standard Soundness: Construction and Applications. In *Advances in Cryptology – CRYPTO 2019*, pages 147–175, Cham, 2019. Springer International Publishing. doi: 10.1007/978-3-030-26948-7_6.

[69] Alberto Sonnino, Mustafa Al-Bassam, Shehar Bano, Sarah Meiklejohn, and George Danezis. Coconut: Threshold Issuance Selective Disclosure Credentials with Applications to Distributed Ledgers. *CoRR*, abs/1802.07344(4):1–15, August 2018. doi: 10.48550/ARXIV.1802.07344.

[70] Kalpana Singh, Omar Dib, Clément Huyart, and Khalifa Toumi. A Novel Credential Protocol for Protecting Personal Attributes in Blockchain. *Computers & Electrical Engineering*, 83:106586, May 2020. doi: 10.1016/j.compeleceng.2020.106586.

[71] Petr Dzurenda, Carles Anglès Tafalla, Sara Ricci, and Lukas Malina. Privacy-Preserving Online Parking Based on Smart Contracts. In *The 16th International Conference on Availability, Reliability, and Security*, pages 1–10, Vienna, Austria, 2021. ACM. doi: 10.1145/3465481.3470058.

[72] Victor Sucasas, Georgios Mantas, Maria Papaioannou, and Jonathan Rodriguez. Attribute-based Pseudonymity for Privacy-Preserving Authentication

in Cloud Services. *IEEE Transactions on Cloud Computing*, 11(1):168–184, May 2021. doi: 10.1109/TCC.2021.3084538.

[73] Jose Maria de Fuentes, Lorena Gonzalez-Manzano, Agusti Solanas, and Fatbardh Veseli. Attribute-based Credentials for Privacy-Aware Smart Health Services in IoT-based Smart Cities. *Computer*, 51(7):44–53, July 2018. doi: 10.1109/MC.2018.3011042.

[74] Gregory Neven, Gianmarco Baldini, Jan Camenisch, and Ricardo Neisse. Privacy-Preserving Attribute-based Credentials in Cooperative Intelligent Transport Systems. In *2017 IEEE Vehicular Networking Conference (VNC)*, pages 131–138, Torino, 2017. IEEE. doi: 10.1109/VNC.2017.8275631.

[75] J. M. de Fuentes, L. González-Manzano, J. Serna-Olvera, and F. Veseli. Assessment of Attribute-based Credentials for Privacy-Preserving Road Traffic Services in Smart Cities. *Personal and Ubiquitous Computing*, 21(5):869–891, October 2017. doi: 10.1007/s00779-017-1057-6.

[76] Ludovic Rousseau. Middleware to Access a Smart Card using SCard API (PC/SC). `https://pcsclite.apdu.fr`, 2023.

[77] Ludovic Rousseau. Generic USB CCID and ICCD Driver for Unix Systems. `https://ccid.apdu.fr`, 2023.

[78] BlueZ Project. BlueZ: Official Linux Bluetooth Protocol Stack. `https://www.bluez.org`, 2022.

[79] tspspi. ANSI C99 JSON Serializer / Deserializer. `https://github.com/tspspi/libcjson`, 2019.

[80] Andy Green. Flexible and Lightweight Pure C Library for Implementing Modern Network Protocols. `https://github.com/warmcat/libwebsockets`, 2023.

[81] Jan Hajny, Petr Dzurenda, Sara Ricci, Lukas Malina, and Kamil Vrba. Performance Analysis of Pairing-based Elliptic Curve Cryptography on Constrained Devices. In *2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pages 1–5, Moscow, Russia, November 2018. IEEE. doi: 10.1109/ICUMT.2018.8631228.

[82] Ben Lynn. The Pairing-Based Cryptography (PBC) Library. `https://crypto.stanford.edu/pbc/`, 2018.

[83] MIRACL Ltd. Multiprecision Integer and Rational Arithmetic Cryptographic Library – the MIRACL Crypto SDK. `https://github.com/miracl/MIRACL`, 2019.

[84] Akira Kanaoka. TEPLA Elliptic Curve and Pairing Library. `https://github.com/TEPLA/tepla-library`, 2016.

[85] Diego de Freitas Aranha, Conrado Porto Lopes Gouvêa, Tobias Markmann, Riad S. Wahby, and K. Liao. RELIC is an Efficient LIbrary for Cryptography. `https://github.com/relic-toolkit/relic`, 2023.

[86] OpenJS Foundation. Open-source and Cross-platform JavaScript Runtime Environment. `https://nodejs.org`, 2023.

[87] Evan You. The Progressive JavaScript Framework. `https://vuejs.org`, 2023.

[88] Vuetify. Vue Component Framework. `https://vuetifyjs.com`, 2023.

[89] Yu Xianjia, Li Qingqing, Jorge Pena Queralta, Jukka Heikkonen, and Tomi Westerlund. Applications of UWB Networks and Positioning to Autonomous Robots and Industrial Systems. In *2021 10th Mediterranean Conference on Embedded Computing (MECO)*, pages 1–6, Budva, Montenegro, 2021. IEEE. doi: 10.1109/MECO52532.2021.9460266.

[90] Ivo Silva, Cristiano Pendao, Joaquin Torres-Sospedra, and Adriano Moreira. TrackInFactory: A Tight Coupling Particle Filter for Industrial Vehicle Tracking in Indoor Environments. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(7):4151–4162, July 2022. doi: 10.1109/TSMC.2021.3091987.

[91] Lingli Zhao and Xiaoqin Yu. Design and Development of Anti-Theft Tracking App based on Geofence. In *2021 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, pages 738–741, Dalian, China, 2021. IEEE. doi: 10.1109/IPEC51340.2021.9421332.

[92] Tanweer Alam, Abdirahman Ahmed Hadi, and Rayyan Qari Shahabuddin Najam. Designing and Implementing the People Tracking System in the Crowded Environment using Mobile Application for Smart Cities. *International Journal of System Assurance Engineering and Management*, 13(1):11–33, February 2022. doi: 10.1007/s13198-021-01277-7.

[93] Pavel Pascacio, Sven Casteleyn, Joaquín Torres-Sospedra, Elena Simona Lohan, and Jari Nurmi. Collaborative Indoor Positioning Systems: A Systematic Review. *Sensors*, 21(3):1002, February 2021. doi: 10.3390/s21031002.

[94] Rainer Mautz. Indoor Positioning Technologies. In *Geodätisch-geophysikalische Arbeiten in der Schweiz*, pages 1–129. ETH Zurich, 2012. doi: 10.3929/ETHZ-A-007313554.

[95] Jian Yang, Christian Poellabauer, Pramita Mitra, and Cynthia Neubecker. Beyond Beaconing: Emerging Applications and Challenges of BLE. *Ad Hoc Networks*, 97:102015, February 2020. doi: 10.1016/j.adhoc.2019.102015.

[96] Vivian Genaro Motti and Kelly Caine. Users' Privacy Concerns about Wearables. In *Financial Cryptography and Data Security*, pages 231–244. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015. doi: 10.1007/978-3-662-48051-9_17.

[97] Sandra Seubert and Carlos Becker. The Democratic Impact of Strengthening European Fundamental Rights in the Digital Age: The Example of Privacy Protection. *German Law Journal*, 22(1):31–44, January 2021. doi: 10.1017/glj.2020.101.

[98] Roger A. Grimes. *Hacking Multifactor Authentication*. Wiley, 1 edition, October 2020. ISBN 978-1-119-67235-7. doi: 10.1002/9781119672357.

[99] Francesco Malandrino, Carlo Borgiattino, Claudio Casetti, Carla-Fabiana Chiasserini, Marco Fiore, and Roberto Sadao. Verification and Inference of Positions in Vehicular Networks through Anonymous Beaconing. *IEEE Transactions on Mobile Computing*, 13(10):2415–2428, October 2014. doi: 10.1109/TMC.2013.2297925.

[100] Zhiquan Liu, Jianfeng Ma, Jian Weng, Feiran Huang, Yongdong Wu, Linfeng Wei, and Yuxian Li. LPPTE: A Lightweight Privacy-Preserving Trust Evaluation Scheme for Facilitating Distributed Data Fusion in Cooperative Vehicular Safety Applications. *Information Fusion*, 73:144–156, September 2021. doi: 10.1016/j.inffus.2021.03.003.

[101] Jiaqi Huang, Yi Qian, and Rose Qingyang Hu. A Privacy-Preserving Scheme for Location-Based Services in the Internet of Vehicles. *Journal of Communications and Information Networks*, 6(4):385–395, December 2021. doi: 10.23919/JCIN.2021.9663103.

[102] Ning Xi, Weihui Li, Lv Jing, and Jianfeng Ma. ZAMA: A ZKP-based Anonymous Mutual Authentication Scheme for the IoV. *IEEE Internet of Things Journal*, 9(22):22903–22913, November 2022. doi: 10.1109/JIOT.2022.3186921.

[103] Tao Peng, Qin Liu, Dacheng Meng, and Guojun Wang. Collaborative Trajectory Privacy Preserving Scheme in Location-Based Services. *Information Sciences*, 387:165–179, May 2017. doi: 10.1016/j.ins.2016.08.010.

[104] Kimmo Jarvinen, Helena Leppakoski, Elena-Simona Lohan, Philipp Richter, Thomas Schneider, Oleksandr Tkachenko, and Zheng Yang. PILOT: Practical Privacy-Preserving Indoor Localization using OuTsourcing. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 448–463, Stockholm, Sweden, June 2019. IEEE. doi: 10.1109/EuroSP.2019.00040.

[105] Ajay K. Gupta and Udai Shanker. OMCPR: Optimal Mobility Aware Cache Data Pre-fetching and Replacement Policy using Spatial K-Anonymity for LBS. *Wireless Personal Communications*, 114(2):949–973, September 2020. doi: 10.1007/s11277-020-07402-2.

[106] Viktoriia Shubina, Aleksandr Ometov, Sergey Andreev, Dragos Niculescu, and Elena Simona Lohan. Privacy versus Location Accuracy in Opportunistic Wearable Networks. In *2020 International Conference on Localization and GNSS (ICL-GNSS)*, pages 1–6, Tampere, Finland, June 2020. IEEE. doi: 10.1109/ICL-GNSS49876.2020.9115424.

[107] Jong Wook Kim, Kennedy Edemacu, Jong Seon Kim, Yon Dohn Chung, and Beakcheol Jang. A Survey of Differential Privacy-based Techniques and Their Applicability to Location-Based Services. *Computers & Security*, 111:102464, December 2021. doi: 10.1016/j.cose.2021.102464.

[108] Paolo Barsocchi, Antonello Calabrò, Antonino Crivello, Said Daoudagh, Francesco Furfari, Michele Girolami, and Eda Marchetti. COVID-19 and Privacy: Enhancing Indoor Localization Architectures towards Effective Social Distancing. *Array*, 9:100051, March 2021. doi: 10.1016/j.array.2020.100051.

[109] Bin Jiang, Jianqiang Li, Guanghui Yue, and Houbing Song. Differential Privacy for Industrial Internet of Things: Opportunities, Applications, and Challenges. *IEEE Internet of Things Journal*, 8(13):10430–10451, July 2021. doi: 10.1109/JIOT.2021.3057419.

[110] Viktoriia Shubina, Aleksandr Ometov, Anahid Basiri, and Elena Simona Lohan. Effectiveness Modelling of Digital Contact-Tracing Solutions for Tackling the COVID-19 Pandemic. *Journal of Navigation*, 74(4):853–886, July 2021. doi: 10.1017/S0373463321000175.

[111] Huijie Yang, Pandi Vijayakumar, Jian Shen, and Brij B. Gupta. A Location-Based Privacy-Preserving Oblivious Sharing Scheme for Indoor Navigation.

Future Generation Computer Systems, 137:42–52, December 2022. doi: 10.1016/j.future.2022.06.016.

[112] Bohan Li, Ruochen Liang, Wei Zhou, Hailian Yin, Han Gao, and Ken Cai. LBS Meets Blockchain: An Efficient Method with Security Preserving Trust in SAGIN. *IEEE Internet of Things Journal*, 9(8):5932–5942, April 2022. doi: 10.1109/JIOT.2021.3064357.

[113] Zhihua Hu, Yunzhi Li, Guosong Jiang, Rui Zhang, and Mande Xie. PriHorus: Privacy-Preserving RSS-based Indoor Positioning. In *ICC 2022 - IEEE International Conference on Communications*, pages 5627–5632, Seoul, Korea, Republic of, May 2022. IEEE. doi: 10.1109/ICC45855.2022.9839103.

[114] Jingtao Guo, Ivan Wang-Hei Ho, Yun Hou, and Zijian Li. FedPos: A Federated Transfer Learning Framework for CSI-based Wi-Fi Indoor Positioning. *IEEE Systems Journal*, pages 1–12, 2023. doi: 10.1109/JSYST.2022.3230425.

[115] Augustin Zidek, Shyam Tailor, and Robert Harle. Bellrock: Anonymous Proximity Beacons from Personal Devices. In *2018 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 1–10, Athens, 2018. IEEE. doi: 10.1109/PERCOM.2018.8444603.

[116] Feng Yin, Zhidi Lin, Qinglei Kong, Yue Xu, Deshi Li, Sergios Theodoridis, and Shuguang Robert Cui. FedLoc: Federated Learning Framework for Data-Driven Cooperative Localization and Location Data Processing. *IEEE Open Journal of Signal Processing*, 1:187–215, November 2020. doi: 10.1109/OJSP.2020.3036276.

[117] Lidia Pocero Fraile and Christos Koulamas. Design and Evaluation of an Indoor Positioning System based on Mobile Devices. In *2022 11th Mediterranean Conference on Embedded Computing (MECO)*, pages 1–6, Budva, Montenegro, June 2022. IEEE. doi: 10.1109/MECO55406.2022.9797091.

[118] Carmen Delgado, Lanfranco Zanzi, Xi Li, and Xavier Costa-Perez. OROS: Orchestrating ROS-driven Collaborative Connected Robots in Mission-Critical Operations. In *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 147–156, Belfast, United Kingdom, June 2022. IEEE. doi: 10.1109/WoWMoM54355.2022.00026.

[119] Pavel Pascacio, Joaquin Torres-Sospedra, Sven Casteleyn, and Elena Simona Lohan. A Collaborative Approach using Neural Networks for BLE-RSS

Lateration-based Indoor Positioning. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 01–09, Padua, Italy, July 2022. IEEE. doi: 10.1109/IJCNN55064.2022.9892484.

[120] Mun On Wong and Sanghoon Lee. Indoor Navigation and Information Sharing for Collaborative Fire Emergency Response with BIM and Multi-User Networking. *Automation in Construction*, 148:104781, April 2023. doi: 10.1016/j.autcon.2023.104781.

[121] Yann Collet. Lossless Compression Algorithm. `https://lz4.github.io/lz4/`, 2023.

[122] RIOT Community. RIOT OS: The Friendly Operating System for the Internet of Things. `https://riot-os.org/`, 2023.

[123] Emmanuel Baccelli, Cenk Gundogan, Oliver Hahm, Peter Kietzmann, Martine S. Lenders, Hauke Petersen, Kaspar Schleiser, Thomas C. Schmidt, and Matthias Wahlisch. RIOT: An Open Source Operating System for Low-End Embedded Devices in the IoT. *IEEE Internet of Things Journal*, 5(6):4428–4440, December 2018. doi: 10.1109/JIOT.2018.2815038.

[124] Greg Roelofs. A Massively Spiffy Yet Delicately Unobtrusive Compression Library. `https://www.zlib.net`, 2022.

[125] Tukaani Developers. Tukaani Project. `https://tukaani.org/xz/`, 2023.

[126] Bluetooth Special Interest Group. Bluetooth Core Specification v4.2. Specification, Bluetooth Special Interest Group, 2014. URL `https://www.bluetooth.com/specifications/specs/core-specification-4-2/`.

[127] Bluetooth Special Interest Group. Bluetooth Core Specification v5.0. Specification, Bluetooth Special Interest Group, 2016. URL `https://www.bluetooth.com/specifications/specs/core-specification-5-0/`.

[128] Jan Hajny, Petr Dzurenda, Raúl Casanova-Marqués, and Lukas Malina. Cryptographic Protocols for Confidentiality, Authenticity and Privacy on Constrained Devices. In *2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pages 87–92, Brno, Czech Republic, October 2020. IEEE. doi: 10.1109/ICUMT51630.2020.9222243.

[129] Aleksandr Ometov, Viktoriia Shubina, Lucie Klus, Justyna Skibińska, Salwa Saafi, Pavel Pascacio, Laura Flueratoru, Darwin Quezada Gaibor, Nadezhda

Chukhno, Olga Chukhno, Asad Ali, Asma Channa, Ekaterina Svertoka, Waleed Bin Qaim, Raúl Casanova-Marqués, Sylvia Holcer, Joaquín Torres-Sospedra, Sven Casteleyn, Giuseppe Ruggeri, Giuseppe Araniti, Radim Burget, Jiri Hosek, and Elena Simona Lohan. A Survey on Wearable Technology: History, State-of-the-Art and Current Challenges. *Computer Networks*, 193: 108074, July 2021. doi: 10.1016/j.comnet.2021.108074.

[130] Petr Dzurenda, Sara Ricci, Raúl Casanova-Marqués, Jan Hajny, and Petr Cika. Secret Sharing-based Authenticated Key Agreement Protocol. In *The 16th International Conference on Availability, Reliability and Security*, pages 1–10, Vienna Austria, August 2021. ACM. doi: 10.1145/3465481.3470057.

[131] Sara Ricci, Petr Dzurenda, Raúl Casanova-Marqués, and Petr Cika. Threshold Signature for Privacy-Preserving Blockchain. In *Business Process Management: Blockchain, Robotic Process Automation, and Central and Eastern Europe Forum*, pages 100–115, Cham, 2022. Springer International Publishing. doi: 10.1007/978-3-031-16168-1_7.

Symbols and abbreviations

AA	Anonymous Attribute
ABA	Attribute-based Authentication
ABC	Attribute-based Credential
ABS	Attribute-based Signature
AC	Anonymous Credential
AES	Advanced Encryption Standard
AKA	Authenticated Key Agreement
APDU	Application Protocol Data Unit
API	Application Programming Interface
AP	Access Point
BB	Boneh-Boyen
BIM	Building Information Modeling
BLE	Bluetooth Low Energy
BN254	Barreto-Naehrig 254-bit
BN	Barreto-Naehrig
C-ITS	Cooperative Intelligent Transport System
CBC	Cipher Block Chaining
CIPS	Collaborative Indoor Positioning System
CLI	Command Line Interface
COVID-19	Coronavirus Disease 2019
CPU	Central Processing Unit
CSI	Channel State Information
DH	Diffie-Hellman
DoS	Denial of Service

DP	Differential Privacy
DSA	Digital Signature Algorithm
e-ID	electronic ID
ECC	Elliptic Curve Cryptography
ECDH	Elliptic Curve Diffie-Hellman
EC	Elliptic Curve
EEPROM	Electrically Erasable Programmable Read-Only Memory
EU	European Union
GDPR	General Data Protection Regulation
GM	Generic Mapping
GPS	Global Positioning System
GUI	Graphical User Interface
I2PA	I Prove Possession of Attributes
ICT	Information and Communication Technology
Idemix	Identity Mixer
IEEE	Institute of Electrical and Electronics Engineers
IIoT	Industrial Internet of Things
IoT	Internet of Things
IoV	Internet of Vehicles
IPS	Indoor Positioning System
IRMA	I Reveal My Attributes
JCVM	Java Card Virtual Machine
JNI	Java Native Interface
k-NN	k-Nearest Neighbors
KVAC	Keyed-Verification Anonymous Credential

LBS	Location-based Service
LibreCIP	Libre Collaborative Indoor Positioning
LSB	Least Significant Bit
MAC	Message Authentication Code
MCU	Microcontroller Unit
MIRACL	Multiprecision Integer and Rational Arithmetic Cryptographic Library
MVC	Model-View-Controller
NDK	Native Development Kit
NFC	Near-Field Communication
NTP	Network Time Protocol
PACE	Password Authenticated Connection Establishment
PBC	Pairing-based Cryptography
PC/SC	Personal Computer / Smart Card
PEAS	Privacy-Enhancing Authentication System
PIN	Personal Identification Number
PQABC	Post-Quantum Attribute-based Credential
PQGS	Post-Quantum Group Signature
PQ	Post-Quantum
q-SDH	q-Strong Diffie-Hellman
RAM	Random Access Memory
RELIC	Efficient LIbrary for Cryptography
REST	REpresentational State Transfer
RKVAC	Revocable Keyed-Verification Anonymous Credential
ROM	Read-Only Memory

RSA	Rivest-Shamir-Adleman
RSS	Received Signal Strength
SDK	Software Development Kit
SHA-1	Secure Hash Algorithm 1
SHA-3	Secure Hash Algorithm 3
TCP	Transmission Control Protocol
TEPLA	University of Tsukuba Elliptic Curve and Pairing Library
TLS	Transport Layer Security
TTP	Trusted Third Party
UI	User Interface
US	United States
UUID	Universally Unique IDentifier
wBB	weak Boneh-Boyen
ZKP	Zero-knowledge Proof

A Formal security and privacy analysis

We demonstrate the security of our Revocable Keyed-Verification Anonymous Credential protocol by providing a detailed security and privacy analysis of the `Show` and `Verify` algorithms. The authentication mechanism relies on a combination of provably secure cryptographic primitives, including the efficient and secure wBB signature scheme [18] and the robust Sigma protocols [19]. The formal analysis of the security, privacy, and functionality properties comprises five propositions, each of which is supported by a corresponding proof to establish the fulfillment of said properties.

Proposition 1. *The proposed revocable keyed-verification anonymous credential protocol is existentially* ***unforgeable*** *under chosen-message attacks in the random oracle model assuming that the discrete logarithm problem is hard.*

Proof. This is based on the fact that our proposal is built on the wBB signature and its unforgeability is proven in [18]. □

Proposition 2. *The proposed revocable keyed-verification anonymous credential protocol is* ***complete****,* ***correct*** *and* ***sound****. I.e., valid authentications will always be verified correctly, and invalid ones will always fail verification.*

Proof. The *completeness* property is satisfied when the verifier correctly reconstructs the commitments t'_{verify}, t'_{revoke}, t'_{sig}, t'_{sigI}, and t'_{sigII}.

$$
\begin{aligned}
t'_{verify} &= \hat{\sigma}^{-ex_0} g_1^{s_v} \hat{\sigma}^{x_r s_{m_r}} \prod_{z \notin \mathcal{D}} \hat{\sigma}^{x_z s_{m_z}} \prod_{z \in \mathcal{D}} \hat{\sigma}^{-ex_z m_z} \\
&= \hat{\sigma}^{-ex_0} g_1^{\rho_v + e\rho} \hat{\sigma}^{x_r(\rho_{m_r} - e m_r)} \prod_{z \notin \mathcal{D}} \hat{\sigma}^{x_z(\rho_{m_z} - e m_z)} \prod_{z \in \mathcal{D}} \hat{\sigma}^{-ex_z m_z} \\
&= \sigma^{-e\rho x_0} g_1^{\rho_v + e\rho} \sigma^{x_r \rho(\rho_{m_r} - e m_r)} \prod_{z \notin \mathcal{D}} \sigma^{x_z \rho(\rho_{m_z} - e m_z)} \prod_{z \in \mathcal{D}} \sigma^{-ex_z m_z \rho} \\
&= \sigma^{-e\rho x_0} g_1^{\rho_v + e\rho} \sigma^{x_r \rho_{m_r} \rho - e m_r x_r \rho} \prod_{z \notin \mathcal{D}} \sigma^{x_z \rho_{m_z} \rho - e x_z m_z \rho} \prod_{z \in \mathcal{D}} \sigma^{-ex_z m_z \rho} \\
&= \sigma^{-e\rho x_0} g_1^{\rho_v} g_1^{e\rho} \sigma^{x_r \rho_{m_r} \rho} \sigma^{-e m_r x_r \rho} \prod_{z \notin \mathcal{D}} \sigma^{x_z \rho_{m_z} \rho} \sigma^{-e x_z m_z \rho} \prod_{z \in \mathcal{D}} \sigma^{-ex_z m_z \rho} \\
&= g_1^{\rho_v} \sigma_{x_r}^{\rho_{m_r} \rho} \sigma^{-e\rho x_0} g_1^{e\rho} \sigma^{-e m_r x_r \rho} \sigma^{-e x_z m_z \rho} \prod_{z \notin \mathcal{D}} \sigma^{x_z \rho_{m_z} \rho} \\
&= g_1^{\rho_v} \sigma_{x_r}^{\rho_{m_r} \rho} \sigma^{-e\rho x_0 - e m_r x_r \rho - e x_z m_z \rho} g_1^{e\rho} \prod_{z \notin \mathcal{D}} \sigma_{x_z}^{\rho_{m_z} \rho} \\
&= g_1^{\rho_v} \sigma_{x_r}^{\rho_{m_r} \rho} \sigma^{-e\rho(x_0 + m_r x_r + x_z m_z} g_1^{e\rho} \prod_{z \notin \mathcal{D}} \sigma_{x_z}^{\rho_{m_z} \rho} \\
&= g_1^{\rho_v} \sigma_{x_r}^{\rho_{m_r} \rho} g_1^{\frac{-e\rho(x_0 + m_r x_r + x_z m_z)}{x_0 + m_r x_r + x_z m_z}} g_1^{e\rho} \prod_{z \notin \mathcal{D}} \sigma_{x_z}^{\rho_{m_z} \rho} \\
&= g_1^{\rho_v} \sigma_{x_r}^{\rho_{m_r} \rho} g_1^{e\rho - e\rho} \prod_{z \notin \mathcal{D}} \sigma_{x_z}^{\rho_{m_z} \rho} = g_1^{\rho_v} \sigma_{x_r}^{\rho_{m_r} \rho} \prod_{z \notin \mathcal{D}} \sigma_{x_z}^{\rho_{m_z} \rho} = t_{verify}
\end{aligned}
$$

$$
\begin{aligned}
t'_{revoke} &= (g_1 C^{-\mathcal{H}(epoch)})^{-e} C^{s_{m_r}} C^{s_i} = g_1^{-e} C^{e\mathcal{H}(epoch)} C^{\rho_{m_r} - em_r} C^{\rho_i + ei} \\
&= g_1^{-e} C^{e\mathcal{H}(epoch)} C^{\rho_{m_r}} C^{-em_r} C^{\rho_i} C^{ei} = C^{\rho_{m_r}} C^{\rho_i} C^{e\mathcal{H}(epoch) - em_r + ei} g_1^{-e} \\
&= C^{\rho_{m_r}} C^{\rho_i} g_1^{\frac{e(i - m_r + \mathcal{H}(epoch))}{i - m_r + \mathcal{H}(epoch)}} g_1^{-e} = C^{\rho_{m_r}} C^{\rho_i} g_1^{e-e} \\
&= C^{\rho_{m_r}} C^{\rho_i} = t_{revoke}
\end{aligned}
$$

$$
\begin{aligned}
t'_{sig} &= g_1^{s_i} h_1^{s_{e_I}} h_2^{s_{e_{II}}} = g_1^{\rho_i + ei} h_1^{\rho_{e_I} - ee_I} h_2^{\rho_{e_{II}} - ee_{II}} = g_1^{\rho_i} g_1^{ei} h_1^{\rho_{e_I}} h_1^{-ee_I} h_2^{\rho_{e_{II}}} h_2^{-ee_{II}} \\
&= g_1^{\rho_i} h_1^{\rho_{e_I}} h_2^{\rho_{e_{II}}} g_1^{e(e_I\alpha_1 + e_{II}\alpha_2)} g_1^{-ee_I\alpha_1} g_1^{-ee_{II}\alpha_2} = g_1^{\rho_i} h_1^{\rho_{e_I}} h_2^{\rho_{e_{II}}} g_1^{ee_I\alpha_1 + ee_{II}\alpha_2 - ee_I\alpha_1 - ee_{II}\alpha_2} \\
&= g_1^{\rho_i} h_1^{\rho_{e_I}} h_2^{\rho_{e_{II}}} = t_{sig}
\end{aligned}
$$

$$
\begin{aligned}
t'_{sigI} &= g_1^{s_v} \hat{\sigma}_{e_I}^{s_{e_I}} \bar{\sigma}_{e_I}^{-e} = g_1^{\rho_v + e\rho} \hat{\sigma}_{e_I}^{\rho_{e_I} - ee_I} \bar{\sigma}_{e_I}^{-e} = g_1^{\rho_v + e\rho} \hat{\sigma}_{e_I}^{\rho_{e_I} - ee_I} (\hat{\sigma}_{e_I}^{-e_I} g_1^{\rho})^{-e} \\
&= g_1^{\rho_v} g_1^{e\rho} \hat{\sigma}_{e_I}^{\rho_{e_I}} \hat{\sigma}_{e_I}^{-ee_I} \hat{\sigma}_{e_I}^{ee_I} g_1^{-e\rho} = g_1^{\rho_v} \hat{\sigma}_{e_I}^{\rho_{e_I}} g_1^{e\rho - e\rho} \hat{\sigma}_{e_I}^{ee_I - ee_I} = g_1^{\rho_v} \hat{\sigma}_{e_I}^{\rho_{e_I}} = t_{sigI}
\end{aligned}
$$

$$
\begin{aligned}
t'_{sigII} &= g_1^{s_v} \hat{\sigma}_{e_{II}}^{s_{e_{II}}} \bar{\sigma}_{e_{II}}^{-e} = g_1^{\rho_v + e\rho} \hat{\sigma}_{e_{II}}^{\rho_{e_{II}} - ee_{II}} \bar{\sigma}_{e_{II}}^{-e} = g_1^{\rho_v + e\rho} \hat{\sigma}_{e_{II}}^{\rho_{e_{II}} - ee_{II}} (\hat{\sigma}_{e_{II}}^{-e_{II}} g_1^{\rho})^{-e} \\
&= g_1^{\rho_v} g_1^{e\rho} \hat{\sigma}_{e_{II}}^{\rho_{e_{II}}} \hat{\sigma}_{e_{II}}^{-ee_{II}} \hat{\sigma}_{e_{II}}^{ee_{II}} g_1^{-e\rho} = g_1^{\rho_v} \hat{\sigma}_{e_{II}}^{\rho_{e_{II}}} g_1^{e\rho - e\rho} \hat{\sigma}_{e_{II}}^{ee_{II} - ee_{II}} = g_1^{\rho_v} \hat{\sigma}_{e_{II}}^{\rho_{e_{II}}} = t_{sigII}
\end{aligned}
$$

Two conditions must be met to prove the *correctness* of the protocol:

1. the challenge e computed by the verifier is identical to the one the user calculated.
2. the pairing computations determine that $\mathbf{e}(\bar{\sigma}_{e_I}, g_2) = \mathbf{e}(\hat{\sigma}_{e_I}, pk_{RA})$ in $\mathbb{G}_T$, and $\mathbf{e}(\bar{\sigma}_{e_{II}}, g_2) = \mathbf{e}(\hat{\sigma}_{e_{II}}, pk_{RA})$ in $\mathbb{G}_T$.

$$
\begin{aligned}
&\mathbf{e}(\bar{\sigma}_{e_I}, g_2) = \mathbf{e}(\hat{\sigma}_{e_I}, pk_{RA}) \rightarrow \mathbf{e}(\hat{\sigma}_{e_I}^{-e_I} g_1^{\rho}, g_2) = \mathbf{e}(\hat{\sigma}_{e_I}, g_2^{sk_{RA}}) \rightarrow \\
&\quad \mathbf{e}(\sigma_{e_I}^{-e_I\rho} g_1^{\rho}, g_2) = \mathbf{e}(\sigma_{e_I}^{\rho}, g_2^{sk_{RA}}) \rightarrow \mathbf{e}(g_1^{\frac{-e_I\rho}{e_I + sk_{RA}}} g_1^{\rho}, g_2) = \mathbf{e}(g_1^{\frac{\rho}{e_I + sk_{RA}}}, g_2^{sk_{RA}}) \rightarrow \\
&\quad \mathbf{e}(g_1^{\frac{-e_I\rho + \rho(e_I + sk_{RA})}{e_I + sk_{RA}}}, g_2) = \mathbf{e}(g_1^{\frac{\rho}{e_I + sk_{RA}}}, g_2^{sk_{RA}}) \rightarrow \\
&\quad \mathbf{e}(g_1^{\frac{-e_I\rho + e_I\rho + \rho sk_{RA}}{e_I + sk_{RA}}}, g_2) = \mathbf{e}(g_1^{\frac{\rho}{e_I + sk_{RA}}}, g_2^{sk_{RA}}) \rightarrow \\
&\quad \mathbf{e}(g_1, g_2)^{\frac{\rho sk_{RA}}{e_I + sk_{RA}}} = \mathbf{e}(g_1, g_2)^{\frac{\rho sk_{RA}}{e_I + sk_{RA}}}
\end{aligned}
$$

$$
\begin{aligned}
&\mathbf{e}(\bar{\sigma}_{e_{II}}, g_2) = \mathbf{e}(\hat{\sigma}_{e_{II}}, pk_{RA}) \rightarrow \mathbf{e}(\hat{\sigma}_{e_{II}}^{-e_{II}} g_1^{\rho}, g_2) = \mathbf{e}(\hat{\sigma}_{e_{II}}, g_2^{sk_{RA}}) \rightarrow \\
&\quad \mathbf{e}(\sigma_{e_{II}}^{-e_{II}\rho} g_1^{\rho}, g_2) = \mathbf{e}(\sigma_{e_{II}}^{\rho}, g_2^{sk_{RA}}) \rightarrow \mathbf{e}(g_1^{\frac{-e_{II}\rho}{e_{II} + sk_{RA}}} g_1^{\rho}, g_2) = \mathbf{e}(g_1^{\frac{\rho}{e_{II} + sk_{RA}}}, g_2^{sk_{RA}}) \rightarrow \\
&\quad \mathbf{e}(g_1^{\frac{-e_{II}\rho + \rho(e_{II} + sk_{RA})}{e_{II} + sk_{RA}}}, g_2) = \mathbf{e}(g_1^{\frac{\rho}{e_{II} + sk_{RA}}}, g_2^{sk_{RA}}) \rightarrow \\
&\quad \mathbf{e}(g_1^{\frac{-e_{II}\rho + e_{II}\rho + \rho sk_{RA}}{e_{II} + sk_{RA}}}, g_2) = \mathbf{e}(g_1^{\frac{\rho}{e_{II} + sk_{RA}}}, g_2^{sk_{RA}}) \rightarrow \\
&\quad \mathbf{e}(g_1, g_2)^{\frac{\rho sk_{RA}}{e_{II} + sk_{RA}}} = \mathbf{e}(g_1, g_2)^{\frac{\rho sk_{RA}}{e_{II} + sk_{RA}}}
\end{aligned}
$$

The proposed authentication technique is built on the foundation of zero-knowledge proofs, and its *soundness* is demonstrated through the construction of a knowledge extractor. This extractor employs the well-established rewinding technique described in[21]. □

Proposition 3. *The proposed revocable keyed-verification anonymous credential protocol is* ***zero-knowledge****. I.e., there exists a simulator* $\mathbb{S}$ *that can efficiently generate a protocol transcript that is indistinguishable from a real protocol transcript.*

Proof. To prove the *zero-knowledge* property, we construct a simulator $\mathbb{S}$ that generates a simulated protocol transcript closely resembling a real protocol execution. Let `RealTranscript` denote the distribution of protocol transcripts resulting from the real execution of the revocable anonymous credential protocol, i.e., C, $(\hat{\sigma}, \hat{\sigma}_{e_I}, \hat{\sigma}_{e_{II}}, \bar{\sigma}_{e_I}, \bar{\sigma}_{e_{II}})$, and $(e, s_{m_{z \notin \mathcal{D}}}, s_v, s_{m_r}, s_i, s_{e_I}, s_{e_{II}})$. Let `SimulatedTranscript` denote the distribution of protocol transcripts generated by running the simulator $\mathbb{S}$, i.e., $\tilde{C}$, $(\tilde{\sigma}, \tilde{\sigma}_{e_I}, \tilde{\sigma}_{e_{II}}, \tilde{\sigma}'_{e_I}, \tilde{\sigma}'_{e_{II}})$, and $(\tilde{e}, \tilde{s}_{m_{z \notin \mathcal{D}}}, \tilde{s}_v, \tilde{s}_{m_r}, \tilde{s}_i, \tilde{s}_{e_I}, \tilde{s}_{e_{II}})$. We assume the existence of an efficient adversary `A` that tries to distinguish between `RealTranscript` and `SimulatedTranscript`. `A` has access to both the real protocol transcripts obtained from actual executions of the protocol and the simulated protocol transcripts generated by the simulator $\mathbb{S}$. Furthermore, `A` can interact with the protocol, submit inputs, receive outputs, and make queries to relevant components, such as the random oracle. It can perform computations in a reasonable amount of time based on the security parameter κ. However, `A` cannot break cryptographic assumptions or solve computationally hard problems beyond its computational limits.

The simulator simulates the operations performed by the revocation authority and the issuer to generate consistent credentials. The simulator undergoes an initialization phase where it obtains the necessary values, such as $\ddot{m}_r$, $(\ddot{e}_1, \ddot{\sigma}_{e_1}), \ldots, (\ddot{e}_k, \ddot{\sigma}_{e_k})$, $\ddot{\sigma}$, $\ddot{\sigma}_{x_1}, \ldots, \ddot{\sigma}_{x_n}$, and $\ddot{\sigma}_{x_r}$. Then, it executes the following steps:

1. inputs *nonce*, *epoch*, and $\mathcal{D}$,
2. consistently selects the per-session value $\tilde{\imath} = \ddot{e}_I \alpha_1 + \ddot{e}_{II} \alpha_2$,
3. consistently generates the pseudonym $\tilde{C} = g_1^{\frac{1}{\tilde{\imath} - \ddot{m}_r + \mathcal{H}(epoch)}}$,
4. consistently computes the randomized credentials $\tilde{\sigma} = \ddot{\sigma}^{\rho}$, $\tilde{\sigma}_{e_I} = \ddot{\sigma}_{e_I}^{\rho}$, $\tilde{\sigma}_{e_{II}} = \ddot{\sigma}_{e_{II}}^{\rho}$, $\tilde{\sigma}'_{e_I} = \tilde{\sigma}_{e_I}^{-\ddot{e}_I} g_1^{\rho}$, and $\tilde{\sigma}'_{e_{II}} = \tilde{\sigma}_{e_{II}}^{-\ddot{e}_{II}} g_1^{\rho}$,
5. consistently computes the commitments $\tilde{t}_{verify} = g_1^{\rho_v} \ddot{\sigma}_{x_r}^{\rho_{m_r} \rho} \prod_{z \notin \mathcal{D}} \ddot{\sigma}_{x_z}^{\rho_{m_z} \rho}$, $\tilde{t}_{revoke} = \tilde{C}^{\rho_{m_r}} \tilde{C}^{\rho_i}$, $\tilde{t}_{sig} = g_1^{\rho_i} h_1^{\rho_{e_I}} h_2^{\rho_{e_{II}}}$, $\tilde{t}_{sigI} = g_1^{\rho_v} \tilde{\sigma}_{e_I}^{\rho_{e_I}}$, and $\tilde{t}_{sigII} = g_1^{\rho_v} \tilde{\sigma}_{e_{II}}^{\rho_{e_{II}}}$,
6. consistently computes the challenge $\tilde{e} = \mathcal{H}(\tilde{t}_{verify}, \tilde{t}_{revoke}, \tilde{t}_{sig}, \tilde{t}_{sigI}, \tilde{t}_{sigII}, \tilde{\sigma}, \tilde{\sigma}_{e_I}, \tilde{\sigma}'_{e_I}, \tilde{\sigma}_{e_{II}}, \tilde{\sigma}'_{e_{II}}, \tilde{C}, nonce)$,
7. consistently computes the responses $\langle \tilde{s}_{m_z} = \rho_{m_z} - \tilde{e} m_z \rangle_{z \notin \mathcal{D}}$, $\tilde{s}_v = \rho_v + \tilde{e}\rho$, $\tilde{s}_{m_r} = \rho_{m_r} - \tilde{e}\ddot{m}_r$, $\tilde{s}_i = \rho_i + \tilde{e}\tilde{\imath}$, $\tilde{s}_{e_I} = \rho_{e_I} - \tilde{e}\ddot{e}_I$, and $\tilde{s}_{e_{II}} = \rho_{e_{II}} - \tilde{e}\ddot{e}_{II}$,
8. outputs $\tilde{C}$, $(\tilde{\sigma}, \tilde{\sigma}_{e_I}, \tilde{\sigma}_{e_{II}}, \tilde{\sigma}'_{e_I}, \tilde{\sigma}'_{e_{II}})$, and $(\tilde{e}, \tilde{s}_{m_{z \notin \mathcal{D}}}, \tilde{s}_v, \tilde{s}_{m_r}, \tilde{s}_i, \tilde{s}_{e_I}, \tilde{s}_{e_{II}})$.

The proof outlines the steps executed by the simulator $\mathbb{S}$ to generate a simulated protocol transcript. Although the adversary can compute commitments $\tilde{t}_{revoke}$, $\tilde{t}_{sig}$, $\tilde{t}_{sigI}$, and $\tilde{t}_{sigII}$, it lacks the necessary secret keys of the issuer or verifier to compute $\tilde{t}_{verify}$. As a result, in both real and simulated transcripts, the adversary is unable to

verify the value of $\tilde{e}$. However, utilizing the public key of the revocation authority, the adversary can attempt pairing verification. Despite the failure of the pairing verification process, the resulting protocol transcript is indistinguishable from that of a malicious user. Therefore, the simulator produces output that demonstrates computational indistinguishability when compared to the real protocol transcript. □

Proposition 4. *The proposed revocable keyed-verification anonymous credential protocol is* ***key-parameter consistent****. I.e., generated keys and parameters conform to the defined rules, specifically by ensuring that the public keys are valid and securely paired with their corresponding private keys.*

Proof. The protocol produces the key pair (sk_{RA}, pk_{RA}). To establish key-parameter consistency, for every valid secret key sk_{RA}, the corresponding public key pk_{RA} has to be computed correctly and satisfy the defined rules. Given a valid secret key $sk_{RA} \in_R \mathbb{Z}_q$, the public key $pk_{RA} = g_2^{sk_{RA}}$ is computed correctly using the generator g_2 and the secret component sk_{RA}. This computation adheres to the defined rules, as it follows the specified procedure for deriving the public key. □

Proposition 5. *The proposed revocable keyed-verification anonymous credential protocol provides* ***anonymity****,* ***unlinkability****, and* ***untraceability****.*

Proof. Due to the proof of knowledge protocol's *zero-knowledge* property, the proof π is always *anonymous*, *unlinkable*, and *untraceable*. The distribution of $\hat{\sigma}$, $\hat{\sigma}_{e_I}$, $\hat{\sigma}_{e_{II}}$, $\bar{\sigma}_{e_I}$, and $\bar{\sigma}_{e_{II}}$ is uniform and random in $\mathbb{Z}_q$ since ρ is selected uniformly and randomly from $\mathbb{Z}_q$. Consequently, the disclosed values are indistinguishable from random elements. Please note that in the case of C, the protocol can only guarantee these properties when the number of pseudonyms remains below or equal to $max_{sessions} = k^j$. Here, $max_{sessions}$ represents the maximum number of combinations that the unique per-session value i can accept within the same epoch. □

B Formal security and privacy analysis

We demonstrate the security of our decentralized Attribute-based Authentication protocol by providing a detailed security and privacy analysis of the `Show` and `Verify` algorithms. The authentication mechanism relies on a combination of provably secure cryptographic primitives, including the widely-used DH key exchange [17], the efficient and secure wBB signature scheme [18], and the robust Sigma protocols [19]. The formal analysis of the security, privacy, and functionality properties comprises six propositions, each of which is supported by a corresponding proof to establish the fulfillment of said properties.

Proposition 6. *The proposed decentralized attribute-based authentication protocol is existentially* ***unforgeable*** *under chosen-message attacks in the random oracle model assuming that the discrete logarithm problem is hard.*

Proof. This is based on the fact that our proposal is built on the wBB signature and its unforgeability is proven in [18]. □

Proposition 7. *The proposed decentralized attribute-based authentication protocol is* ***complete****,* ***correct*** *and* ***sound****. I.e., valid authentications will always be verified correctly, and invalid ones will always fail verification.*

Proof. The *completeness* property is satisfied when the verifier correctly reconstructs the commitment t'_κ.

$$
\begin{aligned}
t'_\kappa &= g_1^{s_\kappa}\hat{\sigma}_{x_0}^{-e}\hat{\sigma}_{x_r}^{-em_r}\hat{\sigma}_{x_{ID}}^{s_{ID}}\mathcal{R}^{i_r} = g_1^{\rho_\kappa+e\rho}\sigma_{x_0}^{-e\rho}\sigma_{x_r}^{-e\rho m_r}\sigma_{x_{ID}}^{\rho_{ID}\rho-e\rho m_{ID}}\mathcal{R}^{i_r} \\
&= g_1^{\rho_\kappa}\sigma_{x_{ID}}^{\rho_{ID}\rho}\mathcal{R}^{i_r}g_1^{e\rho}\sigma^{-e\rho x_0}\sigma^{-e\rho m_r x_r}\sigma^{-e\rho m_{ID}x_{ID}} = g_1^{\rho_\kappa}\sigma_{x_{ID}}^{\rho_{ID}\rho}\mathcal{R}^{i_r}g_1^{e\rho}g_1^{\frac{-e\rho(x_0+m_r x_r+m_{ID}x_{ID})}{x_0+m_r x_r+m_{ID}x_{ID}}} \\
&= g_1^{\rho_\kappa}\sigma_{x_{ID}}^{\rho_{ID}\rho}\mathcal{R}^{i_r}g_1^{e\rho-e\rho} = g_1^{\rho_\kappa}\sigma_{x_{ID}}^{\rho_{ID}\rho}\mathcal{R}^{i_r} = t_\kappa
\end{aligned}
$$

Three conditions must be met to prove the *correctness* of the protocol:

1. the verifier can reconstruct the decryption key $\mathcal{H}(t'_\kappa)$. This condition is always fulfilled due to the completeness property.
2. the challenge e computed by the verifier is identical to the one the user calculated.
3. the pairing computation determines that $\mathbf{e}(\hat{\sigma}_{x_0}, g_2) = \mathbf{e}(\hat{\sigma}, \mathcal{X}_0)$ in $\mathbb{G}_T$.

$$
\begin{aligned}
&\mathbf{e}(\hat{\sigma}_{x_0}, g_2) = \mathbf{e}(\hat{\sigma}, \mathcal{X}_0) \rightarrow \mathbf{e}(\sigma_{x_0}^{\rho}, g_2) = \mathbf{e}(\sigma^{\rho}, \mathcal{X}_0) \rightarrow \mathbf{e}(\sigma^{\rho x_0}, g_2) = \mathbf{e}(\sigma^{\rho}, \mathcal{X}_0) \rightarrow \\
&\quad \mathbf{e}(g_1^{\frac{\rho x_0}{x_0+m_r x_r+m_{ID}x_{ID}}}, g_2) = \mathbf{e}(g_1^{\frac{\rho}{x_0+m_r x_r+m_{ID}x_{ID}}}, g_2^{x_0}) \rightarrow \\
&\quad \mathbf{e}(g_1, g_2)^{\frac{\rho x_0}{x_0+m_r x_r+m_{ID}x_{ID}}} = \mathbf{e}(g_1, g_2)^{\frac{\rho x_0}{x_0+m_r x_r+m_{ID}x_{ID}}}
\end{aligned}
$$

The proposed authentication technique is built on the foundation of zero-knowledge proofs, and its *soundness* is demonstrated through the construction of a knowledge extractor. This extractor employs the well-established rewinding technique described in[21]. □

Proposition 8. *The proposed decentralized attribute-based authentication protocol is* ***zero-knowledge****. I.e., there exists a simulator* $\mathbb{S}$ *that can efficiently generate a protocol transcript that is indistinguishable from a real protocol transcript.*

Proof. To prove the *zero-knowledge* property, we construct a simulator $\mathbb{S}$ that generates a simulated protocol transcript closely resembling a real protocol execution. Let `RealTranscript` denote the distribution of protocol transcripts resulting from the real execution of the decentralized attribute-based authentication protocol, i.e., $\mathcal{R}$, $(\hat{\sigma}, \hat{\sigma}_{x_0}, \hat{\sigma}_{x_r}, \hat{\sigma}_{x_{ID}})$, and (e, s_κ, s_{ID}). Let `SimulatedTranscript` denote the distribution of protocol transcripts generated by running the simulator $\mathbb{S}$, i.e., $\tilde{\mathcal{R}}$, $(\tilde{\sigma}, \tilde{\sigma}_{x_0}, \tilde{\sigma}_{x_r}, \tilde{\sigma}_{x_{ID}})$, and $(\tilde{e}, \tilde{s}_\kappa, \tilde{s}_{ID})$. We assume the existence of an efficient adversary `A` that tries to distinguish between `RealTranscript` and `SimulatedTranscript`. `A` has access to both the real protocol transcripts obtained from actual executions of the protocol and the simulated protocol transcripts generated by the simulator $\mathbb{S}$. Furthermore, `A` can interact with the protocol, submit inputs, receive outputs, and make queries to relevant components, such as the random oracle. It can perform computations in a reasonable amount of time based on the security parameter κ. However, `A` cannot break cryptographic assumptions or solve computationally hard problems beyond its computational limits.

The simulator simulates the operations performed by the issuer to generate consistent credentials. The simulator undergoes an initialization phase where it obtains the necessary values, such as $\ddot{m}_{ID}$, $\ddot{\sigma}$, $\ddot{\sigma}_{x_0}$, $\ddot{\sigma}_{x_r}$, and $\ddot{\sigma}_{x_{ID}}$. Then, it executes the following steps:

1. inputs *timestamp*,
2. consistently generates the transaction identifier $\tilde{\mathcal{R}} = g_1^\tau$,
3. consistently computes the randomized credentials $\tilde{\sigma} = \ddot{\sigma}^\rho$, $\tilde{\sigma}_{x_0} = \ddot{\sigma}_{x_0}^\rho$, $\tilde{\sigma}_{x_r} = \ddot{\sigma}_{x_r}^\rho$, and $\tilde{\sigma}_{x_{ID}} = \ddot{\sigma}_{x_{ID}}^\rho$,
4. consistently computes the commitment $\tilde{t}_\kappa = g_1^{\rho_\kappa} \ddot{\sigma}_{x_{ID}}^{\rho_{ID}\rho} \mathcal{I}_r^\tau$,
5. encrypts λ if required,
6. consistently computes the challenge $\tilde{e} = \mathcal{H}(\tilde{t}_\kappa, \tilde{\sigma}, \tilde{\sigma}_{x_0}, \tilde{\sigma}_{x_r}, \tilde{\sigma}_{x_{ID}}, \tilde{\mathcal{R}}, timestamp, \lambda)$,
7. consistently computes the responses $\tilde{s}_\kappa = \rho_\kappa + \tilde{e}\rho$ and $\tilde{s}_{ID} = \rho_{ID} - \tilde{e}\ddot{m}_{ID}$
8. outputs $\tilde{\mathcal{R}}$, $(\tilde{\sigma}, \tilde{\sigma}_{x_0}, \tilde{\sigma}_{x_r}, \tilde{\sigma}_{x_{ID}})$, $(\tilde{e}, \tilde{s}_\kappa, \tilde{s}_{ID})$.

The proof outlines the steps executed by the simulator $\mathbb{S}$ to generate a simulated protocol transcript. The adversary can compute the commitment $\tilde{t}_\kappa$ in public environments, but it lacks the necessary shared secret key to compute it in private environments. As a result, in both real and simulated transcripts, the adversary can verify $\tilde{e}$ in public environments but is unable to decrypt λ and verify $\tilde{e}$ in private environments. Utilizing the public key of the issuer, the adversary can attempt

pairing verification. Despite the failure of the pairing verification process, the resulting protocol transcript is indistinguishable from that of a malicious user. Therefore, the simulator produces output that demonstrates computational indistinguishability when compared to the real protocol transcript. □

Proposition 9. *The proposed decentralized attribute-based authentication protocol is* ***key-parameter consistent****. I.e., generated keys and parameters conform to the defined rules, specifically by ensuring that the public keys are valid and securely paired with their corresponding private keys.*

Proof. The protocol produces the key pairs $(x_0, \mathcal{X}_0)$ and $(i_r, \mathcal{I}_r)$. To establish key-parameter consistency, for every valid secret key x_0 and i_r, the corresponding public keys $\mathcal{X}_0$ and $\mathcal{I}_r$ have to be computed correctly and satisfy the defined rules. Given a valid secret key $x_0 \in_R \mathbb{Z}_q$, the public key $\mathcal{X}_0 = g_2^{x_0}$ is computed correctly using the generator g_2 and the secret component x_0. Given a valid secret key $i_r \in_R \mathbb{Z}_q$, the public key $\mathcal{I}_r = g_1^{i_r}$ is computed correctly using the generator g_1 and the secret component i_r. These computations adhere to the defined rules, as they follow the specified procedure for deriving the public keys. □

Proposition 10. *The proposed decentralized attribute-based authentication protocol provides* ***anonymity****,* ***unlinkability****, and* ***untraceability****.*

Proof. Due to the proof of knowledge protocol's *zero-knowledge* property, the proof π is always *anonymous*, *unlinkable*, and *untraceable*. The distribution of $\hat{\sigma}$, $\hat{\sigma}_{x_0}$, $\hat{\sigma}_{x_r}$, and $\hat{\sigma}_{x_{ID}}$ is uniform and random in $\mathbb{Z}_q$ since ρ is selected uniformly and randomly from $\mathbb{Z}_q$. Consequently, the disclosed values are indistinguishable from random elements. □

Proposition 11. *The proposed decentralized attribute-based authentication protocol provides* ***confidentiality*** *and* ***integrity****. I.e., the information to be transmitted is not disclosed to unauthorized parties and is tamper-resistant.*

Proof. The *confidentiality* property is achieved by encrypting the data to be transmitted λ with the key $\mathcal{H}(t_\kappa)$. Using $\mathcal{H}(t'_\kappa)$, which is calculated utilizing the random and shareable user values ψ, verifiers can decrypt the information. Additionally, $\mathcal{R}$ and i_r are required for the computation. The value i_r is a secret key shared exclusively among system users. Therefore, an attacker will be unable to read the information since it does not possess the shared secret key. Note that in unrestricted environments, the information is not encrypted and the $\mathcal{R}^{i_r}$ computation is not conducted; hence, there is no *confidentiality*. However, the information remains immutable. The *integrity* property is accomplished by calculating the hash e. If λ or ξ are altered, $e \stackrel{?}{=} \mathcal{H}(\ldots, \lambda)$ will yield a different value, and the verification will fail. □